

**Tecnológico de Costa Rica**

**Carrera de Ingeniería Mecatrónica**



**Diseño de un sistema de adquisición de datos sobre la medida de ancho del producto y velocidad de extrusión de la máquina E16 de la empresa Polymer en El Coyol de Alajuela**

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

**Sebastián Murillo Barrantes**

**Cartago, Agosto de 2021**



Este obra está bajo una [licencia de Creative Commons Reconocimiento 4.0 Internacional](https://creativecommons.org/licenses/by/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, lunes 30 de agosto de 2021



Sebastián Murillo Barrantes

Céd: 1-1556-0941

**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: Sebastián Murillo Barrantes

Proyecto: Diseño de un sistema de adquisición de datos sobre la medida de ancho del producto y velocidad de extrusión de la máquina E16 de la empresa Polymer en El Coyol de Alajuela.

**ANA LUCIA  
MORERA  
BARQUER  
O (FIRMA)** Firmado  
digitalmente por  
ANA LUCIA  
MORERA  
BARQUERO (FIRMA)  
Fecha: 2021.08.24  
07:51:48 -06'00'

---

Ing. Ana Lucía Morera Barquero

Asesora

Cartago, 30 de agosto 2021

**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: Sebastián Murillo Barrantes

Proyecto: Diseño de un sistema de adquisición de datos sobre la medida de ancho del producto y velocidad de extrusión de la máquina E16 de la empresa Polymer en El Coyol de Alajuela.

**Miembros del jurado evaluador**

**JOHANNA  
VANESSA MUÑOZ  
PEREZ (FIRMA)**  
Firmado digitalmente por  
JOHANNA VANESSA  
MUÑOZ PEREZ (FIRMA)  
Fecha: 2021.08.30 16:27:32  
-06'00'

MSc. -Ing. Johanna Muñoz Pérez

Jurado

**IVAN ARAYA  
MENESES  
(FIRMA)**  
Firmado digitalmente por  
IVAN ARAYA MENESES  
(FIRMA)  
Fecha: 2021.08.31 07:45:04  
-06'00'

MSc. -Ing. Iván Araya Meneses

Jurado

**FELIPE GERARDO  
MEZA OBANDO  
(FIRMA)**  
Digitally signed by  
FELIPE GERARDO MEZA  
OBANDO (FIRMA)  
Date: 2021.08.30  
17:46:19 -06'00'

MSc. -Ing. Felipe Meza Obando

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, 30 de agosto 2021

## Resumen

La máquina extrusora E16 en la planta de extrusión Polyfilm en Polymer S.A. en El Coyal de Alajuela carece de un sistema de medición del ancho y la velocidad del plástico extruido, esto representa un problema importante referente a la monitorización de la producción.

En el presente informe se exploran las soluciones más adecuadas para el diseño de un sistema de adquisición de datos de velocidad, un sistema de adquisición de datos del ancho (el cual se replicará en tres etapas de producción), de una interfaz de usuario que permita visualizar y almacenar datos pertinentes sobre la producción de plástico en dicha máquina y de un sistema de alertas que permita conocer cuando las variables de interés se salen del rango permitido.

El sistema de adquisición de velocidad diseñado consta de un rodillo con imanes permanentes que excitan un sensor de efecto hall A 3144 para el cual el tiempo entre pulsaciones es cronometrado y comunicado por un microprocesador Atmega 328 a través de protocolo USART a una Raspberry Pi 4 B+. Este sistema operaría con un factor de seguridad de 341 y tiene una capacidad de medición desde los 0,5 m/min hasta los 417,38 m/min.

El sistema de adquisición de datos del ancho diseñado consiste en un sensor reflectivo CNY70 montado sobre un motor a pasos controlado por un microprocesador Atmega 328, dicho procesador también se encarga de detectar la presencia del plástico a partir de la señal del sensor, de contar los pasos dados por el motor y comunicarlos a través de protocolo USART a la misma Raspberry Pi 4 B+. Este sistema de adquisición operaría con un factor de seguridad de 15,17.

La interfaz ha sido programada en lenguaje Python con ayuda de la biblioteca Tkinter en una Raspberry Pi4. Esta se encarga de interpretar, almacenar, comunicar y mostrar visualmente los datos recibidos. Además, se encarga de notificar visualmente una alerta cuando las variables de interés salen del rango permitido.

Los datos comunicados a través de internet son redirigidos a un libro de Excel 365 almacenado en la nube con la ayuda de la plataforma Integromat. Esta plataforma, a partir de una notificación generada en la interfaz, también se encarga de enviar un correo de Outlook y un mensaje de Telegram cuando se detiene la producción en la máquina.

La inversión inicial prevista se recuperaría en aproximadamente 3 meses y 13 días y tendría un Retorno de Inversión mensual del 3834,41%.

## Abstract

The E16 film blown extrusion machine located in Polymer S.A. in El Coyol, Alajuela does not have a speed and width measuring system of extruded plastic; this represents a significant problem regarding the monitorization of the production.

In this report, the most adequate solutions for the design of an acquisition system that allows the gathering of speed of extrusion and the film width data in three main stages are going to be explored. Also, the design of a user interface that allows the visualization and storage of data, and the design of an alarm system will be addressed.

The designed extrusion speed data acquisition system consists of a set of magnets mounted on a roll that excite an A3144 hall effect sensor. An ATmega 328 microprocessor oversees the measuring of the time between pulses and communicates it to a Raspberry Pi 4 B+ over USART. This measuring system has been designed to operate at a security factor of 341 and at speed rates varying from 0,5 to 417,38 m/min.

The film width data gathering system designed consists of a reflective CNY70 sensor mounted on a platform moved by a stepper motor controlled by an ATmega 328, such microprocessor is also in charge of detecting the presence of the plastic film and keeping count of the motor steps to later communicate them to a Raspberry Pi 4 B+ over USART. This measuring system has been designed to operate at a security factor of 15,17.

The interface has been programmed in Python with the help of the Tkinter library in a Raspberry Pi 4. The aforementioned interface has the function of interpreting, storing, communicating, and visually showing the data it receives. It also oversees the notification of alarms whenever any variable goes outside of the allowed span.

The communicated data sent via Internet are forwarded to an Excel 365 workbook by the platform Integromat; this platform also sends an email via Outlook and an instant message via Telegram whenever the interface notifies the production has stopped.

The initial investment should be recovered in approximately 3 months and 6 days and its monthly return of investment should be 3834,41%.

## **Dedicatoria**

Dedico este proyecto e informe a mis padres Gilberth Murillo Mondragón y Jenny Barrantes González, y a mi hermana Natalia Abigail Murillo Barrantes.



## **Agradecimiento**

Agradezco indudablemente a Dios por todas las oportunidades y la fuerza que me ha regalado en ya tantos años de estudio. Agradezco a mis padres por ser siempre mi soporte y mi apoyo fiel. Agradezco a mi hermanita por su cariño. Agradezco a mis amigos del TEC, de Dosis de Amor y de cualquier otro lugar por acompañarme en este proceso.

Finalmente agradezco a la profesora Ana Lucía por su valiosa asesoría en el informe y al personal de Polymer por la oportunidad de realizar este proyecto tan interesante y desafiante con ellos y por el invaluable aporte durante el diseño y la ejecución.

# Lista de contenidos

<b>LISTA DE FIGURAS .....</b>	<b>IV</b>
<b>LISTA DE TABLAS .....</b>	<b>VIII</b>
<b>LISTA DE ABREVIACIONES .....</b>	<b>IX</b>
<b>CAPÍTULO 1: INTRODUCCIÓN.....</b>	<b>1</b>
ENTORNO DEL PROYECTO.....	1
DEFINICIÓN DEL PROBLEMA.....	2
SÍNTESIS DEL PROBLEMA .....	7
ENFOQUE DE LA SOLUCIÓN .....	7
OBJETIVO GENERAL.....	9
<b>OBJETIVOS ESPECÍFICOS.....</b>	<b>9</b>
<b>CAPÍTULO 2: MARCO TEÓRICO.....</b>	<b>10</b>
<b>SENSORES .....</b>	<b>10</b>
<i>Caracterización de los sensores .....</i>	<i>11</i>
<i>Función de transferencia .....</i>	<i>12</i>
<i>Modelo matemático .....</i>	<i>12</i>
<i>Propiedades de los sensores .....</i>	<i>12</i>
<i>Principios físicos de la medición con sensores .....</i>	<i>16</i>
<b>MECÁNICA DE LOS MATERIALES.....</b>	<b>25</b>
<i>Reacciones de Soporte .....</i>	<i>25</i>
<i>Ecuaciones de Equilibrio .....</i>	<i>25</i>
<i>Diagrama de fuerzas y momentos .....</i>	<i>25</i>
<i>Reacciones Internas Resultantes.....</i>	<i>26</i>
<i>Análisis Estático .....</i>	<i>26</i>
<i>Esfuerzo .....</i>	<i>27</i>
<i>Esfuerzo Normal.....</i>	<i>27</i>
<i>Esfuerzo Cortante.....</i>	<i>27</i>
<i>Diagrama de esfuerzo – deformación. ....</i>	<i>28</i>
<i>Módulo de Young .....</i>	<i>30</i>
<i>Diagramas de Momento y Fuerza Cortante .....</i>	<i>30</i>
<i>Esfuerzo Normal en Vigas.....</i>	<i>33</i>
<b>DISEÑO MECÁNICO .....</b>	<b>36</b>
<i>Factor de Seguridad.....</i>	<i>36</i>
<i>Selección del Material .....</i>	<i>36</i>
<i>Círculo de Mohr.....</i>	<i>37</i>
<i>Falla estática .....</i>	<i>40</i>
<i>Falla por Fatiga.....</i>	<i>43</i>
<b>CAPÍTULO 3: METODOLOGÍA.....</b>	<b>46</b>
ANÁLISIS DEL ESPACIO DISPONIBLE EN LA MÁQUINA PARA LOS MONTAJES DEL SISTEMA DE ADQUISICIÓN DE ANCHO Y EL SISTEMA DE ADQUISICIÓN DE VELOCIDAD ACORDE AL PROCESO PRODUCTIVO DE LA MÁQUINA .....	46
FÓRMULA DE PRODUCCIÓN.....	47
<b>SISTEMA DE MEDICIÓN DE VELOCIDAD DE PRODUCCIÓN LINEAL DE PLÁSTICO Y SU MONTAJE MECÁNICO 48</b>	<b>48</b>
<i>Definición del problema para el diseño del sistema de adquisición de velocidad y su montaje mecánico .....</i>	<i>48</i>

<i>Crterios para seleccin y validacin del sistema de adquisicin de velocidad</i> .....	49
<i>Búsqueda de informacin pertinente para la generacin de la seal del sistema de velocidad</i> .....	49
<i>Generacin de la mayor cantidad de soluciones posibles para la produccin de la seal del sistema de adquisicin de velocidad</i> .....	50
<i>Análisis y descarte de las posibles soluciones acerca de la generacin de la seal del sistema de adquisicin de velocidad</i> .....	51
<i>Crterios para el montaje y diseo mecnicos del sistema de adquisicin de velocidad</i> .....	61
<i>Búsqueda de informacin necesaria para el diseo mecánico del sistema de adquisicin de velocidad</i> .....	62
<i>Generacin y descarte de la mayor cantidad de soluciones para el diseo y el montaje mecnicos del sistema de adquisicin de velocidad</i> .....	62
SISTEMA DE MEDICIÓN DE ANCHO DEL PRODUCTO EXTRUIDO POR LA MÁQUINA E16.....	71
<i>Definición del problema relacionado a la medición del ancho</i> .....	71
<i>Generación y descarte de la mayor cantidad de soluciones para el sistema mecánico de medición del ancho</i> .....	72
<i>Crterios para la seleccin del sensor para el sistema de adquisicin de ancho</i> .....	75
<i>Búsqueda de informacin para la seleccin del sensor para el sistema de medición de ancho</i> .....	76
<i>Generación y descarte de la mayor cantidad de soluciones para el sensor en el sistema de medición de ancho</i> .....	76
<i>Búsqueda de informacin pertinente para la decisin de los materiales a utilizar para la construccin del montaje mecánico del sistema de adquisicin de la velocidad y adquisicin de ancho</i> .....	81
SOBRE LA ADQUISICIÓN DE DATOS .....	83
<i>Generación y descarte de la mayor cantidad de soluciones para el microcontrolador</i> .....	84
<i>Candidatos para el Sistema de Alertas Visuales y la Comunicacin de Datos en Línea</i> .....	89
SOBRE EL REGISTRO DE DATOS EN LÍNEA.....	93
<i>Propuesta de Candidatos para el Registro de Datos en Línea</i> .....	93
<i>Descarte de Candidatos para el Registro de Datos en Línea</i> .....	94
CRITERIOS DE VALIDACIÓN.....	95
<i>Sistema de Adquisicin de Velocidad</i> .....	95
<i>Sistema de Adquisicin de Ancho</i> .....	96
<i>Sistema de Visualizacin, Alarma y Comunicacin de Datos en Línea</i> .....	96
<b>CAPÍTULO 4: PROPUESTA DE DISEÑO</b> .....	<b>97</b>
DISEÑO DE UN SISTEMA DE ALERTAS VISUALES Y DE ADQUISICIÓN DE DATOS .....	97
<i>Interfaz</i> .....	97
<i>Programacin de la Interfaz</i> .....	108
<i>Sobre Alarma Visual</i> .....	130
<i>Resolucin de Problemas</i> .....	132
SISTEMA DE ADQUISICIÓN DE LA VELOCIDAD DE EXTRUSIÓN .....	136
<i>Seccin de Diseo Mecánico del Sistema de Adquisicin de Datos de Velocidad</i> .....	137
<i>Seccin de Diseo del Circuito Electrónico para el Sistema de Adquisicin de Datos de Velocidad</i> .....	144
SISTEMA DE ADQUISICIÓN DEL ANCHO DEL PLÁSTICO EXTRUIDO POR LA EXTRUSORA 16 .....	147
<i>Seccin de Diseo Mecánico del Sistema de Adquisicin de Ancho</i> .....	147
<i>Seccin de Diseo del Circuito para el Sistema de Adquisicin de Ancho</i> .....	152
<b>CAPÍTULO 5: RESULTADOS Y ANÁLISIS</b> .....	<b>158</b>
SISTEMA DE ADQUISICIÓN DE DATOS DE VELOCIDAD .....	158
<i>Deteccin de Cada Pulso del Sensor de Efecto Hall</i> .....	158

<i>Funcionamiento en la Posición Deseada</i> .....	160
<i>Comunicación entre el Atmega328P y la Raspberry Pi 4</i> .....	162
<i>Análisis Estadístico del Muestreo de Datos del Sistema Adquisición de Datos de Velocidad</i> .....	167
<i>Análisis Mecánico del Sistema de Adquisición de Datos de Velocidad</i> .....	169
SISTEMA DE ADQUISICIÓN DE DATOS DEL ANCHO.....	169
<i>La Detección de los Límites del Producto</i> .....	169
<i>Análisis Mecánico del Sistema de Adquisición de Ancho</i> .....	180
INTERFAZ DE USUARIO Y COMUNICACIÓN DE DATOS.....	181
<i>Cálculo Correcto de los Valores Medidos por los Sistemas de Medición</i> .....	181
<i>Activación de la Alerta Visual</i> .....	183
<i>Comunicación y Almacenamiento de los Datos en Línea</i> .....	183
ANÁLISIS ECONÓMICO.....	186
<i>Del Costo Material</i> .....	186
<i>Del Costo Humano</i> .....	186
<i>Cálculo del Beneficio</i> .....	187
<b>CAPÍTULO 6: CONCLUSIONES Y RECOMENDACIONES</b> .....	<b>190</b>
CONCLUSIONES.....	190
RECOMENDACIONES.....	191
<b>CAPÍTULO 7: REFERENCIAS BIBLIOGRÁFICAS</b> .....	<b>192</b>
<b>CAPÍTULO 8: APÉNDICES</b> .....	<b>198</b>
APÉNDICE A - PLANOS DEL SISTEMA DE MEDICIÓN DE ANCHO.....	198
APÉNDICE B – PLANOS DEL SISTEMA DE ADQUISICIÓN DE VELOCIDAD.....	207
APÉNDICE C – ESQUEMÁTICO Y PCB DEL SISTEMA DE ADQUISICIÓN DEL ANCHO.....	212
APÉNDICE D- ESQUEMÁTICO Y PCB DEL SISTEMA DE ADQUISICIÓN DE VELOCIDAD.....	214
APÉNDICE E – ESQUEMÁTICO Y PCB DEL SISTEMA DE ALARMA.....	216
APÉNDICE F – CÓDIGO FUENTE DEL MICROPROCESADOR DEL SISTEMA DE ADQUISICIÓN DE ANCHO.....	217
APÉNDICE G – CÓDIGO FUENTE DEL MICROPROCESADOR DEL SISTEMA DE ADQUISICIÓN DE VELOCIDAD.....	219
APÉNDICE H – CÓDIGO PYTHON PARA LA INTERFAZ DE USUARIO.....	220
APÉNDICE I – CÓDIGO PYTHON DE LA BIBLIOTECA FUNCIONES.....	230
APÉNDICE J – CÓDIGO PYTHON DE LA BIBLIOTECA VARIABLES.....	241
APÉNDICE K – CÓDIGO PYTHON DE LA BIBLIOTECA IMPRODUCTIVOS.....	241
<b>CAPÍTULO 9: ANEXOS</b> .....	<b>242</b>
ANEXO A – CÓDIGO PYTHON PARA IDENTIFICACIÓN DE PUERTOS SERIALES [35].....	242
ANEXO B – BIBLIOTECA USART.H.....	243
ANEXO C- BIBLIOTECA USART.C.....	244

## Lista de Figuras

Figura 1. Tolva (al fondo) y resina granulada (al frente). Fuente: Elaboración propia. ....	2
Figura 2. Etapa de soplado del plástico fundido. Fuente: Elaboración propia. ....	3
Figura 3. Prensa superior (arriba), globo (atrás), lámina (frente). Fuente: Elaboración propia. ....	4
Figura 4. Rodillo de presión y contador de pasos. Fuente: Elaboración propia. ....	5
Figura 5. Las cuchillas se pueden observar a los lados de la filmina. En este caso no se están utilizando. Fuente: Elaboración propia. ....	6
Figura 6. De frente se puede ver la bobina de plástico y atrás todas las partes de la máquina. Fuente: Elaboración propia. ....	6
Figura 7. Representación visual del error por histéresis. Fuente: Adaptado de [1]. ....	14
Figura 8. No-linealidad: la recta de mejor ajuste está definida por $z'$ y $z$ , sin embargo, la señal se mueve entre un rango determinado. Fuente: adaptado de [1]. ....	15
Figura 9. Representación de los comportamientos de reflexión y refracción de la luz. Fuente: adaptado de [3]. ....	21
Figura 10. Ejemplos de manipulación de la luz, a) refracción por lentes, b) reflexión en espejo rotativo, c) reflexión y conducción en cable de fibra óptica. Fuente: adaptado de [1]. ....	22
Figura 11. Geometría de un lente planoconvexo. Fuente: adaptado de [1]. ....	24
Figura 12. Representación visual de la acción que provoca un esfuerzo cortante. Fuente: adaptado de [4]. ....	28
Figura 13. Diagrama de esfuerzo-deformación de un material dúctil (sin escala). Fuente: adaptado de [4]. ....	29
Ilustración 14. Convención de signos para los diagramas de cortante y momento. Fuente: adaptado de [4]. ....	31
Figura 15. Diagramas de Cortante y Momento Flector de una barra simplemente suspendida a la que se le aplica una distribución de carga constante $w$ . Fuente: adaptado de [4]. ....	32
Figura 16. Análisis estático de la viga en la posición $x$ para la construcción de los diagramas en la Figura 20. Fuente: adaptado de [4]. ....	32
Figura 17. Esfuerzos cortantes máximos de acuerdo con perfil de la viga. Fuente: adaptado de [5]. ....	33
Figura 18. Representación visual de la deformación normal a lo largo de la sección transversal. Fuente: adaptado de [4]. ....	33
Figura 19. Representación visual del esfuerzo normal a lo largo de la sección transversal. Fuente: adaptado de [4]. ....	34
Figura 20. Se puede observar un punto de inflexión en la coordenada C. Fuente: adaptado de [1]. ....	35
Figura 21. Esfuerzos tridimensionales para un punto del material. Fuente: adaptado de [5]. ....	37
Figura 22. Representación simplificada de los esfuerzos en un punto del material. Fuente: tomado de [5]. ....	37
Figura 23. Corte oblicuo realizado al elemento de la Figura 22. Fuente: adaptado de [5]. ....	38
Figura 24. Representación gráfica del círculo de Mohr. Fuente: tomado de [5]. ....	39
Figura 25. Ejemplo de falla estática por esfuerzo normal en un perno. Fuente: adaptado de [5]. ....	40
Figura 26. Falla estática por cortante en un resorte. Fuente: adaptado de [5]. ....	41
Figura 27. Representación de la zona segura de operación para un material según la teoría del Esfuerzo cortante máximo. Fuente: adaptado de [5]. ....	43
Figura 28. Inicio de las microgrietas en A, formación de macrogrietas en B y falla catastrófica en C en un perno. Fuente: adaptado de [5]. ....	44
Figura 29. Ubicación de las distintas zonas para el análisis de fatiga en un acero. Fuente: adaptado de [5]. ....	45
Figura 30. Sensor VL6180X. Fuente: adaptado de [10]. ....	54
Figura 31. Isométrico y representación gráfica de sus conos ópticos. Fuente: adaptado de [10]. ....	55
Figura 32. Cámaras Basler Racer. Fuente: adaptado de [11]. ....	56
Figura 33. Diagrama del diafragma piezoeléctrico 10293. Fuente: adaptado de [12]. ....	57

Figura 34. Módulo HC-SR04. Fuente: Adaptado de [13].	58
Figura 35. Isométrico del codificador Wurth Elektronik 482009514001. Fuente: adaptado de [8].	59
Figura 36. Sensor de efecto Hall 3144. Fuente: adaptado de [9].	60
Figura 37. Ilustración de propuesta del electroimán. Fuente: Elaboración propia.	63
Figura 38. Representación visual del montaje del sensor de efecto Hall en el rodillo de la extrusora. Fuente: Elaboración propia.	64
Figura 39. Representación del sensor sobre un rodillo aparte. Fuente: Elaboración propia.	65
Figura 40. Representación de la colocación de imanes permanentes en un rodillo aparte para la adquisición de la velocidad. Fuente: Elaboración propia.	66
Figura 41. Flujo de oportunidad para los sistemas de medición de velocidad. Fuente: elaboración propia.	70
Figura 42. Sensor matricial 11484. Fuente: adaptado de [18].	73
Figura 43. Motor a pasos Nema 17. Fuente: tomado de [19].	75
Figura 44. Sensor capacitivo LJC18A3-B-Z/BX. Fuente: adaptado de [16].	78
Figura 45. Sensor resistivo FSR 402. Fuente: adaptado de [15].	79
Figura 46. Sensor CNY70. Fuente: adaptado de [22].	80
Figura 47. Flujo de oportunidad para la selección del sistema de adquisición de datos del ancho. Fuente: propia.	81
Figura 48. PIC4550. Fuente: adaptado de [16].	86
Figura 49. ATTINY84. Fuente: adaptado de [16].	86
Figura 50. ATmega 328 PU. Fuente: adaptado de [15].	87
Figura 51. Flujo de oportunidad para la selección del microprocesador. Fuente: elaboración propia.	89
Figura 52. Flujo de oportunidad para la selección del centro de la computadora. Fuente: Elaboración propia.	92
Figura 53. Flujo de oportunidad para la selección de la plataforma de registro de datos en línea. Fuente: elaboración propia.	95
Figura 54. Pantalla Inicial de Interfaz de Usuario. Fuente: elaboración propia.	98
Figura 55. Selección del tipo de hora. Fuente: Elaboración propia.	100
Figura 56. Información sobre el paro. Fuente: Elaboración propia.	100
Figura 57. El valor de producción real parpadea en rojo cuando sale del margen. Fuente: Elaboración propia.	100
Figura 58. Pantalla de Autorización. Fuente: Elaboración propia.	101
Figura 59. Pantalla de configuración con los valores predeterminados. Fuente: Elaboración propia.	101
Figura 60. Celdas de alerta en rojo debido a que los límites de tolerancia han sido sobrepasados. Fuente: Elaboración propia.	102
Figura 61. Ventana de modificación de la lista de operadores. Fuente: Elaboración propia.	102
Figura 62. Sección de selección de operador, nombre y número ocultos por confidencialidad. Fuente: Elaboración propia.	103
Figura 63. Aviso sobre número de operador desconocido. Fuente: Elaboración propia.	103
Figura 64. Ventana de alerta por detención. Fuente: Elaboración propia.	104
Figura 65. Restricción para selección de razón de detención. Fuente: Elaboración propia.	106
Figura 66. Botón primario de salida. Fuente: Elaboración propia.	107
Figura 67. Botón secundario de salida. Fuente: Elaboración propia.	107
Figura 68. Botón de ocultamiento. Fuente: elaboración propia.	108
Figura 69. Botón "Mostrar Interfaz". Fuente: Elaboración propia.	108
Figura 70. Distribución de las cajas principales de la interfaz. Fuente: Elaboración propia.	110
Figura 71. produ_set corresponde al texto inferior "110". Fuente: Elaboración propia.	111
Figura 72. nombre_operador corresponde a la primera casilla de la segunda columna. Fuente: propia.	111

Figura 73. Iteración para la alternancia del color en las opciones de produ_presentacion. Fuente: propia.....	112
Figura 74. Grupo de botones de nombre produ_presentacion. Fuente: Elaboración propia. ....	112
Figura 75. Botón para acceder a la ventana de autorización. Fuente: Elaboración propia. ....	113
Figura 76. La variable tipo_hora está a la derecha de color verde. Fuente: Elaboración propia. ..	113
Figura 77. Inicialización de la comunicación serial. Fuente: Elaboración propia.....	114
Figura 78. Adecuación de la información serial entrante. Fuente: Elaboración propia. ....	114
Figura 79. Lista persistente de operadores. Fuente: propia. ....	115
Figura 80. Almacenamiento de cambios y cierre de archivo. Fuente: Elaboración propia.....	115
Figura 81. Definición de la función name(). Fuente: Elaboración propia. ....	121
Figura 82. Ejemplo de código para adjuntar datos en el archivo stream. Fuente: Elaboración propia. ....	121
Figura 83. Registro textual de datos. Fuente: Elaboración propia.....	122
Figura 84. Archivo de valores separados por comas. Fuente: Elaboración propia. ....	122
Figura 85. Definición y configuración de la función temp_setup(). Fuente: Elaboración propia.....	123
Figura 86. Escenarios de Integromat creados para la interfaz. Fuente: Elaboración propia.....	125
Figura 87. Representación gráfica del escenario 1. Fuente: Elaboración propia. ....	125
Figura 88. Modulo disparador Webhooks para el escenario 1. Fuente: Elaboración propia. ....	126
Figura 89. Modulo Microsoft 365 Excel encargado de agregar la información al documento. Fuente: Elaboración propia. ....	126
Figura 90. Escenario 2. Fuente: Elaboración propia.....	127
Figura 91. Módulo Microsoft 365 Email para el escenario 2. Fuente: Elaboración propia. ....	127
Figura 92. Escenario 3. Fuente: Elaboración propia.....	128
Figura 93. Configuración del módulo de Telegram. Fuente: Elaboración propia. ....	129
Figura 94. Mensaje de alerta enviado por el Bot de Telegram. Fuente: propia.....	129
Figura 95. Selección de pines GPIO para la torreta. Fuente: Elaboración propia.....	130
Figura 96. Código fuente en primera ejecución. Fuente: Elaboración propia.....	131
Figura 97. Panel de instrucciones del compilador Thonny. Fuente: Elaboración propia.....	132
Figura 98. Correo de aviso de detención del escenario de Integromat. Fuente: Elaboración propia. ....	133
Figura 99. Panel de herramientas de Integromat. Fuente: adaptado de [3].....	134
Figura 100. Pestaña INCOMPLETE EXECUTIONS. Fuente: adaptado de [3]. ....	134
Figura 101. Información correspondiente a cada adaptador USB-TTL (pasos 5 y 6). Fuente: Elaboración propia. ....	135
Figura 102. Rodillo de presión y rejilla de seguridad. Fuente: Elaboración propia.....	136
Figura 103. Rueda con imanes incrustados. Fuente: Elaboración propia. ....	137
Figura 104. Vista con transparencia de la rueda para la adquisición de la velocidad, sin escala. Fuente: Elaboración propia. ....	139
Figura 105. Dimensiones del eje para la rueda, sin escala. Fuente: Elaboración propia.....	140
Figura 106. Ensamblaje simulado en SolidWorks. Fuente: Elaboración propia. ....	141
Figura 107. Sección transversal de brazo inferior. Fuente: propia. ....	142
Figura 108. Área transversal menor en la base. Fuente: Elaboración propia. ....	142
Figura 109. Vista de corte del pasador que sujeta el brazo a la base. Fuente: Elaboración propia. ....	143
Figura 110. Modelo 3D del carrito completo. Simulado en SolidWorks [43]. Fuente: Elaboración propia.....	148
Figura 111. Representación de la Fuerza Cortante a lo largo de una barra de soporte según la posición del carrito. Generado con Python [54].....	149
Figura 112. Representación del Momento Flector a lo largo de una barra de soporte según la posición del carrito. Generado con Python [54].....	150
Figura 113. Representación gráfica de la conexión del disparador Schmitt. Fuente: propia, confeccionado en EAGLE [56]. ....	153

Figura 114. Terminal atornillada. Fuente: tomado de [50].	154
Figura 115. Circuito del sistema de adquisición de velocidad. Fuente: elaboración propia.	159
Figura 116. Prototipo del circuito para la visualización del pulso, el cable verde transporta la señal del sensor A3144. Fuente: elaboración propia.	159
Figura 117. Observación de los pulsos registrados por el sensor de efecto Hall. Fuente: elaboración propia.	160
Figura 118. Prueba preliminar de fricción entre la rueda y el rodillo de presión. Fuente: elaboración propia.	161
Figura 119. Adición de la cinta de caucho alrededor de la rueda. Fuente: elaboración propia.	161
Figura 120. Rueda de imanes en la posición final. Fuente: elaboración propia.	162
Figura 121. Prototipo de prueba para mantener velocidad constante. Fuente: elaboración propia.	163
Figura 122. Visualización del tren de pulsos enviados por el ATmega 328p hasta la Raspberry Pi 4. Fuente: elaboración propia.	164
Figura 123. Interpretación de los datos recibidos por comunicación Serial. Fuente: elaboración propia.	164
Figura 124. El momento en el que aparece la alerta en el prototipo de la interfaz. Fuente: elaboración propia.	165
Figura 125. Pantalla de alerta de parada. Fuente: elaboración propia.	166
Figura 126. Tacómetro digital de mano utilizado. Fuente: elaboración propia.	167
Figura 127. Representación de la hoja utilizada para la primera prueba. Fuente: elaboración propia.	170
Figura 128. Rodamiento utilizado previo a la segunda prueba. Fuente: adaptado de [70].	172
Figura 129. Configuración para la segunda prueba. Fuente: elaboración propia.	173
Figura 130. Muestras visualizados en la interfaz. Fuente: elaboración propia.	174
Figura 131. Tercera prueba con producto negro. Fuente: elaboración propia.	175
Figura 132. Tiempo de caída de la señal sin capacitor. Fuente: elaboración propia.	176
Figura 133. Tiempo de caída de la señal con capacitor. Fuente: elaboración propia.	176
Figura 134. Prototipo instalado en la máquina. Fuente: elaboración propia.	179
Figura 135. Prueba de visualización de datos. Fuente: elaboración propia.	181
Figura 136. Prueba con los 3 hilos para los 3 dispositivos de medición de ancho. Fuente: elaboración propia.	182
Figura 137. Mensaje enviado por el escenario de Integromat a través de la aplicación Telegram. Fuente: elaboración propia.	183
Figura 138. Mensajes enviados a través de correo electrónico por el escenario de Integromat. Fuente: elaboración propia.	183
Figura 139. Datos recibidos por Excel 365, enviados por la Raspberry Pi 4. Fuente: elaboración propia.	184
Figura 140. Registro en formato .CSV de los datos adquiridos. Fuente: elaboración propia.	184
Figura 141. Registro en formato .CSV enviado por correo electrónico. Fuente: elaboración propia.	185
Figura 142. Flujo de Caja para Análisis Económico. Fuente: elaboración propia.	188



## Lista de Tablas

<i>Tabla 1. Comparación para cálculo de puntuación ponderada de los candidatos. Fuente: elaboración propia. ....</i>	<i>61</i>
<i>Tabla 2. Comparación de las opciones para la construcción del sistema de adquisición de la velocidad. Fuente: elaboración propia. ....</i>	<i>70</i>
<i>Tabla 3. Comparación entre propuestas para el sistema de adquisición del ancho. Fuente: elaboración propia. ....</i>	<i>75</i>
<i>Tabla 4. Comparación de los sensores posibles para el sistema de medición de ancho. Fuente: elaboración propia. ....</i>	<i>80</i>
<i>Tabla 5. Tabla de comparación de candidatos para la adquisición de las señales y el control del motor a pasos. Fuente: elaboración propia. ....</i>	<i>88</i>
<i>Tabla 6. Cálculo de la puntuación ponderada para los candidatos del Sistema de Alertas Visuales y Comunicación de Datos en Línea. Fuente: elaboración propia. ....</i>	<i>92</i>
<i>Tabla 7. Tabla de calificación ponderada para la selección de la plataforma de registro de datos. Fuente: elaboración propia. ....</i>	<i>95</i>
<i>Tabla 8. Muestreo para el sistema de medición de velocidad. Fuente: elaboración propia. ....</i>	<i>167</i>
<i>Tabla 9. Muestreo realizado para la tercera prueba. Fuente: elaboración propia. ....</i>	<i>177</i>

## Lista de Abreviaciones

API

Interfaz de Programación de  
Aplicación por sus siglas en  
inglés., 94

ECM

Esfuerzo cortante máximo, 41, 42

*M. Momento flector*

Momento flector, 26

*N. Fuerza normal*

Fuerza normal, 26

*T*

Torque, 26

TIR

Tasa Interna de Retorno, 188

*V. Fuerza cortante, Fuerza cortante*

Fuerza cortante, 26

# Capítulo 1: Introducción

## Entorno del Proyecto

La empresa POLYMER S.A. Es 100% costarricense y fue fundada en el año 1961 en Barrio Cuba, San José. Inicialmente su nombre era Industrias Nacionales de Plástico (INPLANA).

En el año 1963, debido a su expansión, se traslada a la Uruca para posteriormente, en los años 1964 y 1965, diversificar la producción al introducir productos de uso doméstico.

Durante el año 1969, el entonces dueño de origen norteamericano, decide venderla a la United Fruit Company y su nombre se cambia a POLYMER S.A.; se invierte en equipos de inyección y se empieza a producir envases para la empresa Numar. Al mismo tiempo se comenzó a producir otro tipo de productos como: el suncho plástico, celofán impreso, termoformados y láminas decorativas.

En el año 1970, se inaugura el Departamento de Etiquetas Autoadhesivas Bananeras Domésticas.

Posteriormente, en el año 1985, la expansión continua y se decide dividir la producción en distintas plantas ubicadas en diferentes espacios geográficos que siguen compartiendo la misma localidad administrativa.

En el año 2005 la compañía POLYMER S.A. es adquirida por capital costarricense.

Además, en el año 2007, se adquiere la compañía de nombre Rotoflex y se cambia su nombre a Polyflex.

En mayo de 2009, se inicia el traslado de las instalaciones de POLYMER hacia el Parque Empresarial Novapark en el Coyol de Alajuela. El complejo empresarial POLYMER S.A. en el Coyol está conformado por tres plantas: Polyfilm, Polypak y Polyflex; entre ellas cuentan con más de 700 colaboradores que incluyen el personal de oficina y el personal de planta.

La empresa POLYMER S.A. exporta a República Dominicana, Puerto Rico, México, Brasil y con menor frecuencia, a países africanos. [1]

Actualmente cuentan con 33 máquinas de extrusión laminar de plástico en las que la mayoría de ellas carecen de un sistema de medición de velocidad y ancho que permita tanto al personal de planta como a la administración, el monitoreo a distancia de su producción.

## Definición del Problema

La Máquina E16 es una extrusora de polímero soplado que consta de las siguientes etapas:

### 1. Alimentación de la resina:

La materia prima en forma de plástico granulado se introduce en un contenedor metálico donde es movida por un motor con transportador de tornillo que desplaza el producto con una rapidez controlada.



*Figura 1. Tolva (al fondo) y resina granulada (al frente). Fuente: Elaboración propia.*

### 2. Fundido:

La resina granulada es arrastrada por un transportador de tornillo movido por el motor hasta la posición en la que se calentará a su punto de fusión para posteriormente ser soplada.

### 3. Soplado:

Una vez en posición, el elemento de la máquina conocido como 'dado' se encarga de darle forma cilíndrica al plástico e inyectarle aire a presión para así mantener su forma.



*Figura 2. Etapa de soplado del plástico fundido. Fuente: Elaboración propia.*

La etapa de soplado controla directamente el ancho inicial del tubo e indirectamente el desperdicio de material. Al cilindro se le conoce como 'globo'.

#### 4. Prensado:

Esta etapa es llevada a cabo por la prensa superior, encargada de darle al globo su forma laminar, dentro de esta misma etapa se le da un tratamiento superficial al polímero para que se facilite la impresión de estampados y diseños.



*Figura 3. Prensa superior (arriba), globo (atrás), lámina (frente). Fuente: Elaboración propia.*

#### 5. Estirado:

La lámina se hace llegar hasta el rodillo de presión; este rodillo de acero está en contacto directo con un rodillo de caucho para estirar la lámina y poder darle movilidad. En esta etapa se controla la velocidad de extrusión con la que sale el producto de la máquina y de forma indirecta se controla el espesor.



*Figura 4. Rodillo de presión y contador de pasos. Fuente: Elaboración propia.*

Actualmente, el monitoreo de la velocidad se lleva con un cuentapasos conectado a una pantalla separada, este dato es revisado por el operador continuamente, sin embargo, no es utilizado para ningún cálculo ni mucho menos para una alerta automática. La veracidad de este dato está sujeto al error humano y la alerta, en caso de velocidades de producción menores a las tolerables, está bajo el cargo y buena voluntad del operador.

#### 6. Cortado:

Posterior al estirado, el tubo aplastado de plástico se hace pasar por cuchillas laterales para darle el ancho final y para separar el tubo en dos láminas simples. En esta etapa se controlan directamente el ancho final y la presentación, la cual puede ser lámina, tubo o semi-laminar en la que solo se corta un lado del producto.

En esta etapa se recoge el desperdicio generado por el proceso de corte; qué tanto desperdicio se genere depende del ancho del globo en la etapa de soplado y del corte hecho.



*Figura 5. Las cuchillas se pueden observar a los lados de la filmina. En este caso no se están utilizando. Fuente: Elaboración propia.*

#### 7. Rebobinado:

Esta es la etapa final en lo que corresponde al proceso de extrusión. Si la presentación es laminar, se utilizarán ambos lados, si la presentación es tubular o semi-laminar, se utilizará un solo lado.



*Figura 6. De frente se puede ver la bobina de plástico y atrás todas las partes de la máquina. Fuente: Elaboración propia.*



Los valores del ancho se miden con cinta métrica, lo que representa una gran fuente de incertidumbre debido a los errores de paralaje y al error humano al extraer el dato, el espesor se obtiene por separado en un muestreo realizado en laboratorio y la velocidad se mide con un contador de pasos.

No se cuenta con un sistema de medición automático de la velocidad y datos de producción.

Tampoco se cuenta con un sistema de medición, que facilite el cálculo del desperdicio generado por el corte final, ni de un sistema que asegure que el ancho final en las bobinas de producto sea el correcto, pues, la medición actualmente se lleva a cabo con una cinta métrica.

No se lleva un control adecuado de los valores reportados por el dispositivo que dependen únicamente del reporte que realiza el operador. Adicionalmente, no se cuenta con un sistema visual que notifique del momento preciso en el que la extrusora en cuestión detiene su operación. Este aviso se da únicamente cuando el operador observa que la máquina no se encuentra trabajando mientras hace su recorrido, dicha situación provoca una pérdida de tiempo de producción y de control estadístico.

### **Síntesis del Problema**

Se requiere del diseño de un sistema de adquisición de datos de velocidad lineal de extrusión de producto plástico, del ancho de dicho producto generado por la extrusora E16, además, se requiere el diseño de un sistema de recopilación de estos datos de manera que estén disponibles para su análisis y control en la línea de producción. Adicionalmente, se requiere de un sistema de alerta visual que avise a los operadores de la planta y al personal de supervisión del momento en que haya un paro del funcionamiento en dicha extrusora.

### **Enfoque de la Solución**

A pesar de la existencia de máquinas que cuentan con la medición y la comunicación integrada de datos de velocidad lineal, la extrusora E16 se encuentra en buen estado y su vida útil aun es apreciable, además, en los alrededores de la máquina como tal existe el espacio físico suficiente para un montaje seguro y sencillo de un nuevo sistema de adquisición de velocidad, de ancho y de espesor. Por lo tanto, tomando en cuenta el gran costo económico de una extrusora nueva más actualizada(entre 15000 USD y 100000 USD [77]), se considera que el diseño

de un nuevo sistema de medición de velocidad lineal y del ancho del producto es lo más cómodo y económicamente beneficioso.

Seguidamente se describirá el enfoque de la solución:

- Estudio del proceso de producción de la planta: es completamente necesario el conocimiento detallado de la forma en que opera la planta y sus extrusoras para un entendimiento global y detallado de la forma de operación y consecuentemente la confección de un plan de acción acorde a los recursos, cronogramas y objetivos de la empresa.
- Familiarización con la extrusora E16: para conseguir una noción correcta del enfoque del diseño es necesario familiarizarse tanto con el proceso de producción de la máquina E16 como con su ubicación física y espacios libres.
- Valoración de posibles soluciones: junto con el ingeniero eléctrico de la planta y el supervisor de planta se discutirán y analizarán las posibles soluciones sobre el diseño más adecuado del sistema de medición de velocidad lineal, sistema de medición del ancho y posteriormente el diseño de un sistema de comunicación de datos para su visualización remota.
- Validación de la solución propuesta: El funcionamiento de la alternativa seleccionada se realizará a través de cálculos técnicos, simulaciones y un análisis económico de la solución. Como criterio adicional se tomará en cuenta la retroalimentación brindada por la gerencia.
- Diseño mecánico del ensamblaje del sistema de medición de velocidad y del sistema de visualización de datos: para la medición fiable y pronta de la velocidad lineal que permita el cálculo de producción en kg/hora se requiere un diseño adecuado de un nuevo sistema de adquisición de velocidad y un sistema de visualización de datos en la máquina.
- Diseño mecánico del sistema de adquisición de ancho: para la obtención del dato del ancho de la película es necesario diseñar el sistema de adquisición junto a su montaje y su circuito.
- Diseño de un sistema de comunicación de datos: se requiere también que los datos sean observados y almacenados en tiempo real para que se permita al departamento administrativo realizar los cálculos pertinentes, correspondientes a la producción del producto y al control estadístico.
- Propuesta de un sistema de visualización móvil completo: Se considerarán distintas opciones y procedimientos posibles para la visualización de los datos obtenidos en distintos dispositivos móviles. Incluyendo en ella la información de la razón de paro para la máquina.

## **Objetivo General**

Diseñar un sistema mecatrónico para el monitoreo de las dimensiones del ancho y la velocidad de extrusión de plástico en la máquina E16 en la planta Polymer en el Coyol de Alajuela.

## **Objetivos Específicos**

- Diseñar un sistema de adquisición de datos junto a su montaje mecánico para la medición de velocidad de producción lineal de producto plástico extruido en la extrusora E16.
- Diseñar el sistema de medición de ancho del producto extruido por la máquina E16.
- Diseñar un sistema de alertas visuales y de adquisición de datos que permita su utilización por parte del área gerencial.
- Realizar un análisis económico de la solución planteada.

## Capítulo 2: Marco Teórico

Todo proceso automático o máquina autónoma requiere de la capacidad de observar y cuantificar el entorno en el que cumplirá su función, esto es: las variables físicas de la materia prima para el proceso que se llevará a cabo. En el entorno del presente proyecto las variables que se medirán serán principalmente la distancia entre el inicio y el final de la extensión transversal del plástico extruido y la velocidad con la que es entregado por la máquina. Para definir los conceptos necesarios, se explicarán los fenómenos físicos que imperan en dichas variables, las propiedades físicas que se medirán, los protocolos utilizados en su medición y los instrumentos con los que serán medidos: luz, magnetismo, microprocesadores, protocolos de comunicación, programación de interfaces gráficas, entre otros.

Además, también es necesario tomar en cuenta las propiedades mecánicas de los elementos de soporte y unión, y utilizar las técnicas adecuadas de diseño mecánico para que el producto final soporte las condiciones del entorno y las características propias de los equipos usados. Para esto también se expondrán los principios de resistencia y diseño mecánicos.

### Sensores

Según [1], un sensor es aquel dispositivo que recibe un estímulo y responde con una señal eléctrica. El estímulo es la cantidad, propiedad o condición medida por el dispositivo y posteriormente convertida en la señal eléctrica transmitida. Dicha señal, además, tiene que ser compatible con el circuito que se va a utilizar para procesarla; esta señal puede estar en la forma de corriente, voltaje o carga eléctrica, esto es conocido como el “formato de señal de salida”.

Lo anterior se puede sintetizar asegurando que un sensor tiene propiedades de entrada y propiedades eléctricas de salida; y el estímulo de entrada, que puede ser de cualquier tipo de naturaleza física o química, es convertido de acuerdo con dichas propiedades. La relación entre su entrada y su salida es conocida como “función de transferencia”, y será descrita más adelante.

Es necesario también señalar que un sensor es solo una parte en un sistema de adquisición de información. Y tal sensor puede estar en la etapa inicial del sistema donde se encarga de la medición directa de la cantidad, propiedad o condición; o en la etapa interna para mantener el control del funcionamiento del sistema de adquisición de información.

## Caracterización de los sensores

Es importante también mencionar que existen dos grandes grupos de sensores, los sensores sencillos y los compuestos [1]:

- Sencillos: convierten directamente la señal de entrada en eléctrica a través de determinada reacción fisicoquímica.
- Compuestos: como su nombre lo señala, están compuestos por uno o más transductores y un sensor sencillo.

Los transductores son elementos o dispositivos que convierten un tipo de energía en otro que no sea eléctrico; por ejemplo: una bocina, que convierte energía eléctrica en energía mecánica y sónica.

Otra gran caracterización de los sensores es su grado de pasividad; pueden ser activos o pasivos:

- Pasivos: generan las señales eléctricas sin necesidad de consumir energía desde el circuito.
- Activos: requiere de una señal de operación que es proporcionada por el circuito de alimentación y dicha señal es modificada de acuerdo con la entrada del sensor. Los sensores activos son también conocidos como paramétricos pues “los parámetros del sensor modulan la señal de excitación y la modulación transporta la información del valor medido” [1: 7].

De acuerdo con la referencia utilizada, los sensores se pueden clasificar como absolutos o relativos:

- Absolutos: detecta un estímulo en relación con una escala física absoluta, y esta es independiente de las condiciones de medición.
- Relativos: el estímulo detectado es relacionado a una escala preexistente y la señal de salida se interpreta de acuerdo con la referencia utilizada en dicha medición.

Los sensores también se pueden clasificar de acuerdo con sus propiedades de interés:

- Especificaciones del sensor, en: sensibilidad, rango, exactitud, histéresis, resolución, linealidad, entre otros.
- Material del sensor, en: orgánico, biológico, inorgánico, aislante, conductor, semiconductor, entre otros.
- Propiedad de detección, en: biológico, químico, eléctrico, magnético o electromagnético, térmico, mecánico, de radioactividad, óptico, acústico, entre otros.

## Función de transferencia

Como se comentó previamente, la función de transferencia es la relación, invariable en el tiempo, entre el estímulo, representado como  $s$  y la señal eléctrica, representada como  $S$ . Esta puede ser escrita como  $S = f(s)$ . Cuando se conoce la función de transferencia  $S = f(s)$ , y la señal de salida  $S$ , sin embargo, se desconoce el estímulo  $s$ ; para obtenerlo se requiere la aplicación de la fórmula inversa

$$S = f^{-1}(S). \quad (1.1)$$

## Modelo matemático

Siempre existirá una ley física o química que regirá el comportamiento de un sensor, esta ley necesariamente debe permitir ser expresada como una fórmula matemática, dicha fórmula es luego utilizada para calcular la función inversa del sensor de interés y por consiguiente se podrá obtener el valor del estímulo  $s$  a partir de la señal  $S$ .

En la práctica, no existe una fórmula previamente conocida para cada sensor por lo que es necesario recurrir a aproximaciones de las funciones de transferencia inversas y directas [1].

## Propiedades de los sensores

### Ámbito

Es el rango dinámico de todos los valores de medición que puede recibir un sensor antes de retornar una respuesta con un grado de exactitud inaceptable. En sensores con rangos dinámicos muy grandes o con respuestas no lineales, su ámbito se expresa en decibeles (dB).

Los decibeles son “una medida logarítmica de la relación de, ya sea, potencia o fuerza (voltaje)” [1: 30]. La escala logarítmica permite mantener una alta resolución cuando los valores son pequeños, a la vez que se mantiene una apreciación a gran escala cuando los valores son muy altos.

La fórmula utilizada para calcular los decibeles es:

$$1 \text{ dB} = 20 \log \frac{S_2}{S_1} \quad (1.2)$$

## Salida a escala completa

FSO, por sus siglas en inglés, comprende la cantidad absoluta entre el valor de señal de salida cuando se aplica el mínimo estímulo detectable, sustraído del valor de la señal de salida cuando se aplica el máximo estímulo detectable:

$$FSO = f(s_{max}) - f(s_{min}) \quad (1.3)$$

## Precisión

Corresponde a la desviación máxima dada por el sensor en comparación al valor de salida ideal, dado por su función de transferencia, según el estímulo recibido. Puede ser descrita como la máxima diferencia absoluta entre los valores generados por el sensor a lo largo de múltiples mediciones y el que se espera teóricamente dado cierto estímulo previamente conocido.

Es sumamente importante conocer esta magnitud pues en la práctica nunca se tendrá un valor perfectamente coincidente con el que se espera en la teoría debido a las imperfecciones de manufactura, la histéresis, la respuesta en frecuencia y la variabilidad de las materias primas, etc., por esta razón cuando se diseña un sistema que depende de uno o varios sensores, se debe tomar en cuenta el peor escenario posible respecto al desempeño del sensor en términos de precisión.

La precisión puede ser expresada en 4 diferentes formas [1]:

- Directamente en términos del valor medido: se utiliza cuando el error es independiente de la magnitud de entrada.
- En porcentaje del ámbito de entrada a escala completa: es muy útil cuando la función de transferencia es lineal y se relaciona grandemente a la forma anterior pues prácticamente todos los sensores tienen un ámbito predefinido.
- En porcentaje de la señal medida: el error se expresa como una fracción de la señal medida. Es sumamente útil para sensores que no tienen una función de transferencia lineal.
- En términos de la señal de salida: es típicamente útil para sensores con un formato de salida digital pues permite expresar el error en términos del bit menos significativo.

## Error de calibración

Es la imprecisión permitida por la casa de manufactura cuando la calibración de dicho sensor se realiza en la fábrica. Este error es de naturaleza sistémica y se representa como una constante sumada a la señal esperada en cierto rango de valores. No necesariamente se mantendrá constante a lo largo del ámbito de la señal.

## Histéresis

El error por histéresis se observa cuando la salida varía en un punto específico de la entrada, según la dirección en la que se aplique el estímulo.

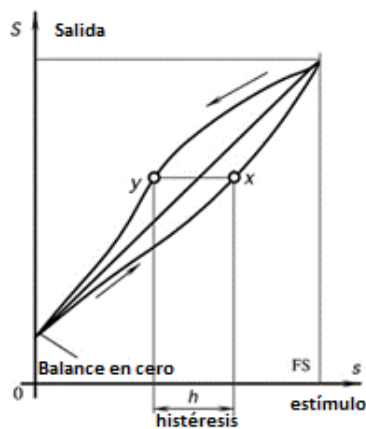


Figura 7. Representación visual del error por histéresis. Fuente: Adaptado de [1]

En el caso de la Figura 7, el punto se fija en la señal de salida y se puede observar cómo en sentido de izquierda a derecha se requiere un valor  $x$  para producir el mismo valor de salida generado por dicho valor  $y$  en dirección derecha a izquierda.

## No linealidad

Este error es especificado en sensores para los cuales su función de transferencia puede ser aproximada por una línea recta. La no linealidad es el valor de la desviación máxima de la función de transferencia lineal respecto a la recta de mejor ajuste de la señal de salida. Este error se observa cuando se realizan varios estudios de calibración en un sensor.



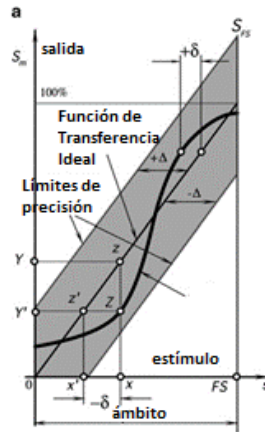


Figura 8. No-linealidad: la recta de mejor ajuste está definida por  $z'$  y  $z$ , sin embargo, la señal se mueve entre un rango determinado. Fuente: adaptado de [1].

## Saturación

Como todo dispositivo físico, los sensores no están exentos de un límite de operación; una vez que se sobrepasa determinado límite de un sensor, su salida no responderá al cambio en el estímulo por lo que se dice que el sensor se satura. También se puede considerar como el punto donde el sensor deja de comportarse de manera lineal y su comportamiento no puede ser predicho por la función de transferencia lineal.

## Repetitividad

El error por repetitividad es causado por la falta de constancia entre los valores dados por el sensor respecto a ciclos de calibración anteriores. La repetitividad es representada por la diferencia máxima entre los valores generados en dichos ciclos de calibración.

## Banda muerta

Es el ámbito de valores de estímulo hacia los cuales el sensor se muestra nula o pocamente receptivo, el valor de salida será típicamente constante y cercano a cero.

## Resolución

Representa el mínimo incremento entre los valores de entrada para el cual el sensor genera una diferencia apreciable en su salida. Vale notar que esta diferencia no siempre será lineal y no se comportará de la misma manera a lo largo de todo el

ámbito de entrada, sin embargo, aun siendo así, el sensor es capaz de generar una salida distinta y apreciable con respecto al valor inicial del estímulo previo al incremento mínimo según su resolución.

### **Propiedades especiales**

Algunas de estas propiedades son importantes en tareas muy específicas o para tipos especializados de sensores. Estas pueden ser: impedancia de salida, formato de salida, excitación, entre otros.

### **Principios físicos de la medición con sensores**

#### **Campos, cargas y potenciales eléctricos**

Existen dos tipos de cargas eléctricas: las positivas y las negativas; el nombre de estas cargas fue asignado por Benjamin Franklin [1], se sabe que las cargas distintas se atraen entre ellas y las cargas iguales se repelen. A partir de este comportamiento se pueden observar algunos efectos eléctricos, tales como el efecto triboeléctrico que responde a la redistribución mecánica de cargas en dos objetos o superficies en contacto: una de ellas pierde electrones y la otra gana electrones, el objeto o superficie que “dona” electrones, termina con una carga positiva y el que los recibe, con una carga negativa.

Cuando dos cargas se encuentran en proximidad la una de la otra, experimentarán una fuerza contraria o a favor de la dirección relativa entre ellas según sean cargas iguales o distintas; independientemente de esto descrito, el volumen de espacio en el que una carga puede ejercer una fuerza sobre la otra es conocido como campo eléctrico [1]. Estos tres conceptos se relacionan entre ellos según la siguiente fórmula:

$$E = \frac{f}{q_0} \quad (1.4)$$

Donde,  $E$  representa el campo eléctrico,  $f$  representa la fuerza entre ambas cargas y  $q_0$  es el valor de una de estas.  $E$  y  $f$  son vectores orientados en la misma dirección y  $q_0$  es una magnitud escalar.

El valor del vector  $E$  puede variar respecto al tiempo y esta variación generará lo que se conoce como flujo. Si se imagina una superficie cerrada  $S$ , la relación entre la carga y el flujo puede ser descrita de la siguiente manera:

$$\varepsilon_0 \oint E ds = q \quad (1.5)$$

$$\oint E ds = \phi_E \quad (1.6)$$

Donde  $\varepsilon_0$  es la constante de permisividad eléctrica y  $\phi_E$ , al igual que  $\oint E ds$  es el flujo eléctrico que atraviesa la superficie S.

Con respecto a la forma en que los materiales interactúan con las cargas eléctricas, se pueden agrupar en tres grandes grupos [1]:

- Aislantes: no existe material perfectamente aislante sin embargo para fines prácticos se puede considerar que estos son los que restringen el movimiento de libre de las cargas eléctricas a través de su extensión.
- Conductores: contrario a los aislantes, estos permiten el movimiento libre de cargas a través de su extensión. No todos los conductores tienen el mismo grado de conductividad.
- Semiconductores: se encuentran en la banda media del espectro y son grandemente utilizados en la manufactura de componentes electrónicos; a estos se les puede agregar trazas de otros materiales para facilitar o dificultar la conducción de acuerdo con el fin requerido.

## Magnetismo

Los elementos que exhiben propiedades magnéticas siempre tienen dos secciones o polos: norte (N) y sur (S) y, de manera homóloga a las cargas eléctricas, los polos similares se repelen y los distintos se atraen. Y si estos elementos se parten en cualquier lugar de su extensión, los fragmentos resultantes también tendrán un polo norte y un polo sur. Este comportamiento sugiere que el magnetismo se asocia a los átomos, a su configuración o a ambos.

Los materiales que reaccionan de forma perceptible al magnetismo se conocen como materiales ferromagnéticos; estos experimentan una fuerza de atracción cuando se encuentran en la proximidad de un material magnético. De forma homóloga al campo eléctrico, el volumen de espacio donde se genera esta fuerza desde un material magnético a uno ferromagnético u otro material magnético se conoce como campo magnético.

Es de conocimiento fundamental [1] que cuando se provoca el movimiento de cargas eléctricas (corriente eléctrica) se produce un campo eléctrico y esto sucede por el alineamiento de los electrones en el material conductor cuando la carga se está moviendo a través de dicho material.

## Ley de Faraday

Michael Faraday descubrió que, así como un campo eléctrico en movimiento puede generar un campo magnético, un campo magnético cambiante puede también generar un movimiento de cargas eléctricas (corriente). Esto se genera como producto de la fuerza eléctrica en las cargas que provoca un movimiento en ellas.

El campo magnético puede relacionarse con la carga y la velocidad de dicha carga con la siguiente ecuación:

$$F_B = q_0VB \quad (1.7)$$

Donde  $F_B$  es la fuerza experimentada por una carga  $q_0$  que se mueve a través de un campo magnético  $B$  con una velocidad  $V$  perpendicular al campo.

Además, para el flujo electromagnético se tiene la ecuación:

$$\oint B ds = \phi_B \quad (1.8)$$

Donde  $\phi_B$  es el flujo electromagnético y  $ds$  es un elemento infinitesimal de una superficie cerrada  $S$ .

## Imanes permanentes

Los imanes permanentes son dispositivos de gran utilidad en el diseño de sistemas de adquisición de datos que estén basados en las propiedades magnéticas de los sensores. Estos sistemas de adquisición de datos basados en sensores magnéticos son frecuentemente utilizados en la detección de movimiento, desplazamiento o velocidad [1]. Para seleccionar un imán para una actividad de medición en particular, se requiere tomar en cuenta las siguientes características [1]:

- Inductancia residual: qué tan fuerte es el imán.
- Fuerza cohesiva: cuál sea su resistencia a fuerzas desmagnetizantes.
- Producción máxima de energía.
- Coeficiente de temperatura: qué tanto cambia  $B$  de acuerdo con cambio de temperatura.

En la actualidad, el imán permanente más fuerte conocido hasta el momento es el imán de neodimio conformado por una aleación cristalina de Neodimio, Hierro y Boro en la forma  $Nd_2Fe_{14}B$

## Efecto de Hall

En el momento de su descubrimiento en 1879 por E. H. Hall, el efecto con el mismo nombre tenía aplicaciones muy limitadas, sin embargo, en la actualidad se utiliza frecuentemente en detección de campos electromagnéticos, detección de partes móviles [1], también pueden ser utilizados en la medición de flujo de fluidos y presión [2].

El efecto hall se relaciona directamente con la ecuación (7) pues este se presenta cuando interactúan cargas eléctricas en movimiento con un campo magnético externo: dicho campo provoca el cambio de movimiento de las cargas debido a la fuerza resultante descrita en el apartado de la ley de Faraday. La interacción entre las cargas y el campo magnético produce un efecto conocido como: *diferencia de potencial transversa de Hall* [1: 104] que puede ser medido entre los lados de un material plano [2: 45] y está descrita por la ecuación:

$$V_H = \frac{IB}{ned} \quad (1.9)$$

Donde  $V_H$  es el voltaje de Hall,  $I$  es la corriente que fluye a través del material,  $n$  es la densidad de carga del material,  $e$  es la carga del electrón igual a  $1,602 \times 10^{-19}C$  y  $d$  es el grosor del material [2]

## Ley de Ampere

La Ley de Ampere permite calcular los flujos magnéticos alrededor de un conductor utilizando la integral de línea del campo magnético a lo largo de dicho conductor [3].

$$\oint \vec{B} \cdot d\vec{l} = \mu_0 I \quad (1.10)$$

Donde  $I$  es la corriente encerrada por la línea que se integra y  $\mu_0$  es una constante y su valor es  $4\pi \times 10^{-7} T \cdot m/A$ .

Esta difiere de la Ley de Gauss para campos magnéticos y eléctricos pues no aplica una integral de superficie cerrada, que en el caso del flujo magnético su suma vectorial sería cero debido a la orientación de las líneas de campo magnético que entran y salen del mismo conductor [3].

## Luz

La medición de la luz resulta de particular interés para la ingeniería y el diseño de sistemas de adquisición de datos debido a la gran cantidad de información que carga y la gran banda del espectro electromagnético que comprende. Se considera como luz a aquellas ondas que comprenden desde los  $0,1\mu m$  hasta los  $100\mu m$ ; donde el extremo con la longitud de onda más pequeña que percibe el ojo humano es conocida como ultravioleta, aquella con la longitud de onda más grande que la

que puede ser vista por el ojo humano es la luz de infrarrojo, y este último bloque se ha dividido de manera arbitraria en: infrarrojos cercanos desde el  $0,9\mu m$  a los  $1,5\mu m$ ; en infrarrojos medios desde los  $1,5\mu m$  a los  $5\mu m$ , e infrarrojos lejanos desde los  $5\mu m$  a los  $100\mu m$  [1]

La luz se puede utilizar para medir distancia, movimiento, temperatura, composición química, entre muchas otras [1]. La luz tiene propiedades duales por lo que puede ser considerada como partícula u onda electromagnética a la misma vez.

Al ser una onda electromagnética tiene propiedades de frecuencia, velocidad y longitud de onda; estas se relacionan de la siguiente manera:

$$f = \frac{c}{\lambda} \quad (1.11)$$

Donde  $f$  es la frecuencia,  $c$  es la velocidad de la luz al vacío y  $\lambda$  es la longitud de onda.

## Reflexión y refracción de la luz

Ambos fenómenos se presentan cuando la luz se enfrenta a un cambio de medio entre medios transparentes: parte de ella es desviada y sigue su curso en el otro medio, a esto se le conoce como refracción y parte de ella es devuelta al medio anterior, a esto se le conoce como reflexión.

Estos dos fenómenos se rigen por dos principios sencillos: el ángulo de reflexión es de igual magnitud respecto a la normal de la superficie sin embargo su orientación es opuesta al ángulo de incidencia; y, el comportamiento refractivo responde a la relación entre los índices de refracción entre ambos medios y estos índices de refracción son la relación entre las velocidades de vacío y la de propagación en el medio por el que se propaga la luz en dicho momento.

Las representaciones matemáticas del índice de refracción y la relación entre ángulo de refracción e índice de refracción son [3]:

$$n = \frac{c}{v} \quad (1.12)$$

$$n_a \sin \theta_a = n_b \sin \theta_b \quad (1.13)$$

Donde  $n$  es el índice de refracción de los medios  $a$  y  $b$  respectivamente,  $v$  es la velocidad de propagación de la luz en determinado medio y  $\theta$  es el ángulo de entrada/salida en los medios  $a$  y  $b$ .

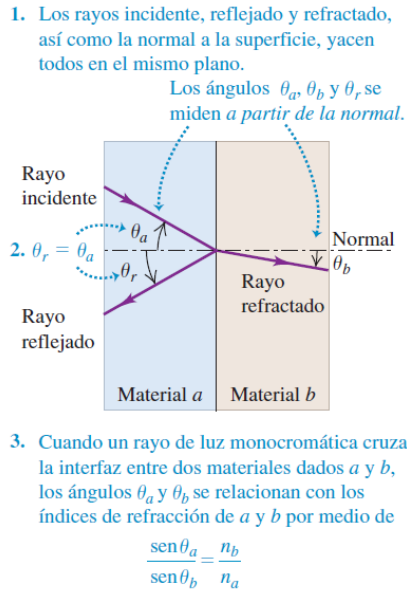


Figura 9. Representación de los comportamientos de reflexión y refracción de la luz. Fuente: adaptado de [3].

## Componentes ópticos de los sensores

Antes de poder medir las propiedades de la luz, esta debe de ser generada; puede ser de manera artificial o de manera natural. Las fuentes naturales de luz pueden ser: el sol, la luna, las estrellas, la bioluminiscencia y cualquier objeto natural que genere calor por sí mismo, en el caso de los infrarrojos; las fuentes artificiales de luz pueden ser: filamentos eléctricos en bombillos, diodos emisores de luz (LED, por sus siglas en inglés), lámparas de descarga de gas, láseres, diodos laser, calentadores, etc. [1]

Una vez que la luz sea generada, debe ser modulada o manipulada para poder ser medida por el sensor que se vaya a utilizar; para estos efectos se utilizan elementos tales como espejos, filtros y/o lentes.

Una vez que se haya logrado la manipulación de la luz emitida, se debe diseñar el sistema de adquisición de la información. Las propiedades que pueden ser tomadas en cuenta para la medición son: intensidad, dirección, polarización, velocidad, tiempo de vuelo, fase, etc. [1]

En la Figura 10., se ejemplifican algunos métodos de medición de la luz.

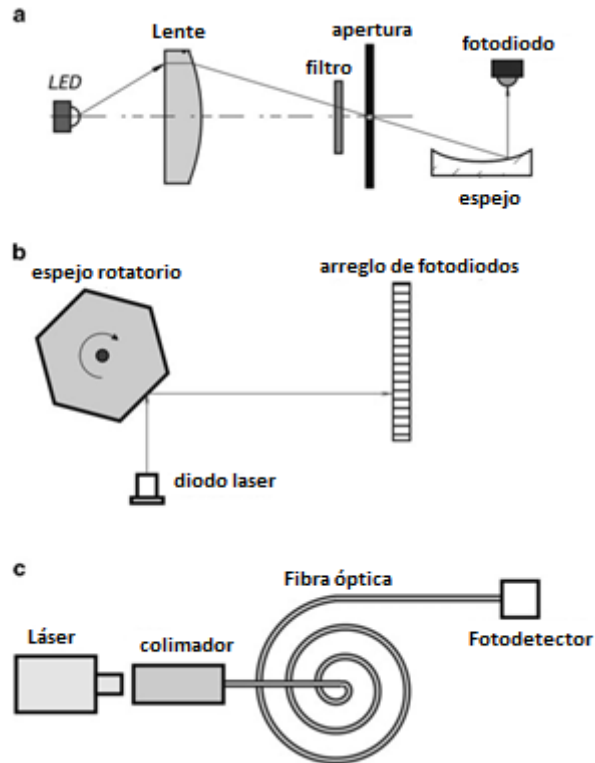


Figura 10. Ejemplos de manipulación de la luz, a) refracción por lentes, b) reflexión en espejo rotativo, c) reflexión y conducción en cable de fibra óptica. Fuente: adaptado de [1].

## Fotometría

La parametrización de las propiedades tanto de la luz como de los sensores que la miden es una actividad fundamental para el diseño de un sistema de adquisición de datos basado en las propiedades ópticas de la luz.

El flujo luminoso es la medida de potencia de una fuente de luz, sin embargo, debido a que el ojo humano tiene distintos niveles de sensibilidad para diferentes longitudes de onda, el flujo luminoso no se puede utilizar para expresar la intensidad o el brillo de una fuente [1].

Como respuesta a este dilema, se conceptualizó la unidad llamada Lumen, que: “está basada en una fuente de radiación estándar con platino fundido en la forma de un cuerpo oscuro y visible a través de una apertura específica dentro de un ángulo sólido de un estereorradián” [1: 154]. Este concepto está definido por la siguiente fórmula como [1]:

$$\omega = \frac{A}{r^2} \quad (1.14)$$



Donde  $A$  es la superficie esférica y  $r$  es el radio de la esfera. Un estereorradián surge cuando  $A = r$ .

La iluminancia está dada por la ecuación:

$$E = \frac{dF}{dA} \quad (1.15)$$

Y representa la relación entre un diferencial de flujo luminoso,  $F$ , y un diferencial de área; típicamente se expresa como Lumen por estereorradián o candela.

La intensidad luminosa se define como:

$$I_L = \frac{dF}{d\omega} \quad (1.16)$$

Y se especifica el flujo sobre un ángulo sólido. Si la intensidad luminosa es constante respecto al ángulo de emisión, se describe con la siguiente ecuación [1]:

$$I_L = \frac{F}{\omega} \quad (1.17)$$

La ecuación (16) ejemplifica lo asegurado en el primer párrafo del apartado de fotometría pues si se varía la longitud de onda  $\omega$  y se mantiene la intensidad luminosa constante, el flujo luminoso varía [1].

## Lentes

Los lentes son sumamente útiles en las aplicaciones por sensores porque permiten redireccionar y enfocar la luz para ser utilizada por aquellos sensores sensibles al espectro electromagnético.

Un ejemplo de ello es el lente planoconvexo en la Ilustración 4.

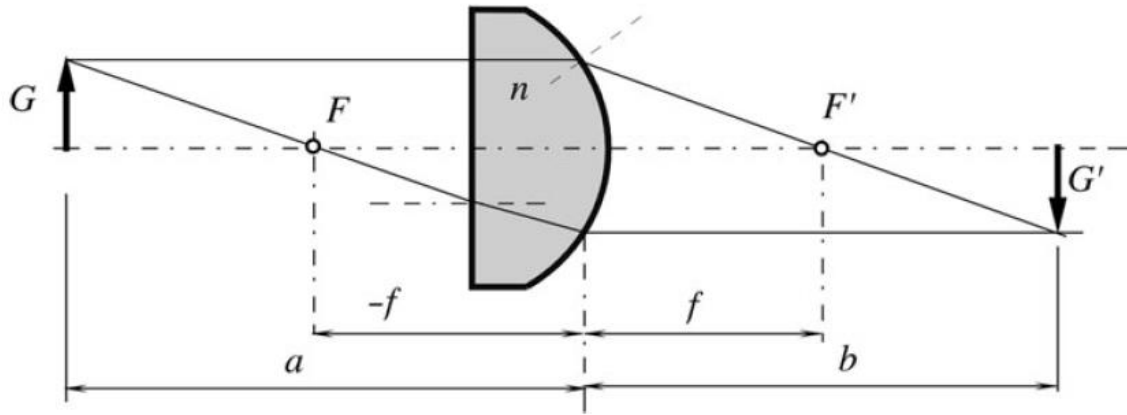


Figura 11. Geometría de un lente planoconvexo. Fuente: adaptado de [1].

Para un lente que tiene un grosor mucho menor al radio de curvatura [1], se tiene la ecuación de puntos focales:

$$\frac{1}{f} = (n - 1) \left( \frac{1}{r_1} + \frac{1}{r_2} \right) \quad (1.18)$$

Donde  $r_1$  y  $r_2$  son los radios de curvatura del lente; en este caso en particular se omite uno de ellos.

De manera similar [1], la distancia  $b$  a la que se encuentra la imagen proyectada  $G'$  puede ser encontrada con la ecuación:

$$\frac{1}{f} = \frac{1}{a} + \frac{1}{b} \quad (1.19)$$

Según la aplicación, la geometría general del lente puede variar, así como el material del cual está hecho.

Cuando a un lente, por ejemplo, se le va a dar un uso donde la focalización o la nitidez de la imagen no es crucial, se puede utilizar un lente de Fresnel [1]: estos están contruidos a partir de ranuras concéntricas que dirigen la luz a un punto focal en común.

Dichos lentes pueden estar fabricados de materiales tales como el vidrio y el acrílico, para luz visible e infrarrojos cercanos, y el polietileno, para infrarrojos medios y lejanos [1].

## Fototransistores

Un fototransistor convierte los fotones entrantes en pares electrón-agujero. Los pares electrón-agujero se forman cuando un electrón salta de un átomo a otro, dejando un agujero de carga. Dicho salto provoca un movimiento de cargas o corriente [1].

El contraste entre un fototransistor y otros elementos fotosensibles como los fotodiodos es que los primeros pueden además generar una ganancia de corriente por lo que se obtiene una sensibilidad mayor [1].

Si el fototransistor está conectado a un circuito con una fuente de poder, una corriente foto inducida fluye a través de la maya y esta corriente es amplificada por el transistor como sucede típicamente.

Es normal encontrar sensores encapsulados, conformados por un diodo emisor de luz infrarroja y un fototransistor.

## Mecánica de los Materiales

### Reacciones de Soporte

A las fuerzas superficiales que se desarrollan en los puntos de contacto entre dos cuerpos se les llama *reacciones*. Si un soporte previene la traslación en una dirección dada, una reacción de soporte debe ser observada en esa dirección; de igual manera, si un soporte previene una rotación, se debe de observar un par de torsión en esa dirección de giro [4].

### Ecuaciones de Equilibrio

Para que un cuerpo se encuentre en equilibrio, ósea, detenido o con movimiento uniforme no acelerado, sus ecuaciones de equilibrio deben resultar en un valor neto de cero, tanto en su balance de fuerzas como en su balance de momentos.

Si se está realizando un estudio en tres dimensiones se deben tomar en cuenta todos los componentes de fuerzas y momento por separado.

$$\sum F_x = 0 \quad \sum F_y = 0 \quad \sum F_z = 0 \quad (1.20)$$

$$\sum M_x = 0 \quad \sum M_y = 0 \quad \sum M_z = 0 \quad (1.21)$$

### Diagrama de fuerzas y momentos

La mejor manera de observar y resolver tales ecuaciones es con una representación ilustrativa de las fuerzas y momentos que actúan en el cuerpo de estudio.

## Reacciones Internas Resultantes

El análisis estático previamente descrito permite la obtención de las reacciones de soporte. A partir de estas reacciones de soporte se pueden estimar las fuerzas internas resultantes y los momentos internos resultantes en cualquier punto de un cuerpo homogéneo.

Algunos conceptos importantes para tener en cuenta en el análisis estático de elementos y resistencia de materiales son:

- Fuerza Normal ( $N$ ): Es la fuerza que actúa normal al área. Se desarrolla cuando las fuerzas externas producen tensión o compresión.
- Fuerza Cortante ( $V$ ): Esta fuerza actúa paralela al plano y se desarrolla cuando las cargas externas hacen que los segmentos del cuerpo se deslicen uno sobre el otro.
- Momento de torsión o torque ( $T$ ): Este efecto se observa cuando las cargas externas tienden a girar un extremo del objeto con respecto al otro a través de un eje perpendicular al área.
- Momento flector ( $M$ ): Este efecto se da cuando las cargas externas tienden a doblar el objeto a la largo de un eje que pertenece al mismo plano del área.

## Análisis Estático

El análisis estático completo para obtener todas las posibles cargas dentro de un cuerpo es el siguiente [4]:

- Se debe decidir cuál segmento del cuerpo debe ser considerado y hacer el estudio de las reacciones de soporte de acuerdo con los criterios expuestos antes.
- Dibujar el diagrama de cuerpo libre con todas las fuerzas y momentos externos ubicados en el lugar exacto (o más aproximado posible) de su aplicación.
- Es importante que en el diagrama de cuerpo libre se dibujen también las reacciones resultantes  $N$ ,  $V$ ,  $M$  y  $T$ . Estas reacciones resultantes deben estar ubicadas en el centroide del miembro.
- Si el miembro es sujeto a un sistema coplanar de fuerzas, sólo  $N$ ,  $V$  y  $M$  actúan en el centroide.
- Es también necesario establecer el sistema de ejes coordenados con su origen en el centroide (o a lo largo del eje central longitudinal, como en el caso de las vigas) y representar las reacciones para cada uno de los ejes.
- Si la solución de las ecuaciones de equilibrio resultara en una respuesta negativa, quiere decir que la resultante está en sentido contrario a como se representó en la ilustración.

## Esfuerzo

Es la distribución de la intensidad de la fuerza que atraviesa determinado plano en el material de un cuerpo. Cuando se hace un estudio de esfuerzo en un miembro, este debe considerarse como continuo, es decir, sin vacíos, y cohesivo, o sea, que todas sus porciones están conectadas la una con la otra.

### Esfuerzo Normal

La intensidad de la fuerza que actúa normal al plano del área se define como esfuerzo normal ( $\sigma$ ). Si esta fuerza “jala” el material de manera que tiende a separarse, el esfuerzo se conoce como de tensión. Si la fuerza tiende a “juntar” el material, el esfuerzo se conoce como de compresión.

El esfuerzo normal se calcula con la fórmula:

$$\sigma = \frac{P}{A} \quad (1.22)$$

### Esfuerzo Cortante

Es la intensidad de la fuerza que actúa tangente al plano del área y se representa con la letra  $\tau$ . Este tipo de esfuerzo tiende a hacer que el material *fluya* sobre sí mismo.

El esfuerzo cortante se calcula con la siguiente fórmula:

$$\tau_{prom} = \frac{V}{A} \quad (1.23)$$

Donde  $V$  es la fuerza interna resultante que actúa tangente al plano del área  $A$ .

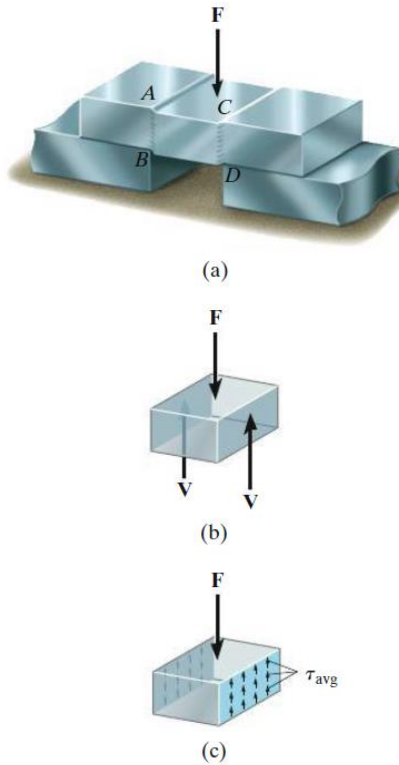


Figura 12. Representación visual de la acción que provoca un esfuerzo cortante.  
Fuente: adaptado de [4]

### Diagrama de esfuerzo – deformación.

Extensivas pruebas extensiométricas a lo largo de la historia de la ingeniería de los materiales han permitido a los ingenieros relacionar la deformación de un elemento calibrado con la deformación física que dicho elemento de determinado material sufre cuando se aplica una carga controlada. La curva resultante de estas pruebas que relaciona ambos fenómenos físicos se llama diagrama de esfuerzo – deformación. Este diagrama permite a los diseñadores delimitar las cargas a las que estará sometido el elemento que se está diseñando o estudiando.

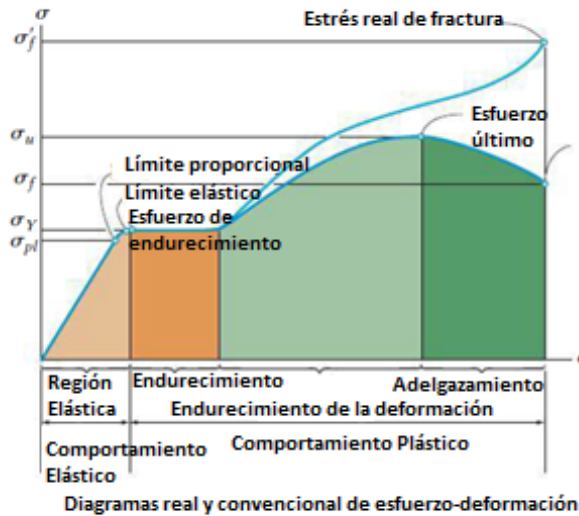


Figura 13. Diagrama de esfuerzo-deformación de un material dúctil (sin escala).  
Fuente: adaptado de [4]

Las regiones observadas en el diagrama son las siguientes:

- Región elástica o de comportamiento elástico: en esta región el comportamiento es completamente lineal por lo que la deformación estará directamente relacionada con el esfuerzo aplicado. Esta región es de particular interés para los ingenieros pues toda deformación que se experimente por el cuerpo dentro de estos límites es completamente reversible.
- Cedencia: durante esta etapa, el material comienza un proceso de deformación no recuperable en el que el incremento de extensión del material no es lineal al cambio de esfuerzo, y en realidad este último es constante a lo largo de esta región y es conocido como esfuerzo de cedencia. Este es el que se utiliza para calcular el factor de seguridad discutido anteriormente.
- Endurecimiento: una vez sobrepasado el esfuerzo de cedencia, el esfuerzo necesario para seguir deformando el material incrementa rápidamente hasta llegar al esfuerzo último ( $\sigma_{ult}$ ). Durante estas dos últimas etapas, el comportamiento del material se conoce como plástico y se refiere a la tendencia de mantener la deformación alcanzada de manera permanente, aunque el esfuerzo sea retirado.
- Adelgazamiento: al terminar la etapa de endurecimiento, el material comenzará a adelgazarse significativamente en un punto determinado hasta alcanzar la falla catastrófica. En la ilustración 10, se observan dos curvas distintas y esto se debe a que la curva que llega más alto se ha calculado con la nueva menor área que el adelgazamiento paulatino va provocando; y la curva que se observa más abajo se ha calculado utilizando el área original del elemento.

Los ingenieros trabajan, en la mayoría de los casos, dentro del área de comportamiento elástico debido principalmente a que las aplicaciones están diseñadas cuidadosamente para medidas de longitud, área y volumen predefinidas y a que, una vez alcanzados los límites elásticos del material, al seguir aumentando la carga, quedan esfuerzos residuales en este a pesar de que la carga pueda haberse liberado.

## Módulo de Young

Es la pendiente de la recta que sigue la curva de esfuerzo – deformación dentro de la región de comportamiento plástico. También es conocido como módulo de elasticidad (E).

$$E = \frac{\sigma}{\epsilon} \quad (1.24)$$

Donde  $\epsilon$  es la deformación unitaria. Debido a que la deformación unitaria no tiene unidades, las unidades del módulo de Young serán las mismas que las de la presión y el esfuerzo según el sistema utilizado.

El conocimiento de dicho módulo permite a los ingenieros calcular con relativa precisión la deformación que se espera que un elemento tenga bajo las cargas de presión permitidas por su factor de seguridad.

## Diagramas de Momento y Fuerza Cortante

Por lo general, a los elementos mecánicos donde su longitud es considerablemente mayor a su grosor y están suspendidos sobre uno o más soportes, son llamados vigas. El análisis de fuerzas y momentos de dichos elementos es de gran importancia pues permite una representación visual y numérica de las condiciones de esfuerzo a las que estará sometido y, por consiguiente, generará oportunidades de mejora del diseño o prevención de falla por carga estática.

Algunas consideraciones generales para los diagramas de momento y fuerza cortante [4]:

- Se debe de escoger el origen y la dirección positiva para los signos. Por lo general se utiliza el sentido de izquierda a derecha.
- Si la viga está sujeta a fuerzas transversales y momentos externos, se debe realizar el análisis para la sección creada entre cada uno de estos agentes externos.
- La convención de signos se representa en la Figura 14. Una fuerza cortante que tiende a provocar un movimiento contrario a las agujas del reloj se



considera como positivo; y un momento que tiende a flexionar la viga con su vértice hacia arriba se considera como positivo.

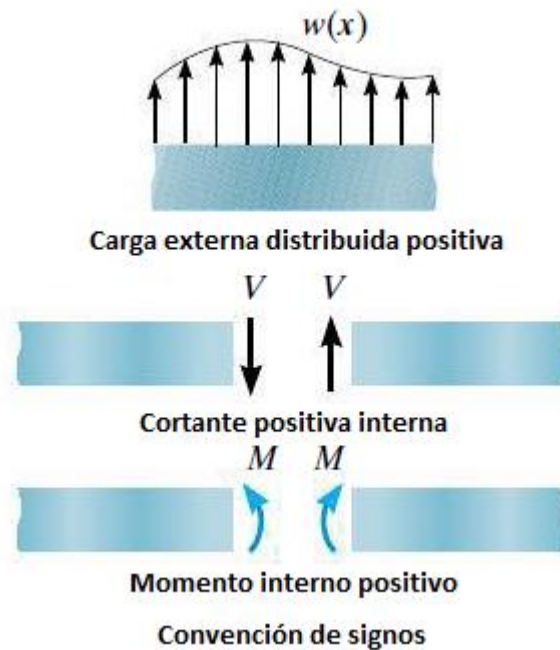


Figura 14. Convención de signos para los diagramas de cortante y momento.  
Fuente: adaptado de [4]

Para construir un diagrama de momento-cortante se llevan a cabo los siguientes pasos [4]:

1. Una vez decidido el origen, se hace el recorrido de la barra hacia la dirección opuesta.
2. Cada reacción de soporte se registra según su dirección: si es hacia arriba, como positiva, pues provoca una fuerza interna cortante positiva para mantener la estática del cuerpo según el diagrama de cuerpo libre; y si es hacia abajo, como negativa.
3. El momento flector se calcula en función de la distancia  $x$  y la fuerza cortante a través de una sumatoria de momentos en dicha posición.
4. Si los valores de momento y cortante son positivos, se incluyen en la parte superior del diagrama, si son negativos, en la parte inferior.

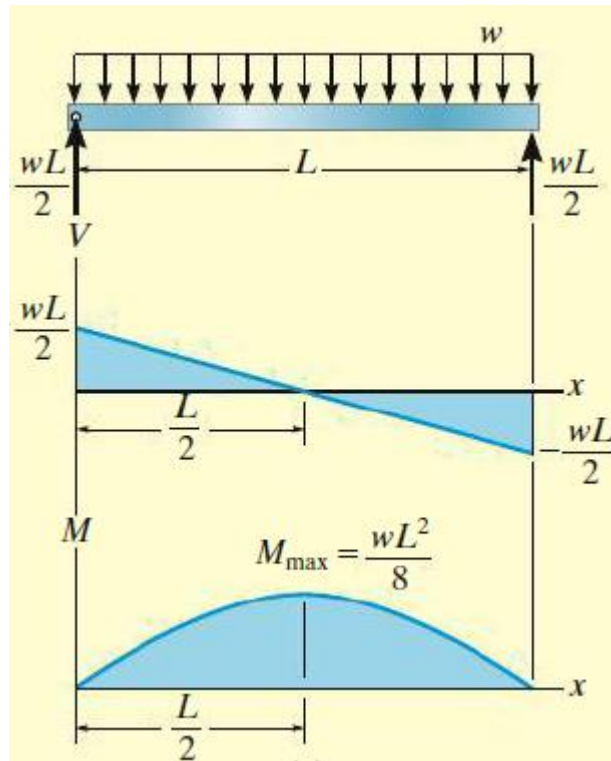


Figura 15. Diagramas de Cortante y Momento Flector de una barra simplemente suspendida a la que se le aplica una distribución de carga constante  $w$ . Fuente: adaptado de [4]

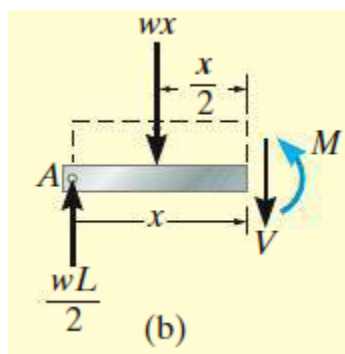


Figura 16. Análisis estático de la viga en la posición  $x$  para la construcción de los diagramas en la Figura 20. Fuente: adaptado de [4]





Forma de la viga	Fórmula	Forma de la viga	Fórmula
 Rectangular	$\tau_{\text{máx}} = \frac{3V}{2A}$	 Circular hueca con pared delgada	$\tau_{\text{máx}} = \frac{2V}{A}$
 Circular	$\tau_{\text{máx}} = \frac{4V}{3A}$	 Viga I estructural (con pared delgada)	$\tau_{\text{máx}} = \frac{V}{A_{\text{Alma}}}$

Figura 17. Esfuerzos cortantes máximos de acuerdo con perfil de la viga. Fuente: adaptado de [5].

### Esfuerzo Normal en Vigas

Cuando una viga simétrica está expuesta a un momento flector interno, en ella se generan esfuerzos normales distribuidos en su área transversal debido a la deformación que sufre como producto del momento.

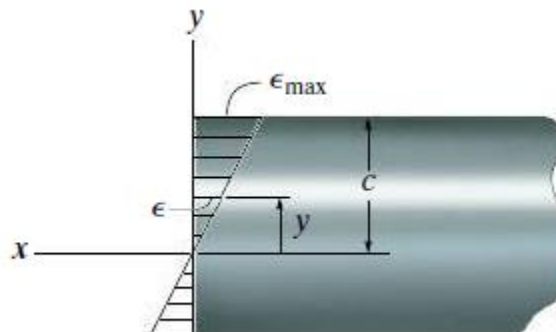


Figura 18. Representación visual de la deformación normal a lo largo de la sección transversal. Fuente: adaptado de [4]

En la Figura 18, se puede observar la deformación que sufre el elemento debido al momento flector.

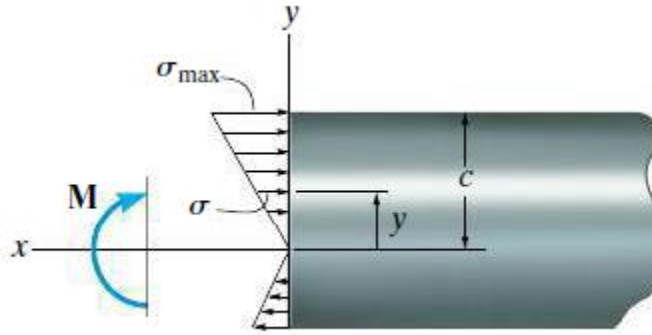


Figura 19. Representación visual del esfuerzo normal a lo largo de la sección transversal. Fuente: adaptado de [4]

La convención de signos es de suma importancia pues un momento flector positivo provoca un esfuerzo normal negativo, o de compresión, por encima de la superficie neutra; y un esfuerzo normal positivo, o de tensión, por debajo de la superficie neutra.

El valor de los esfuerzos de compresión y tensión máximos en una viga simétrica se calcula con la siguiente fórmula [4]:

$$\sigma_{max} = \pm \frac{Mc}{I} \quad (1.25)$$

Donde  $M$  es el momento interno resultante,  $I$  es el momento de inercia de la sección transversal y  $c$  es la distancia perpendicular desde el eje de simetría de la sección transversal.

## Deflexión de Vigas

El trayecto que describe la línea neutra de una viga cuando se somete a momentos flectores es conocida como *curva elástica*.

La construcción de dicha curva es sencilla sin embargo es importante tomar en cuenta la forma en que los distintos soportes restringen la traslación o rotación de la línea neutra (asumiendo que son soportes ideales) y algunas reglas necesarias [1]:

- Cuando hay soportes que restringen una fuerza también restringen la traslación de la línea neutra y los soportes que restringen un momento también restringen la rotación o el curvado de la línea neutra.
- Un momento interno positivo tiende a flexionar la viga con una concavidad hacia abajo y un momento interno negativo tiende a flexionar la viga con una concavidad hacia arriba.

- Cuando el momento interno cambia de signo, se observará un punto de inflexión.

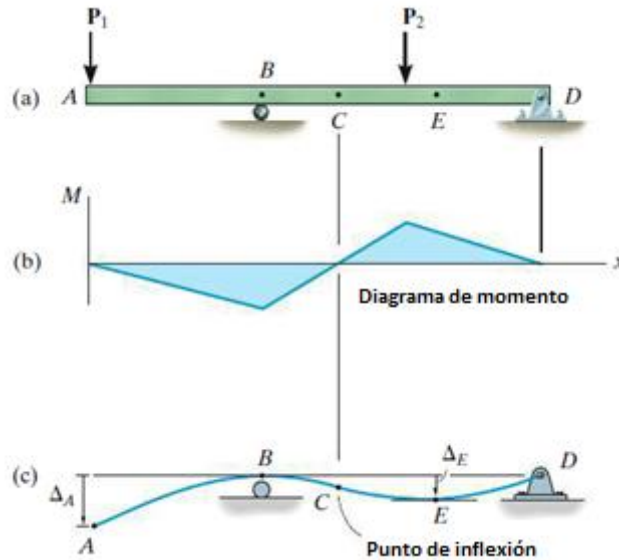


Figura 20. Se puede observar un punto de inflexión en la coordenada C. Fuente: adaptado de [1].

Para un material homogéneo y lineal la ecuación que relaciona el radio de curvatura  $\rho$  con el momento interno  $M$  en una viga que presenta un momento de inercia  $I$  y tiene un módulo de elasticidad  $E$ , es [1]:

$$\frac{1}{\rho} = \frac{M}{EI} \quad (1.26)$$

Para el cálculo completo de la *curva elástica*, se utilizan las siguientes ecuaciones:

$$EI \frac{d^4 v}{dx^4} = w(x) \quad (1.27)$$

$$EI \frac{d^3 v}{dx^3} = V(x) \quad (1.28)$$

$$EI \frac{d^2 v}{dx^2} = M(x) \quad (1.29)$$

Donde  $w(x)$  es la función de la carga distribuida en la viga respecto  $x$ ,  $V(x)$  es la fuerza cortante interna respecto de  $x$  y  $M(x)$  es el momento flector interno respecto de  $x$ .

## Diseño Mecánico

### Factor de Seguridad

El diseño o análisis de un elemento mecánico real requiere de la definición de las cantidades permisibles de esfuerzos dentro del material de acuerdo con la resistencia y configuración geométrica de dichos material y elemento. La relación entre la cantidad permissible de carga y la condición catastrófica del sistema se conoce como factor de seguridad o **F.S.**

$$F.S. = \frac{\sigma_{Falla}}{\sigma_{Permisible}} \quad (1.30)$$

Para el esfuerzo normal y:

$$F.S. = \frac{\tau_{Falla}}{\tau_{Permisible}} \quad (1.31)$$

Para el esfuerzo cortante.

En materia de diseño de ingeniería esta relación dependerá de la aplicación y la naturaleza de los esfuerzos que sufra el elemento dicho (fatiga, impacto, estático, dinámico); por lo general se utiliza un F.S. de entre 1,2 y 1,5. En ningún caso el factor de seguridad puede ser igual o menor a 1 pues esto significaría una falla inminente o falla catastrófica respectivamente [5].

### Selección del Material

Una serie de pasos aconsejados por [5] para la selección del material son:

1. Enlistar las propiedades más importantes que el material debe de ofrecer según su función en el sistema diseñado, que puede incluir: resistencia, rigidez, dureza, costo, entre otros. Posteriormente se puede asignar una medida de ponderación para estas propiedades.
2. Para cada una de estas propiedades definidas anteriormente, se enlistarán los materiales disponibles ordenados de acuerdo con el mejor material para cada propiedad.
3. De la matriz construida en los pasos anteriores se extraen los materiales que estén más arriba en la lista y se someten a una comparación más detallada.

Es importante señalar que, aunque esta es una práctica saludable de diseño, no todos los pasos podrán ser cumplidos en todas las ocasiones pues frecuentemente la selección del material puede estar sujeta a condiciones logísticas como la disponibilidad y la proveeduría, entre otros.

## Círculo de Mohr

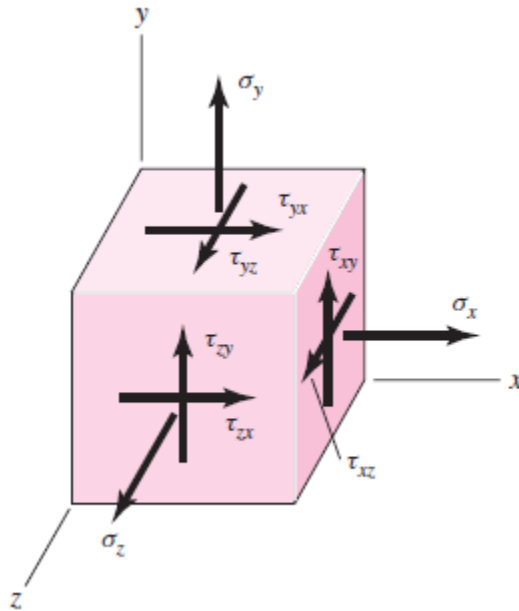


Figura 21. Esfuerzos tridimensionales para un punto del material. Fuente: adaptado de [5].

Si se representa un punto del material como en la Figura 21., y se igualan los esfuerzos cortantes  $\tau_{yx} = \tau_{xy}, \tau_{zy} = \tau_{yz}$  y  $\tau_{zx} = \tau_{xz}$  de manera tal que el equilibrio del cuerpo se mantenga, la representación anterior se puede simplificar como:

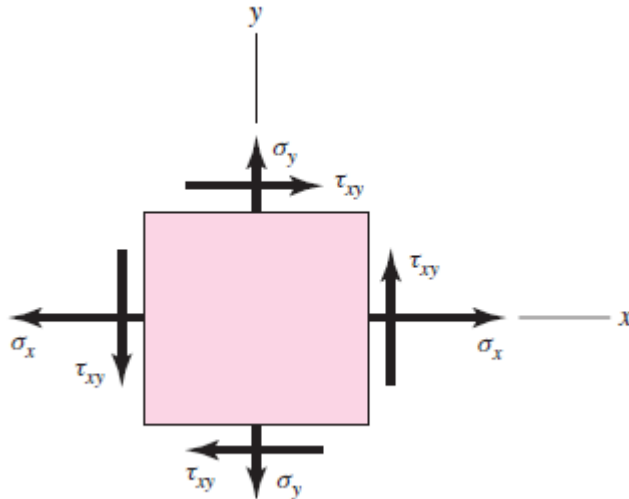


Figura 22. Representación simplificada de los esfuerzos en un punto del material. Fuente: tomado de [5].

Si el elemento plano de la Figura 22., se cortara “por un plano oblicuo con una normal  $n$  a un ángulo arbitrario  $\phi$  en sentido inverso al de las manecillas del reloj con respecto al eje  $x$ ” [5: 76] como se observa en la siguiente figura:

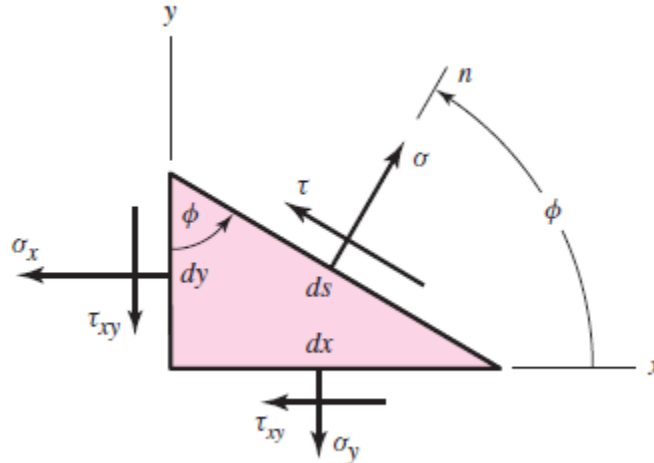


Figura 23. Corte oblicuo realizado al elemento de la Figura 22. Fuente: adaptado de [5].

Al hacer un análisis estático de las fuerzas producidas por los esfuerzos, se pueden obtener las *ecuaciones de transformación de esfuerzo plano* [5]:

$$\sigma = \frac{\sigma_x + \sigma_y}{2} + \frac{\sigma_x - \sigma_y}{2} \cos(2\phi) + \tau_{xy} \sin(2\phi) \quad (1.32)$$

$$\tau = -\frac{\sigma_x - \sigma_y}{2} \sin(2\phi) + \tau_{xy} \cos(2\phi) \quad (1.33)$$

Al aplicar la derivada de la primera ecuación con respecto al ángulo  $\phi$  e igualar a 0 se obtiene [5]:

$$\tan 2\phi_p = \frac{2\tau_{xy}}{\sigma_x - \sigma_y} \quad (1.34)$$

De la ecuación (1.34) se puede observar que se toman en cuenta dos valores para el ángulo  $2\phi_p$ , de estos, uno define el esfuerzo normal máximo y el otro define el esfuerzo normal mínimo; estos esfuerzos con conocidos como *esfuerzos principales* y sus direcciones como *direcciones principales* que están a un ángulo de  $90^\circ$  la una de la otra [5].

Si la ecuación (1.33) se iguala a 0 y se compara con (1.34) se puede obtener que  $\tau = 0$  por lo que se puede asegurar que “las superficies que contienen esfuerzos principales tienen esfuerzos cortantes iguales a cero.” [5: 77]

Además, si la ecuación se deriva respecto al ángulo  $\phi$  y se iguala a 0, se obtiene [5]:



$$\tan 2\phi_s = -\frac{\sigma_x - \sigma_y}{2\tau_{xy}} \quad (1.35)$$

La ecuación anterior define los valores de  $2\phi_s$  donde el esfuerzo cortante  $\tau$  alcanza su valor máximo, el ángulo entre las superficies donde el esfuerzo cortante es extremo es igual a  $90^\circ$  [5].

Si (1.35) se sustituye en (1.32) según las reglas trigonométricas, tenemos la ecuación reducida:

$$\sigma = \frac{\sigma_x + \sigma_y}{2} \quad (1.36)$$

Por lo tanto, se concluye que las superficies donde el esfuerzo cortante es extremo, se observan esfuerzos normales iguales a  $\frac{\sigma_x + \sigma_y}{2}$ .

Adicionalmente se tiene que  $\tan 2\phi_s$  es el recíproco negativo a  $\tan 2\phi_p$  entonces se sabe que el ángulo entre las superficies en las que se observan los esfuerzos principales cortantes y normales es de  $\pm 45^\circ$ .

Posteriormente al sustituir el ángulo de (1.34) en (1.34) se genera:

$$\sigma_1, \sigma_2 = \frac{\sigma_x + \sigma_y}{2} \pm \sqrt{\left(\frac{\sigma_x - \sigma_y}{2}\right)^2 + \tau_{xy}^2} \quad (1.37)$$

Y para el esfuerzo cortante:

$$\tau_1, \tau_2 = \pm \sqrt{\left(\frac{\sigma_x - \sigma_y}{2}\right)^2 + \tau_{xy}^2} \quad (1.38)$$

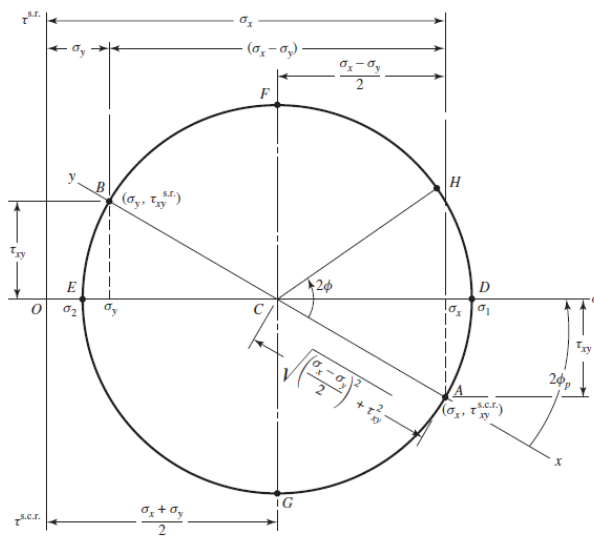
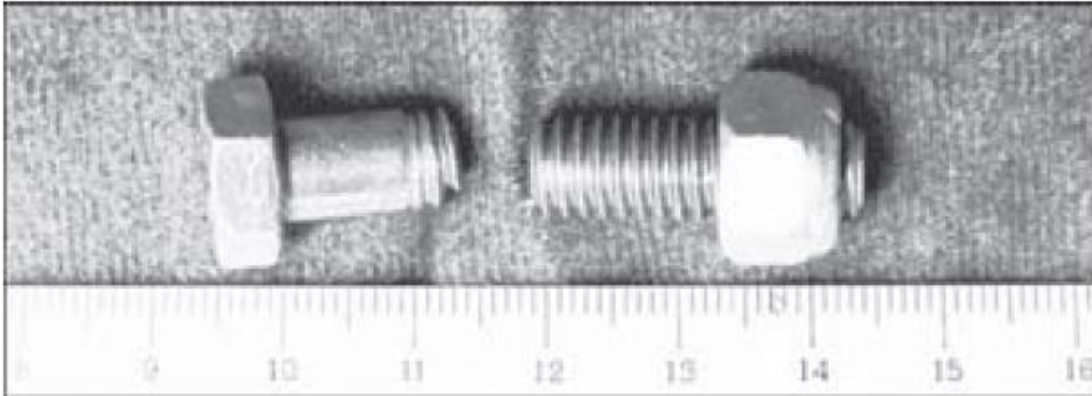


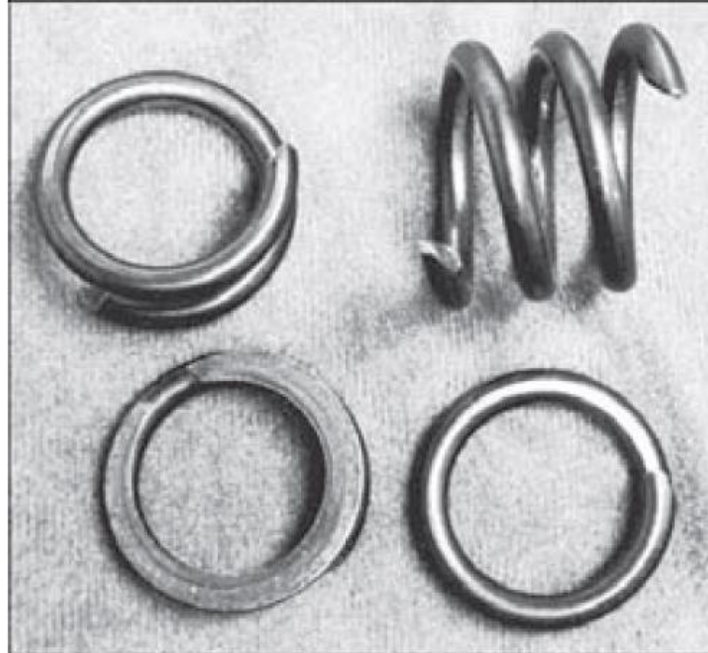
Figura 24. Representación gráfica del círculo de Mohr. Fuente: tomado de [5].

## Falla estática

La definición de falla según [5: 206] es “que una parte se ha separado en dos o más piezas; se ha distorsionado permanentemente, arruinando de esta manera su geometría; se ha degradado su confiabilidad; o se ha comprometido su función por cualquier razón.” Para que una falla sea estática, debe de haberse dado en condiciones de carga y esfuerzos completamente constantes a lo largo del tiempo.



*Figura 25. Ejemplo de falla estática por esfuerzo normal en un perno. Fuente: adaptado de [5]*



*Figura 26. Falla estática por cortante en un resorte. Fuente: adaptado de [5].*

## **Materiales Dúctiles y Frágiles**

Para identificar un material como dúctil es importante poder observar una resistencia a la fluencia identificable y una deformación unitaria menor al 0,05; a menudo las resistencias de compresión y tensión son iguales en estos materiales [5].

Los materiales frágiles no presentan resistencia a la fluencia identificable y su deformación unitaria es menor al 0,05; típicamente se clasifican por sus resistencias últimas a la tensión y a la compresión [5].

Las teorías de falla más comunes son [5]:

- Para materiales dúctiles:
  - Esfuerzo cortante máximo (ECM).
  - Energía de distorsión (ED).
  - Mohr-Coulomb dúctil (CMD).
- Para materiales frágiles:
  - Esfuerzo normal máximo (ENM).
  - Mohr-Coulomb frágil (CMF).
  - Mohr modificada (MM).

Los materiales que se utilizan típicamente en proyectos similares a los del presente desarrollo son dúctiles, y entre las teorías de falla para materiales dúctiles, la teoría del ECM es la teoría más conservadora de las tres [5] por lo que dicha teoría es la que resulta relevante para este Marco Teórico.

## Teoría del Esfuerzo Cortante Máximo

“La teoría del esfuerzo cortante máximo estipula que la fluencia comienza cuando el esfuerzo cortante máximo de cualquier elemento iguala al esfuerzo cortante máximo en una pieza de ensayo a tensión del mismo material cuando esa pieza comienza a fluir.” [5: 211]

Si se toma en cuenta que las líneas de fractura se observan a aproximadamente  $45^\circ$  y esto es consistente con la teoría del Circulo de Mohr que propone que el cortante máximo sucede a  $45^\circ$  del esfuerzo normal máximo.

La teoría del ECM sintetiza una relación entre el cortante y la resistencia a la fluencia  $S_y$  con la ecuación [5]:

$$\tau_{m\acute{a}x} = \frac{\sigma_1 - \sigma_3}{2} \geq \frac{S_y}{2} \quad (1.38)$$

Ó

$$\sigma_1 - \sigma_3 \geq S_y \quad (1.39)$$

Esto implica que la resistencia a la fluencia,  $S_{sy}$ , en cortante está dada por [5]:

$$S_{sy} = 0,5S_y \quad (1.40)$$

Si se toma en cuenta el factor de seguridad, la ecuación se puede escribir de la siguiente manera [5]:

$$\tau_{m\acute{a}x} = \frac{S_y}{2n} \quad (1.41)$$

Ó

$$\sigma_1 - \sigma_3 \geq \frac{S_y}{n} \quad (1.42)$$

Comúnmente, en los problemas de esfuerzo plano, uno de los esfuerzos principales es igual a 0 y los otros dos esfuerzos  $\sigma_A$  y  $\sigma_B$  pueden ser calculados con la ecuación (1.37). Para el análisis de la teoría del ECM, se consideran tres casos posibles [5]:

1.  $\sigma_A \geq \sigma_B \geq 0$ : en este caso la ecuación (1.39) se reduce a:

$$\sigma_A \geq S_y \quad (1.43)$$

2.  $\sigma_A \geq 0 \geq \sigma_B$ : la ecuación (1.39) se convierte en:

$$\sigma_A - \sigma_B \geq S_y \quad (1.44)$$

3.  $0 \geq \sigma_A \geq \sigma_B$ : la ecuación (1.39) se reformula como:

$$\sigma_B \leq S_y \quad (1.45)$$

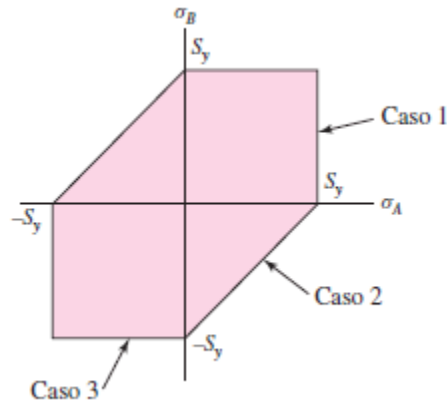


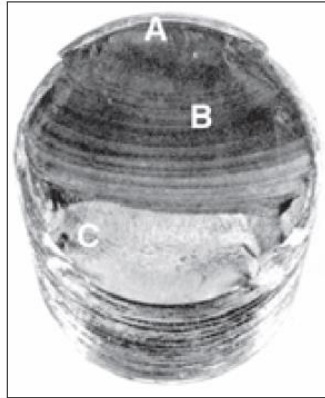
Figura 27. Representación de la zona segura de operación para un material según la teoría del Esfuerzo cortante máximo. Fuente: adaptado de [5].

## Falla por Fatiga

La falla por fatiga se da cuando un material falla, aunque el esfuerzo esté considerablemente por debajo del límite de fluencia sin embargo se repite una gran cantidad de veces [5].

La falla por fatiga se provoca por lo general en tres etapas [5]:

- Etapa I: se empiezan a formar microgrietas debido a la deformación plástica cíclica del material.
- Etapa II: las microgrietas se convierten en macrogrietas, las cuales forman superficies paralelas que se rozan entre sí.
- Etapa III: cuando las macrogrietas son demasiado grandes, el material restante no puede soportar las cargas y falla de manera catastrófica.



*Figura 28. Inicio de las microgrietas en A, formación de macrogrietas en B y falla catastrófica en C en un perno. Fuente: adaptado de [5]*

Cuando en un ensayo o implementación, la fatiga se presenta en el rango desde los 1 a 1000 ciclos, se considera como *fatiga de bajos ciclos*, para un número de 1000 o más ciclos, se considera como *fatiga de altos ciclos* [5]. En los aceros existe una tercera zona conocida como zona de vida infinita y está por debajo del límite de fatiga; una vez sobrepasada este límite inferior, se considera que el elemento no fallará nunca por fatiga.

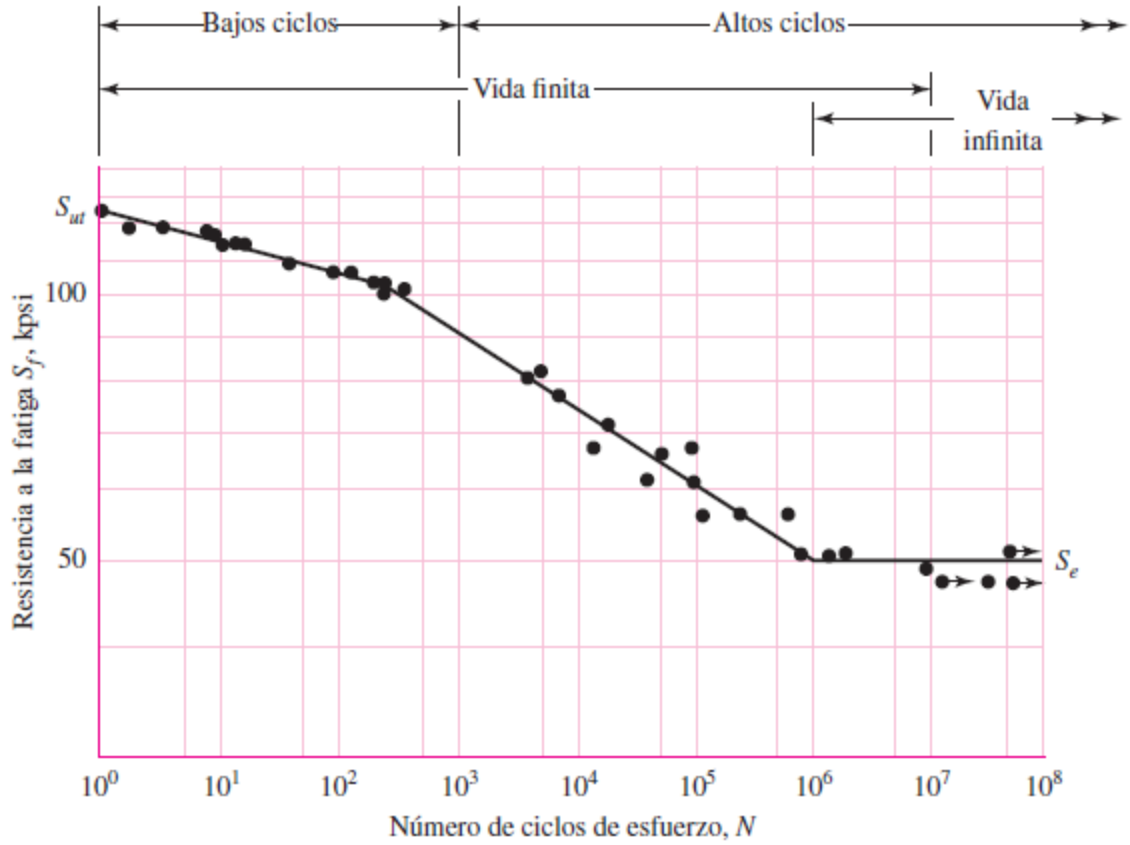


Figura 29. Ubicación de las distintas zonas para el análisis de fatiga en un acero.  
Fuente: adaptado de [5].

## Capítulo 3: Metodología

Para el planteo de la metodología que se seguirá en el presente proyecto, se deben observar directamente los objetivos planteados que permitirán la solución final del problema en cuestión.

Según [6], los problemas de ingeniería se pueden categorizar en abiertos o cerrados. Cuando un problema es *abierto* puede tener varias soluciones y cuando es *cerrado* solo podrá tener una y esta se encuentra a través de cálculos.

El objetivo general del presente proyecto tiene un entregable muy claro: “un sistema mecatrónico para el monitoreo de las dimensiones del ancho y la velocidad de extrusión...” Para alcanzar este entregable se han presentado una serie de objetivos específicos que permitirán diseñar dicho sistema mecatrónico.

### **Análisis del espacio disponible en la máquina para los montajes del sistema de adquisición de ancho y el sistema de adquisición de velocidad acorde al proceso productivo de la máquina**

Debido a que la extrusora E16 es un sistema discreto que no se va a modificar o expandir de ninguna manera, por lo tanto, sus dimensiones físicas y su proceso productivo quedarán iguales.

Dicho lo anterior, primero se analizará el proceso productivo y los pasos que sigue la materia prima dentro de la máquina para convertirse en el producto final: este paso permitirá aislar las principales etapas productivas y la ubicación física de los elementos que participan en estas etapas.

Una vez definida la finalidad de cada etapa y su ubicación física en la extrusora, se podrán observar las limitaciones y las libertades de trabajar dentro de ella y si lo más beneficioso para la solución adecuada del proyecto es diseñar el o los sistemas para que funcionen antes, dentro o después de la etapa que se está considerando.

La selección de la etapa o las etapas en las que se adjuntarán el o los sistemas de medición deberá ser consecuente con la respuesta a las siguientes preguntas:

- ¿Se cuenta con el espacio físico para implementar el/los sistemas de medición? La respuesta a esta pregunta tiene que ser afirmativa pues de lo contrario será imposible una posible implementación del sistema o alguna de las etapas de medición.
- ¿Modifica de alguna manera negativa la producción y/o la calidad del producto la ubicación de una de las etapas de medición o del sistema completo? La respuesta a esta pregunta debe ser negativa pues los estándares de calidad no pueden ser degradados y, además, la producción debe mantenerse sin ser afectada de manera negativa.



- ¿Dificulta el trabajo de los operadores, ingenieros, supervisores o alguna otra persona de interés, la implementación del sistema de medición o alguna de sus etapas en esta zona en particular? La respuesta a esta pregunta debe ser negativa pues este sistema de medición no puede influir de manera negativa en ninguna de las etapas del proceso productivo o en las labores del personal de la planta Polymer.
- ¿La información que puede ser medida en esta etapa es relevante para el control de la producción? La respuesta a esta pregunta debe de ser positiva pues de otra manera se estarían desperdiciando recursos y se estaría introduciendo un sesgo en el sistema de adquisición de datos.
- ¿La información que será medida en esta etapa puede ser adquirida de manera confiable, constante, precisa y regular? Esta pregunta sigue un orden lógico a la anterior y su respuesta debería de ser positiva pues de nada serviría la implementación de uno o varios sistemas de medición si la recopilación de información no se mantendrá constante o confiable durante el transcurso de la jornada de producción de la máquina y podrían necesitarse más dispositivos de medición redundantes en otras etapas para complementar la pérdida de información.

### Fórmula de producción

Para el cálculo de la producción en términos de masa por unidad de tiempo, se necesita aplicar una fórmula básica que relacione todas las variables involucradas en el proceso de extrusión del plástico en la extrusora. Esta fórmula es la siguiente:

$$kg/h = \frac{(a)*(v*60)*(25,4*e \times 10^{-6})*(d*1000)}{100} \quad (2.1)$$

Donde  $a$  es el ancho en centímetros,  $v$  es la velocidad en metros por minuto,  $e$  es el espesor en milésimas de pulgada y  $d$  es la densidad en  $g/cm^3$ . El factor de 60 junto a la velocidad es para convertirla a metros por hora, el factor de  $25,4 \times 10^{-6}$  para el espesor permite convertirlo en metros pues se utiliza el dato en milésimas de pulgada, el factor de 1000 junto a la densidad es para convertirla a kilogramos entre metros cúbicos y finalmente el denominador de 100 es para convertir el ancho a metros.

Como anteriormente se discutió, la medición de las variables de densidad y espesor están fuera del alcance de este proyecto y por ahora la gerencia ha decidido manejarlas como un valor constante que han estimado de manera teórica desde tiempo atrás.

## **Sistema de medición de velocidad de producción lineal de plástico y su montaje mecánico**

La solución para este objetivo es claramente abierta, pues, existen muchas formas de medir la velocidad lineal y el caso de interés no es restrictivo, por lo tanto, es necesario seguir el método de diseño en ingeniería para asegurar que la solución planteada será la correcta.

Según [6], los pasos del método de diseño en ingeniería son los siguientes:

1. Definición del problema.
2. Establecimiento de los criterios para escoger la mejor solución, así como las limitaciones que se deben cumplir.
3. Búsqueda de información pertinente.
4. Generación de la mayor cantidad de soluciones posibles.
5. Análisis y descarte de las soluciones que no son viables.
6. Selección de la mejor solución entre las que quedaron.
7. Especificaciones detalladas de la selección escogida: se aborda en el siguiente apartado del informe.
8. Comunicación escrita sobre la solución escogida: se aborda en el siguiente apartado del informe.

### **Definición del problema para el diseño del sistema de adquisición de velocidad y su montaje mecánico**

Una de las variables para la fórmula del cálculo de la producción en kg/h que requiere la gerencia de Polymer es indudablemente la velocidad. Esta velocidad debe ser expresada en términos de metros por minuto y debe ser medida continuamente durante todo el tiempo en que la máquina esté funcionando. Además, como dato adicional, se requiere de una alerta cuando la máquina esté detenida, por lo tanto, el dispositivo debe de ser capaz de medir también una velocidad de 0 m/min.

El mejor lugar para la colocación del sistema de adquisición de velocidad es sobre el rodillo de presión. Este lugar es de fácil acceso y la información obtenida representa directamente la magnitud de la rapidez con la que el producto está siendo extruido. Ver Figura 4.

## **Criterios para selección y validación del sistema de adquisición de velocidad**

- La velocidad de producción de la máquina E16 va desde los 10 m/min hasta los 110 m/min por lo que el sistema de adquisición de velocidad debe tener el rango suficiente para medir todos los valores entre esos dos límites.
- El sistema de adquisición de velocidad debe de tener la capacidad de detectar cuando la máquina está detenida o cuando no haya recibido alguna señal por más de 4 segundos para lo que se considerará detenida.
- Los lugares más adecuados para la recopilación de la velocidad son los rodillos pues son los elementos que se mantiene en movimiento constante junto a la película, sin embargo, un contacto directo con la película plástica puede poner en riesgo la integridad del producto, por lo que la naturaleza del sistema de adquisición de velocidad debe de ser acorde a esta condición.
- Se considerará como parte del sistema de adquisición de velocidad todo lo vaya desde las partes físicas en contacto con las partes móviles hasta el circuito electrónico encargado de proporcionarle energía y de interpretar el fenómeno físico medido.
- El montaje y el mantenimiento del sistema de adquisición de velocidad debe de ser lo más sencillo posible en número de pasos y en complejidad.

## **Búsqueda de información pertinente para la generación de la señal del sistema de velocidad**

La búsqueda de información se realizará a través de las hojas de datos o publicaciones oficiales del proveedor que proporcionen la información técnica necesaria, de no ser posible conseguir la información de manera oficial, se considerará la opción de adquirir el equipo, elemento o sensor y llevar a cabo pruebas aisladas que puedan proporcionar información útil sobre el comportamiento de dicho elemento.

Los datos más importantes que se deben encontrar en la hoja de datos serán:

- Respuesta en frecuencia: es de vital importancia que la banda activa del sensor permita que la velocidad medida vaya desde los 0m/min hasta los 121m/min.
- Propiedades eléctricas: si bien no necesariamente el equipo será excluido por esta razón, es importante tomar en cuenta si será necesario algún tipo de convertidor de voltaje o protección adicional contra corriente para el circuito.
- Tamaño: esto influye directamente en el tamaño del montaje y diseño mecánicos sistema de adquisición de velocidad y podrá ser una eventual limitante.

- Complejidad de la señal: por cuestiones de tiempo y manejo de los recursos, es preferible una única señal discreta en comparación a una señal analógica o varias señales discretas.

### **Generación de la mayor cantidad de soluciones posibles para la producción de la señal del sistema de adquisición de velocidad**

Si se toma en cuenta que la medición de velocidad se hace por lo general con el valor de la distancia entre el valor del tiempo podemos descomponer este problema en dos problemas más sencillos: la medición de la distancia y/o posición, y la medición del tiempo.

### **Medición de la distancia y/o posición**

Para adquirir una medida de distancia y/o posición se pueden utilizar equipos ópticos, piezoeléctricos, ultrasónicos, codificadores, entre otros.

Se procederá a mencionar los posibles equipos que cumplen con esta condición y a contrastarlos con los criterios y la información obtenida:

- Equipo óptico: el equipo óptico para la medición de la distancia puede tener dos naturalezas:
  - Medición de tiempo de vuelo.
  - Medición de distorsión o cambio de la imagen según un sistema óptico de detección de bordes o líneas que se mueven de un punto a otro de la imagen o estereoscópico de al menos dos cámaras.
- Equipo piezoeléctrico: estos equipos generan una señal eléctrica analógica cuando reciben un valor de presión.
- Equipo ultrasónico: por lo general constan de un emisor de 10 pulsos ultrasónicos en serie y un detector de eco para la medición del tiempo de vuelo del sonido.
- Codificador: Usualmente se utilizan en sistemas donde la posición relativa del elemento a controlar es muy importante y para medirla se utilizan foto diodos o foto transistores que puedan contar el número y la posición de los pasos. Existe otro tipo de codificador más sencillo llamado *codificador incremental*.
- Sensor de efecto Hall.

## **Análisis y descarte de las posibles soluciones acerca de la generación de la señal del sistema de adquisición de velocidad**

Para la selección del sensor adecuado se utilizarán tablas de comparación tomando en cuenta una nota ponderada, serán los criterios mencionados anteriormente los que dicten las propiedades de interés y se ordenarán según la relevancia que cada una de ellas represente.

Adicionalmente se hará un análisis cualitativo con las respectivas ventajas y desventajas que pueda ofrecer cada posible solución para la ejecución global del proyecto, estas consideraciones serán importantes en caso de algún empate técnico entre las opciones presentadas.

Las propiedades principales para tener en cuenta son:

- Adaptabilidad al proceso de producción: en respuesta a la necesidad de que el sistema de medición sea lo menos invasivo posible. A esta propiedad se le asignan 7 puntos sobre el ponderado.
- Respuesta en frecuencia: a esta propiedad se le asignará un peso de 6 puntos sobre el ponderado.
- Precio: a esta propiedad se le asignan 5 puntos sobre el ponderado.
- Complejidad de la señal: una señal muy compleja puede subir los costos y además agotar el recurso del tiempo sin necesidad. A esta propiedad se le asignan 4 puntos sobre el ponderado.
- Tamaño: a esta propiedad se le asigna un peso de 3 puntos sobre el ponderado.
- Disponibilidad: a esta propiedad se le asignan 2 puntos sobre el ponderado.
- Propiedades eléctricas: al no ser restrictiva se le asigna 1 punto sobre el ponderado.

Para el cálculo de la nota ponderada, a cada casilla se le asignan 5 posibles valores:

- 0: totalmente inadecuado.
- 1: parcialmente inadecuado.
- 2: indefinido.
- 3: parcialmente adecuado.
- 4: totalmente adecuado.

A continuación, se explica el significado de las puntuaciones para cada una de las propiedades:

- Adaptabilidad al proceso de producción:
  - Totalmente inadecuado: cuando el producto mismo tenga que ser modificado para posibilitar la medición.

- Parcialmente inadecuado: cuando los elementos móviles de la extrusora deban de ser modificados o la producción se deba detener por un tiempo largo.
  - Indefinido: cuando no exista información referente.
  - Parcialmente adecuado: cuando la producción deba de ser detenida por un tiempo corto debido al montaje del sistema de adquisición.
  - Totalmente adecuado: cuando no haya paro de la producción ni deba de haber modificación a las partes móviles.
- Respuesta en frecuencia:
    - Totalmente inadecuado: cuando la frecuencia de muestreo sea menor a 1 Hz.
    - Parcialmente inadecuado: cuando la mayor frecuencia esté en el límite de lo que permita la medición adecuada del rango de producción o sea menor o igual a los 5 Hz.
    - Indefinido: cuando no exista información referente.
    - Parcialmente adecuado: cuando la frecuencia de muestreo sea mayor a los 5 Hz y menor a los 100 Hz.
    - Totalmente adecuado: cuando la frecuencia de muestreo sea mayor a los 100 Hz.

En caso de que haya dos o más sensores en una de estas categorías, se otorgará el mayor puntaje al que tenga la mayor frecuencia comprobable en la literatura.

La razón por la que se escogen estas frecuencias es debido a los límites de producción que se van a utilizar; si se convierten los 121 m/min de velocidad máxima a m/s se obtiene 2,02 m/s, esto quiere decir que si se tiene una frecuencia de muestreo de 1Hz se tendrían que medir hasta 2 metros con una sola muestra. Si se tienen 5Hz se tendrían que medir 40 cm con cada pulso, con 100 Hz se tendría una resolución de 2 cm y esto es un error de apenas 1%.

- Precio: se ordenarán de menor a mayor precio con un 0 cuando el precio sea el mayor. O se asignará una puntuación de 0 cuando no se tenga disponible.
- Complejidad de la señal:
  - Totalmente inadecuado: cuando requiera de equipo altamente especializado adicional a un solo microprocesador o computador personal.
  - Parcialmente inadecuado: cuando su señal pueda ser obtenida directamente por un microprocesador, pero requiera de un protocolo distinto a USART. Este protocolo es preferible debido a la facilidad de implementación y flexibilidad que ofrece pues es universal sin

necesidad de sincronizar las frecuencias de operación de los distintos equipos.

- Indefinido: no se utilizará.
  - Parcialmente adecuado: cuando la señal sea analógica, pero se pueda filtrar utilizando algún circuito digital y un microprocesador, o cuando sea discreta en más de dos niveles.
  - Totalmente adecuado: cuando la señal sea discreta y binaria.
- Tamaño: se ordenará de menor a mayor.
  - Disponibilidad:
    - Totalmente inadecuado: cuando no se provea en el país.
    - Parcialmente inadecuado: cuando se haya agotado, pero se espera su existencia o provisión pronta, o esté disponible en una sola tienda con menos de 3 en existencia.
    - Indefinido: no se utilizará
    - Parcialmente adecuado: cuando hayan menos de 10 en existencia en una sola tienda.
    - Totalmente adecuado: cuando hayan más de 10 en cualquier tienda o en varias tiendas.
  - Propiedades eléctricas:
    - Totalmente inadecuado: cuando su alimentación esté por encima de los 12 voltios.
    - Parcialmente inadecuado: cuando su alimentación esté por encima de los 5 voltios o por debajo de los 3,3 voltios.
    - Indefinido: cuando no se tenga información.
    - Parcialmente adecuado: cuando su alimentación esté entre los 0 y 3,3 voltios.
    - Totalmente adecuado: cuando no requiera alimentación o su alimentación sea de 5 voltios.

La razón de estos valores es debido a lo largamente estandarizado de la operación de los equipos digitales y analógicos entre los 3,3 V y los 5 V.

Después de realizar una investigación sobre algunos sistemas en el mercado que permiten la adquisición de datos sobre la velocidad, se seleccionaron los siguientes candidatos:

## Sensor de tiempo de vuelo VL6180X

La medición del tiempo de vuelo se hace al cronometrar el tiempo que un haz de luz dura en rebotar desde un objeto hasta el sensor. Para el caso estudiado, esto representa un problema mayor pues surge la pregunta: ¿entre qué punto y qué punto de la película plástica se medirá esta distancia? ¿se puede medir esto sin modificar la naturaleza de la filmina plástica y su calidad?

La respuesta a la primera pregunta es completamente incierta, además representaría el costo adicional de marcar toda la filmina desde su formación para poder tener dos puntos de referencia lo cual es completamente inaceptable, o de la modificación del rodillo en el que se mueve la filmina plástica. La respuesta a la segunda pregunta es negativa y esto genera un conflicto directo también con la segunda pregunta de los criterios de selección para el espacio físico “¿Modifica de alguna manera negativa la producción y/o la calidad del producto la ubicación de una de las etapas de medición o del sistema completo?”

Según su hoja de datos [10], en la categoría de medidor de tiempo de vuelo:

- Utiliza el protocolo  $I^2C$  para la comunicación.
- Tiene un rango de operación de entre los 0 y 10 cm.
- Su tamaño es de 4,8 x 2,8 x 1 mm por lo que es muy compacto.
- Su voltaje de operación es desde los 2,7 V hasta los 2,9 V.
- Su frecuencia de operación es de 10 Hz.
- En la página de Mouser [17] no está en existencia y su precio no se conoce.
- Debido a su funcionamiento, requiere de un punto de referencia para medir la distancia y un dispositivo adicional para la medición del tiempo entre señales.

Debido a las razones anteriores su puntaje ponderado es de 1,714. Ver Tabla 1.

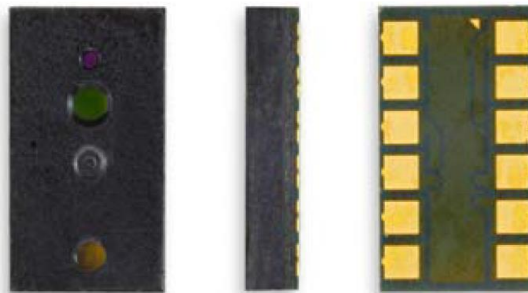
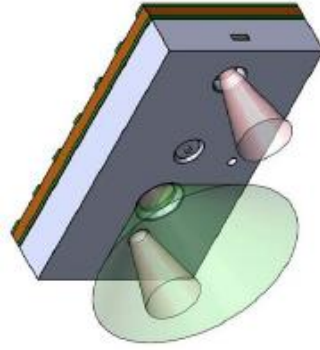


Figura 30. Sensor VL6180X. Fuente: adaptado de [10]





*Figura 31. Isométrico y representación gráfica de sus conos ópticos. Fuente: adaptado de [10]*

### **Cámara Basler Racer GigE**

La detección de bordes o líneas móviles de un punto a otro puede ser realizada con una cámara de no muy alta resolución y sería una muy buena opción en el caso de existir dichas líneas o bordes de manera fácilmente apreciable (esto es, de alto contraste) en la filmina plástica, sin embargo, como no es el caso, también la implementación de este sistema modificaría inevitablemente el producto o el rodillo al hacer necesaria la creación de estas líneas de referencia y esto es también inaceptable en el caso del producto e inconveniente en el caso del rodillo.

Según su manual de usuario [11], en la categoría de sistema óptico:

- Utiliza el protocolo Ethernet.
- La frecuencia de muestreo está por encima de los 100 Hz.
- Alimentación desde los 12 V hasta los 24 V.
- Su tamaño es mayor a los 36,12 x 56 x 62 mm.
- Requiere del software Basler pylon Camera Software Suite.
- Se comercializa en Alemania.
- Su precio es de los 1120 USD según [14].
- De acuerdo con su funcionamiento, requeriría de la creación de líneas visibles en el producto.

Debido a los criterios anteriores, su ponderado es de 0,857. Ver Tabla 1.



Figura 32. Cámaras Basler Racer. Fuente: adaptado de [11].

### **Diafragma piezoeléctrico 10293**

Por lo general este tipo de sensores se utiliza cuando se requiere medir la intensidad de presión o la vibración en algún equipo de interés. En el caso del presente proyecto se podría utilizar como un tipo de sensor de presencia de la filmina plástica en el rodillo: se activa cuando el rodillo gira hasta donde está pasando el plástico y se obtiene una señal, esta señal terminaría cuando dejen de coincidir la filmina y el sensor; la señal podría tratarse con un *Schmitt trigger* para convertirla a discreta y marcar el momento en el que se detecta la filmina y el momento en que se detiene el contacto con la filmina.

Esta posible solución genera algunas dificultades como el manejo de los cables conductores pues estarían girando en todo momento y lo principalmente negativo es la necesidad de entrar en contacto directo con el material y esto podría poner en juego su integridad. Adicionalmente se desconoce la presión que ejercería la película sobre el sensor y si esta presión fuese suficiente como para detectar el contacto.

Utilizado comúnmente para la producción de vibraciones, también se puede utilizar de forma contraria para recibir un voltaje e interpretarlo, según su hoja de datos [12]:

- La señal que entrega es analógica sin embargo puede ser susceptible a ruido.
- La frecuencia de operación es de menos de  $2,2 \pm 0,3$  kHz en los de menor frecuencia.
- No se especifica su uso como sensor de movimiento o vibración por lo que no se especifica su voltaje de operación.
- Su tamaño es de 41 mm de diámetro máximo y un grosor de 0,63 mm como máximo.
- Se comercializa localmente en tiendas como Cibernética [15] y Microjpm [16].

- En Microjpm [16] no está disponible sin embargo se ha solicitado y su precio es de \$1,5; en Crcibernética [15] sólo hay 5 en existencia y tiene un costo de \$0,5.
- Para que funcione debe de estar en contacto directo con la filmina plástica.

Su puntuación ponderada es de 3,036. Ver Tabla 1.

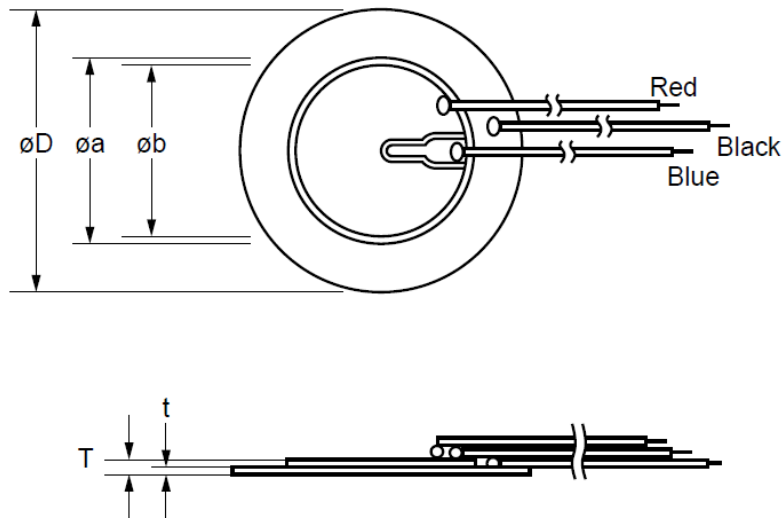


Figura 33. Diagrama del diafragma piezoeléctrico 10293. Fuente: adaptado de [12].

### Sensor de ultrasonido HC-SR04

Este sensor tiene la ventaja de ser altamente resistente al ruido externo pues su receptor de eco reacciona a la frecuencia emitida, sin embargo, de manera similar a los sistemas ópticos, requeriría la realización de puntos de referencia a lo largo de toda la filmina o en el rodillo y esto no es práctico ni recomendable.

Según su hoja de datos [13]:

- Tiene una resolución de 3 mm y necesita de una superficie plana y perpendicular a su señal para ser lo más exacto posible.
- Su frecuencia de muestreo es de 40 Hz.
- Su voltaje de alimentación es 5V.
- Su tamaño es de 45 x 20 x 15 mm.
- Su señal es un pulso en alto al recibir el eco de 8 pulsos ultrasónicos; la distancia se interpreta al medir el tiempo de retorno.
- Al momento de la selección, en la tienda Crcibernética [15] se encuentran 8 unidades, con un precio de \$4,45 y en la tienda Microjpm [16] se encuentran 15 unidades con un precio de \$4,95.

Para que el sensor pueda medir la distancia, se debe de modificar alguno de los elementos móviles de la máquina y así permitir que haya una superficie plana.

Su ponderado es de 2,643. Ver Tabla 1.



Figura 34. Módulo HC-SR04. Fuente: Adaptado de [13].

### **Codificador Wurth Elektronik 482009514001**

La página web de Mouser [7] ofrece más de 150 opciones similares en tanto a codificadores se refiere, para agilizar el descarte, se pre-seleccionó el 482009514001 como candidato por las características descritas a continuación.

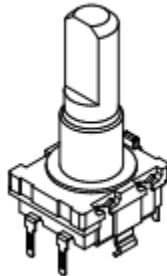
El codificador ofrece una solución discreta que únicamente requiere de su montaje en el rodillo de la máquina y un elemento de medición de luz para el control de la posición y el movimiento con una o varias entradas discretas.

En el párrafo de introducción a este apartado se mencionan 'entre otros' las opciones para la medición de la velocidad y esto es porque cualquier elemento o sensor que permite observar el cambio de posición relativa respecto del tiempo puede ser utilizado para medir la velocidad, algunas opciones como el decodificador pueden ayudar a controlar incluso la dirección y la posición de manera muy precisa.

Según su hoja de datos [8]:

- Su tamaño es de 13,75 x 11,7 x 26,6 mm.
- Su resolución es de 8 pulsos por vuelta.
- No se especifica la frecuencia de operación.

- Su voltaje de operación son 5V.
- La señal entregada es de pulsos discretos.
- Está limitado a una vida útil de 15000 ciclos.
- Su precio es de ₡2801.
- Según [7] existen más de 700 en el momento de la selección.



*Figura 35. Isométrico del codificador Wurth Elektronik 482009514001. Fuente: adaptado de [8].*

Al ser de naturaleza rotatoria, su eje se puede desgastar rápidamente y dar información incorrecta o fallar mecánicamente y comprometer la productividad de la máquina y la integridad del sistema de medición de la velocidad.

El decodificador, como se mencionó antes, resulta altamente útil cuando otra información necesaria sea la posición y la dirección del movimiento, sin embargo, para los objetivos del presente proyecto, estas características de la producción no son relevantes por lo tanto si los beneficios del decodificador sobre el sensor de efecto Hall no son significativos, se procederá a descartar este y a seleccionar el de efecto Hall. Su ponderado según la Tabla 1 es de 2,571.

### **Sensor de efecto Hall A3144**

Como su nombre lo expresa, se basa en el efecto hall y en la ley de Faraday que produce un campo magnético en movimiento al pasar cerca de su circuito en el cual se produce un corto debido al movimiento de cargas.

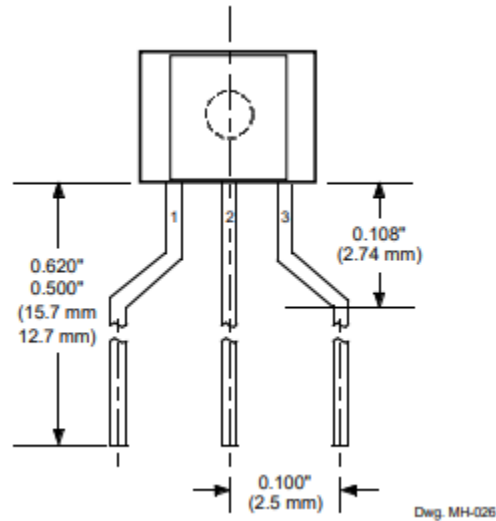
La interpretación de este cambio de estado es muy útil para el cronometraje del movimiento relativo de un elemento giratorio y, por consiguiente, el cálculo de su frecuencia.

Según su hoja de datos [9]:

- El voltaje de operación puede estar entre los 4,5V a los 24V.
- El tamaño es de 4,17 x 1,57 x 19,36 mm como máximo.
- Su señal se da en forma de pulsos.

- Los tiempos de los flancos son de  $2\mu\text{s}$  por lo que en total el pulso completo tendría una extensión de un mínimo de  $4\mu\text{s}$ , esto significa que la frecuencia de operación máxima es de 500 kHz.
- Se encuentra disponible en la tienda Crcibernetica a un precio de \$0,99 y con 199 en existencia al momento de la selección.

Según los criterios de selección, su puntaje ponderado es de 3,929. Ver Tabla 1.



*Figura 36. Sensor de efecto Hall 3144. Fuente: adaptado de [9].*

El tamaño tan reducido de este sensor permite la ubicación cómoda y un diseño más flexible de su montaje mecánico. Y lo único adicional que se requiere es un imán permanente que se encuentre a una distancia adecuada del sensor para generar un pulso cada vez que el rodillo rote.

Una ventaja adicional de esta opción es que se puede diseñar un ensamblaje de tal manera que la distancia siempre se mantenga constante y lo único por medir sea el tiempo que se toman los pulsos en generarse.

Otro criterio importante para preferir el sensor de efecto Hall sobre el decodificador es la complejidad de la señal pues el primero ofrece solamente un cambio de estado y el segundo produce dos o más señales discretas que no se necesitan realmente por lo que se estarían desperdiciando y provocan una carga computacional más alta.

Tabla 1. Comparación para cálculo de puntuación ponderada de los candidatos. Fuente: elaboración propia.

Nombre del sensor	Adaptabilidad al proceso de producción (7)	Respuesta en frecuencia (6)	Precio (5)	Complejidad de la señal (4)	Tamaño (3)	Disponibilidad (2)	Propiedades eléctricas (1)	Puntuación
STMicroelectronics VL6180	1	3	0	1	4	0	1	1,500
Cámara Basler Racer	0	4	0	0	0	0	0	0,857
Diafragma piezo eléctrico 10293	2	2	4	3	3	3	2	2,679
Sensor Ultrasónico HC-SR04	1	3	2	4	1	4	4	2,357
Codificador Wurth Elektronik 482009514001	3	2	1	4	2	4	4	2,571
Sensor de efecto Hall A3144	4	4	3	4	5	4	4	3,929

La Tabla 1. Representa gráficamente los criterios utilizados, para el cálculo de la puntuación se utiliza la fórmula:

$$Puntuación = \frac{A \times 7 + R \times 6 + P \times 5 + C \times 4 + T \times 3 + D \times 2 + P.e. \times 1}{7!} \quad (2.2)$$

Donde: *A* es el puntaje asignado en adaptabilidad al proceso de producción, *R* es el puntaje asignado en la respuesta en frecuencia, *P* es el puntaje asignado en precio, *C* es el puntaje asignado en la complejidad de la señal, *T* es el puntaje asignado en el tamaño, *D* es el puntaje asignado en la disponibilidad y *P.e.* es el puntaje asignado en las propiedades eléctricas.

### Criterios para el montaje y diseño mecánicos del sistema de adquisición de velocidad

- El diseño y montaje de la parte mecánica deben ser lo más sencillos posibles para optimizar los recursos disponibles de tiempo y mano de obra sin sacrificar desempeño y durabilidad.
- El diseño y montaje mecánicos deben buscar la reducción de cualquier fuente de desgaste posible tanto en sí mismos como en la máquina.
- La naturaleza del diseño no será restringida en su factor de forma, sin embargo, debe de ser lo más sencillamente representable y transportable sin sacrificar desempeño y durabilidad.
- Los elementos móviles se deben mantener al mínimo posible sin sacrificar desempeño y durabilidad para evitar la necesidad de reponer piezas por el mayor tiempo posible.
- Los elementos móviles deben soportar la carga y el movimiento por, al menos, el tiempo requerido para recuperar la inversión inicial e idealmente con una vida útil infinita.

- El montaje mecánico del sistema de adquisición debe de ser lo menos invasivo posible sin sacrificar desempeño y durabilidad para no afectar el calendario de producción de la extrusora, o afectarlo en la menor medida posible.
- Los elementos estáticos deben soportar la carga estática con un factor de seguridad de al menos el 1.2 para garantizar su operación segura.
- Los elementos del montaje mecánico del sistema de adquisición de velocidad que estén sometidos a cargas de fatiga deben de soportar esta condición por, al menos, el tiempo en que se recupere la inversión inicial e idealmente con una vida útil infinita.
- El sensor de efecto Hall escogido, funciona en conjunto con un campo magnético generado por un imán permanente o artificial, por lo que inevitablemente el diseño mecánico del sistema de adquisición tiene que tomar en cuenta la colocación de dichos imanes.

### **Búsqueda de información necesaria para el diseño mecánico del sistema de adquisición de velocidad**

Algunos puntos clave que se puedan necesitar para tomar una decisión responsable e informada respecto al diseño y eventual construcción del sistema de adquisición de velocidad son:

- Propiedades mecánicas de los materiales propuestos y su resistencia a la compresión, tensión y esfuerzo cortante.
- Resistencia a la fatiga de los posibles materiales a utilizar; o de los elementos móviles tales como los cojinetes.
- Las condiciones físicas en las que podría llegar a estar ubicado el sistema de adquisición: esta información será recopilada en sitio y se tomará en cuenta para el análisis en fatiga de los elementos móviles.

### **Generación y descarte de la mayor cantidad de soluciones para el diseño y el montaje mecánicos del sistema de adquisición de velocidad**

Una de las medidas cruciales para el cálculo de la velocidad es la distancia, sin embargo, esta no tiene por qué ser necesariamente variable, sino que puede ser controlada o definida de manera constante y que quede sujeta al comportamiento intrínseco del sistema de adquisición.

Para lograr esta definición constante de la velocidad, se puede diseñar un mecanismo con un comportamiento cíclico.



Algunas opciones para la naturaleza cíclica del ensamblaje son:

- Selección de un electroimán que marque el paso por el sensor de efecto Hall:

Esta posibilidad requiere necesariamente de un circuito de alimentación y un espacio adicional para dicho circuito y si bien esta propuesta no entra en conflicto directo con ninguna de las condiciones de validación, agrega elementos innecesarios al montaje y al sistema de adquisición de la velocidad. En caso de que otra opción más simple no funcione, se revisará pues esta opción tiene la ventaja de que la intensidad de campo magnético se puede cambiar.

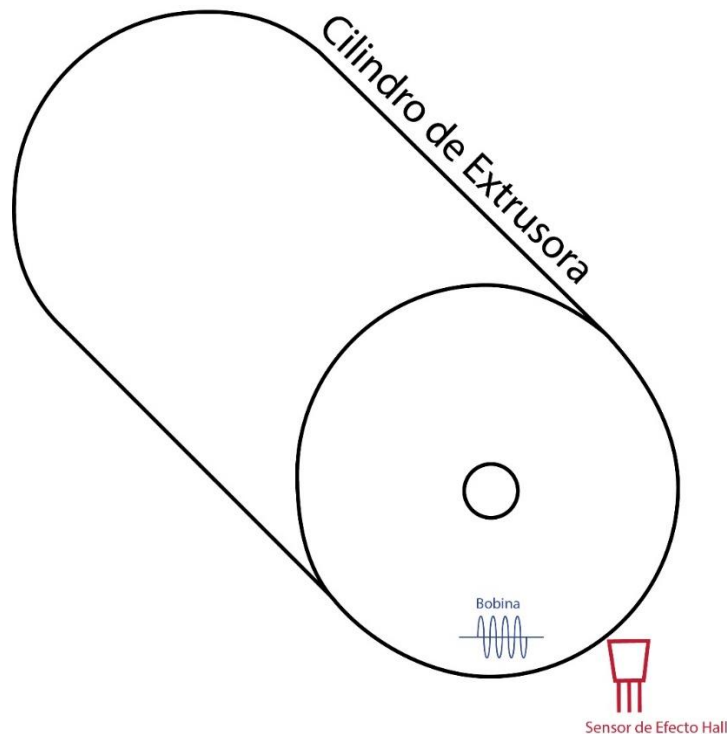
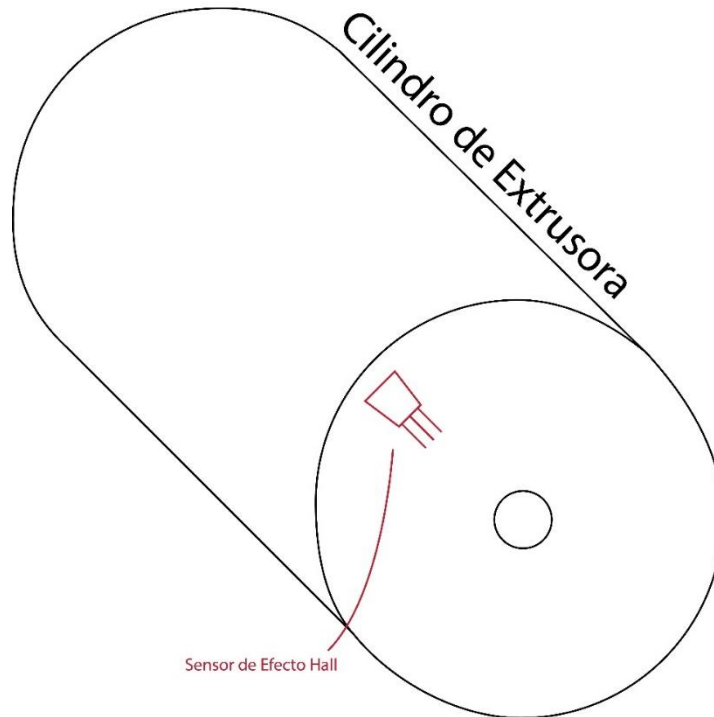


Figura 37. Ilustración de propuesta del electroimán. Fuente: Elaboración propia.

- Montaje del sensor de efecto Hall en el rodillo de la extrusora:

Esta opción significaría mantener el sensor en movimiento y dicho movimiento puede introducir error innecesario, además, de fatiga en los contactos y el movimiento del cable y/o el sistema de alimentación para el sensor. Esta propuesta entra en conflicto directo con la sencillez del ensamblaje y la reducción de sus partes móviles, además, también podría representar la detención de la máquina para instalar el sensor de efecto Hall en el rodillo.

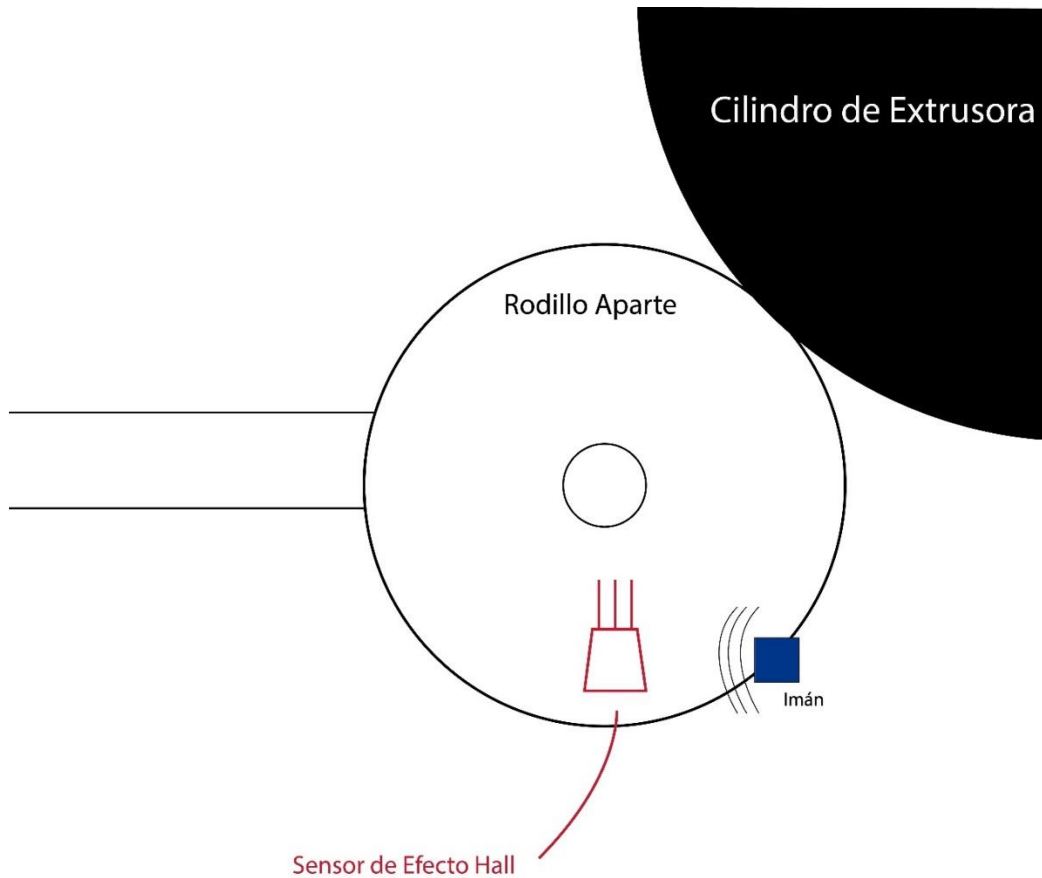


*Figura 38. Representación visual del montaje del sensor de efecto Hall en el rodillo de la extrusora. Fuente: Elaboración propia.*

- Montaje del sensor de efecto Hall en un rodillo/rueda aparte:

Esta propuesta deja de lado la posible detención de la máquina, sin embargo, al igual que la anterior, representaría una fuente de movimiento innecesario para el sensor, lo que puede conllevar a un error de medición y también a la fatiga de los cables de alimentación. Estas consideraciones entran directamente en conflicto con las condiciones de la menor cantidad de partes móviles y la sencillez del ensamblaje.

Aun así, un rodillo aparte podría ser una buena opción para evitar la detención de la máquina y, además, lo hace muy fácil de transportar en caso de que se requiera replicar en otra etapa de la máquina o en otra máquina.



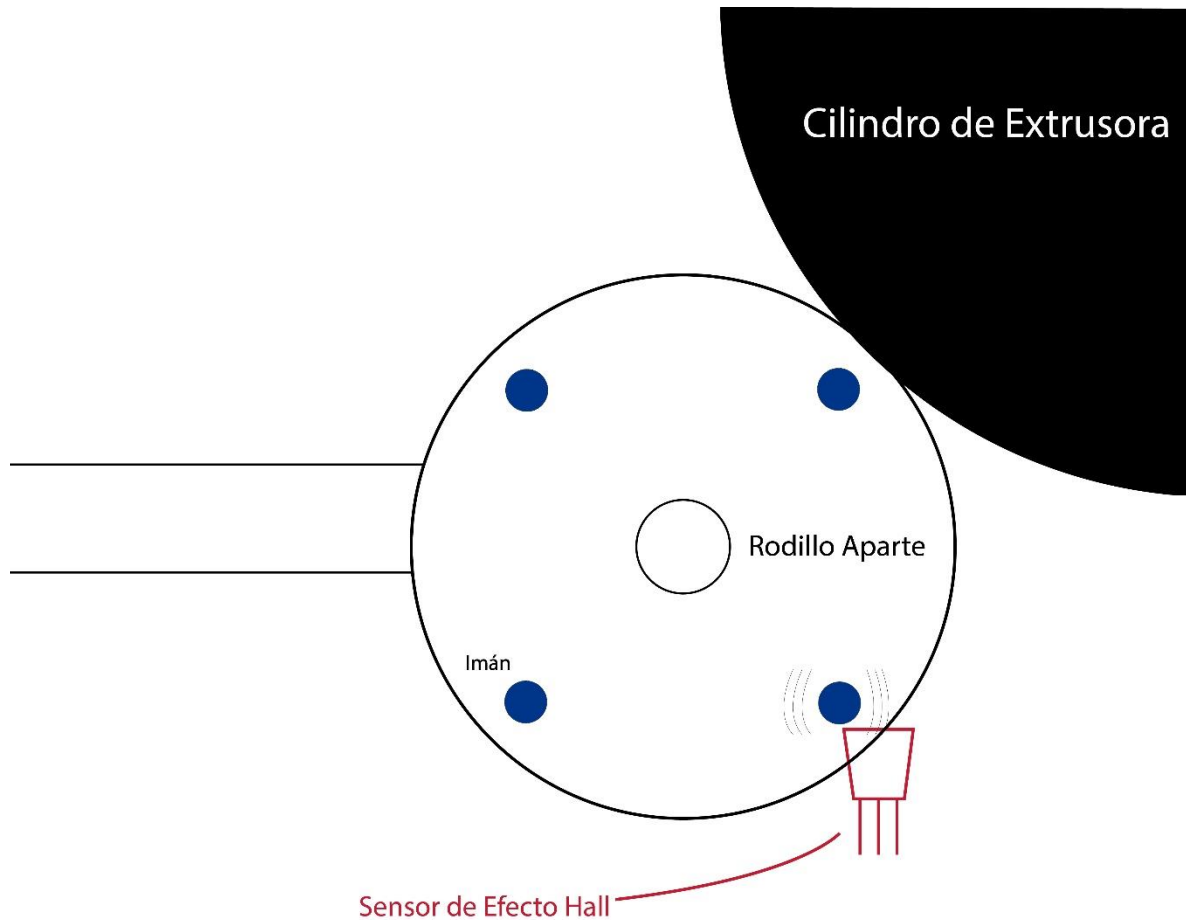
*Figura 39. Representación del sensor sobre un rodillo aparte. Fuente: Elaboración propia.*

- Montaje de uno o varios imanes permanentes en un rodillo aparte:

El hecho de que los imanes en este caso sean permanentes, elimina la necesidad de un circuito de alimentación para ellos; además, el tamaño del rodillo permite mantener una medida constante de la distancia entre los pulsos medidos y permite, hasta cierto límite, 'jugar' con la frecuencia de muestreo del sensor de efecto Hall.

También esta posibilidad permite que el sensor de efecto Hall se mantenga en una posición fija por lo que se reduce la cantidad de elementos móviles y la complejidad total del ensamblaje sin sacrificar su desempeño.

Además, este rodillo debe de ser de un material no ferromagnético para evitar corrientes parásitas en sí mismo y campos magnéticos artificiales que puedan introducir error al sistema.



*Figura 40. Representación de la colocación de imanes permanentes en un rodillo aparte para la adquisición de la velocidad. Fuente: Elaboración propia.*

### **Criterios de selección del mecanismo con comportamiento cíclico**

De manera similar al proceso de selección del sensor como tal, se procederá a definir una escala de puntuación de acuerdo con distintos ciertos criterios importantes con la misma escala utilizada del 0 al 4:

- 0: totalmente inadecuado.
- 1: parcialmente inadecuado.
- 2: neutral.
- 3: parcialmente adecuado.
- 4: totalmente adecuado.

Y los criterios de selección del mecanismo:

- Complejidad de su construcción: esto será un indicador de la cantidad de pasos para su construcción, piezas móviles y no móviles que sean necesarias para su construcción y de la cantidad de elementos que deban

someterse a la fatiga. Las opciones se ordenarán de mayor a menor y se le asignará un peso de 4 a este criterio. Un puntaje mayor significa una complejidad de construcción menor.

- Ajuste al proceso de producción: esto será un indicador de qué tan invasivo sería el mecanismo como tal dentro del proceso de producción. A este criterio se le asignarán 3 puntos sobre el ponderado.
  - Totalmente inadecuado: cuando alguna etapa de la producción deba de ser modificada de manera permanente para su utilización.
  - Parcialmente inadecuado: cuando alguna etapa deba de ser modificada de manera temporal para su funcionamiento o que su posicionamiento resulte incómodo para los operadores.
  - Neutral: no se utilizará.
  - Parcialmente adecuado: cuando su operación pueda incomodar levemente a los operadores de la máquina con respecto a su tamaño, posición, movimiento, etc.
  - Totalmente adecuado: cuando no se presente ninguno de estos inconvenientes.
- Probabilidad de detención de la máquina: es posible que se deba realizar una detención de máquina para adaptar el mecanismo del sistema de medición de velocidad. A este criterio se le agregaran 2 puntos sobre el ponderado.
  - Totalmente inadecuado: cuando la máquina deba de ser detenida por un espacio mayor a 3 horas aproximadamente para su montaje.
  - Parcialmente inadecuado: cuando la máquina deba de ser detenida una cantidad de tiempo mayor a 1 hora, pero menor a 3.
  - Neutral: cuando la máquina deba de ser detenida por un tiempo mayor a 30 minutos para su montaje.
  - Parcialmente adecuado: cuando la máquina deba de ser detenida por menos de 30 minutos para su montaje.
  - Totalmente adecuado: cuando no se requiera detención en absoluto.
- Facilidad de mantenimiento: de ser necesaria la sustitución o el rediseño de alguna pieza, el ensamblaje debe de ser lo más amigable y accesible posible. Se considera una parte no móvil al sistema de adquisición mismo. A este criterio se le agregará 1 punto sobre el ponderado.
  - Totalmente inadecuado: cuando se deba remover varias partes móviles de la extrusora para poder darle mantenimiento al sistema de adquisición de datos de velocidad.
  - Parcialmente inadecuado: cuando se deba remover una parte móvil de la extrusora para poder darle mantenimiento al sistema de adquisición de datos de velocidad.
  - Neutral: cuando se deba remover varias partes no móviles de la extrusora.
  - Parcialmente adecuado: cuando se deba remover sólo una parte no móvil de la extrusora.

- Totalmente adecuado: cuando no se requiera la remoción de ninguna parte de la extrusora.

La fórmula utilizada para calcular la puntuación ponderada de cada opción será:

$$Puntuación = \frac{C \times 4 + A \times 3 + D \times 2 + M \times 1}{4!} \quad (2.3)$$

Donde  $C$  es la complejidad de construcción,  $A$  es el ajuste al proceso de producción,  $D$  es la probabilidad de detención de la máquina y  $M$  es la facilidad de mantenimiento.

### **Análisis de las opciones planteadas para el movimiento cíclico**

- Construcción de electroimán que marque el paso cerca del sensor de efecto Hall:

De acuerdo con su funcionamiento, se requeriría de un circuito de alimentación para el electroimán y para el sensor de efecto hall; adicional a esto, alguno de los dos elementos tendrá que estar en movimiento por lo que los conductores para alimentación y señal estarían sometidos a constante movimiento y fatiga.

La complejidad, además, significa necesariamente un tamaño mayor del sistema completo por lo que también podría influir negativamente en el proceso productivo.

El electroimán o el sensor podrían necesitar ser cambiados por lo que el mantenimiento puede ser más costoso.

La complejidad del circuito podría también representar la necesidad de detener la máquina por una o más horas para mantener la seguridad del personal.

- Complejidad de construcción: muy alta. 1 punto.
- Ajuste al proceso de producción: parcialmente inadecuada. 1 punto.
- Probabilidad de detención de la máquina: parcialmente inadecuada. 1 punto.
- Facilidad de mantenimiento: neutral. 2 puntos.

Su puntuación ponderada es 1,1. Ver Tabla 2.

- Montaje del sensor de efecto Hall en el rodillo de la extrusora:

Para esta propuesta, los conductores de alimentación y de señal estarán en constante movimiento y expuestos a la fatiga; adicionalmente, sería obligatoria la detención de la máquina para poder remover inicialmente el rodillo y para darle mantenimiento.

El movimiento constante de los cables conductores podría representar una amenaza a la seguridad de los operadores y a la integridad física de la máquina.

- Complejidad de construcción: alta. 2 puntos.
- Ajuste al proceso de producción: totalmente inadecuada. 0 puntos.
- Probabilidad de detención de la máquina: totalmente inadecuada. 0 puntos.
- Facilidad de mantenimiento: totalmente inadecuada. 0 puntos.

Su puntuación ponderada es 0,8. Ver Tabla 2.

- Montaje del sensor de efecto Hall en un rodillo/rueda aparte:

Con esta opción se eliminaría la necesidad de desmontar el rodillo de la extrusora sin embargo se mantiene la condición de movimiento para los cables conductores de la alimentación y de la señal.

- Complejidad de construcción: media. 3 puntos.
- Ajuste al proceso de producción: parcialmente inadecuado. 1 punto.
- Probabilidad de detención de la máquina: parcialmente adecuada. 3 puntos.
- Facilidad de mantenimiento: parcialmente adecuada. 3 puntos.

Su puntuación ponderada es 2,4. Ver Tabla 2.

- Montaje de uno o varios imanes permanentes en un rodillo aparte:

En este caso los únicos elementos móviles que habrá que considerar son el cojinete necesario para el rodillo aparte y el rodillo como tal. Los imanes permanentes eliminan la necesidad de un circuito de alimentación para el actuador magnético y del movimiento de los cables conductores por lo que se mejora la seguridad y la durabilidad.

- Complejidad de construcción: baja. 4 puntos.
- Ajuste al proceso de producción: totalmente adecuada. 4 puntos.
- Probabilidad de detención de la máquina: totalmente adecuada. 4 puntos.
- Facilidad de mantenimiento: totalmente adecuada. 4 puntos.

Su puntuación ponderada es 4. Ver Tabla 2.

Tabla 2. Comparación de las opciones para la construcción del sistema de adquisición de la velocidad. Fuente: elaboración propia.

Propuesta	Complejidad de construcción	Ajuste al proceso de producción	Probabilidad de detención	Facilidad de mantenimiento	Puntuación
Electroimán	1	1	1	2	1,100
Montaje de sensor en rodillo de máquina	2	0	0	0	0,800
Montaje del sensor en rodillo aparte	3	1	3	3	2,400
Montaje de imanes en rodillo aparte	4	4	4	4	4,000

Gracias al análisis anterior, se puede determinar que la mejor opción es la colocación de imanes permanentes en un rodillo aparte.



Figura 41. Flujo de oportunidad para los sistemas de medición de velocidad. Fuente: elaboración propia.

## Sobre el giro del rodillo

Como se discutió anteriormente, la opción más completa es la adición de un rodillo móvil aparte, este rodillo móvil tendrá que estar sobre algún elemento que le permita girar; si se remitiera el diseño al ajuste de eje-agujero la fricción podría ser demasiado significativa y esta fricción podría provocar fatiga térmica y la formación de concentradores de esfuerzo en la superficie tanto del eje como del agujero. Lo mejor en este caso sería seleccionar un cojinete que permita el giro libremente.



Para la selección del rodamiento se tomarán en cuenta:

- El peso del rodillo, del eje y de los imanes.
- El tamaño posible del eje.
- Una vida útil de por lo menos un año e idealmente de vida infinita.
- El catálogo de NSK [9]: esta es la línea de productos es la que se trabaja en la compañía; siempre y cuando el desempeño del rodamiento cumpla con las especificaciones necesarias para este diseño.

Debido a que la selección de este elemento es recursiva, la descripción del cojinete por utilizar se describirá en el siguiente apartado de este informe: Propuesta de Diseño, según el método descrito en [9].

### **Sistema de medición de ancho del producto extruido por la máquina E16**

La solución para este objetivo es claramente abierta, pues, existen muchas formas de medir el ancho de la película y el caso de interés no es restrictivo, por lo tanto, es necesario seguir el método de diseño en ingeniería para asegurar que la solución planteada será la correcta.

Se seguirán los pasos de [6] de la misma manera que en el subapartado anterior.

Como se explicó en el apartado de la Definición del Problema, las etapas donde se controla el ancho son en el soplado, ver Figura 2., y en el corte, ver Figura 5. En la etapa de soplado, el acceso a la infraestructura de la máquina es complejo y puede resultar peligroso para la implementación de un sistema de medición; adicionalmente, el trabajo en esta sección puede obligar a la detención de la producción o poner en riesgo la integridad del producto.

Además, debido a la necesidad de llevar el control del desperdicio generado, deben existir sistemas de medición tanto antes como después del corte de las cuchillas. Justo sobre el rodillo de presión, se cuenta con el espacio disponible para adjuntar un sistema de adquisición de datos, el sitio es de fácil acceso, la película plástica tiene una forma adecuada para su medición contraria a la forma tubular justo después del soplado, y la geometría tanto del espacio como del producto es muy similar antes y después del corte por lo que la naturaleza de los 3 sistemas de medición de ancho puede ser replicada.

### **Definición del problema relacionado a la medición del ancho**

Actualmente, la medida del ancho en la extrusora E16 es realizada manualmente por los operadores con cinta métrica, esto introduce una gran cantidad de errores en la medición: error de paralaje, error de repetitividad, pues, el factor humano

provoca la falta de constancia en la medición, error de *offset* pues el operador reportará este dato (en caso de ser necesario únicamente) de manera atrasada por lo que incluso podría introducir error al recordar la medida.

Además, actualmente no se cuenta con una base de datos que almacene las medidas de manera automática para así aplicar la Formula de Producción, ni tampoco con un sistema que alerte a la gerencia de producción en caso de que el ancho se salga de los límites establecidos.

Para el diseño de un sistema de adquisición de ancho se presentan dos problemas principales que es necesario resolver si se quiere determinar el ancho de manera precisa: la detección de los bordes de la película plástica y la adquisición de la posición de estos bordes el uno respecto del otro.

Próximamente, se procederá a bosquejar las posibles propuestas que pueden resolver estos dos problemas, se hará una comparación cualitativa entre ellas con una tabla de ponderados y una vez se haya decidido cual sea la mejor propuesta, se hará una comparación cuantitativa entre los elementos que se requieran para satisfacer la ejecución de la propuesta finalista.

### **Generación y descarte de la mayor cantidad de soluciones para el sistema mecánico de medición del ancho.**

Como se determinó anteriormente, las dos variables principales por medir son la presencia del plástico y, por consiguiente, el borde de la filmina; y la distancia entre estos dos bordes. Estas dos variables deben medirse en conjunto para poder extraer esta información, comunicarla e interpretarla.

Los criterios cualitativos que se utilizarán para clasificar las propuestas serán:

- Ajuste al proceso productivo: para ello se tomarán en cuenta el tamaño y la naturaleza de la medición en caso de requerir contacto o la modificación del producto. Se le asignará un peso de 3 puntos.
- Precio. Se le asignará un peso de 2 puntos.
- Posible carga computacional: algunos sistemas de adquisición de información requieren altas cargas computacionales para la interpretación de imágenes y demás; por lo que se preferirán aquellas propuestas que puedan satisfacer la medición de las variables con el menor costo computacional. Se le asignará un peso de 1 punto. En el caso de la carga computacional, un mayor puntaje significa una menor carga

La asignación de las notas ponderadas será según el número de propuestas de acuerdo con el análisis cualitativo y la menor calificación en cada criterio será 1.

Para calcular la nota ponderada se aplicará la siguiente fórmula:

$$Ponderado = \frac{A x^3 + P x^2 + C x + 1}{3!} \quad (2.4)$$

Donde A es el ajuste, P es el precio y C es la carga computacional.

### **Sistema matricial para detección de bordes**

Este sistema funcionaría de manera similar al sistema de detección de la velocidad propuesto para el objetivo anterior, para ello es necesario contar con un sensor de video especializado para cada uno de los sistemas de medición de ancho que estén sobre una base a una distancia invariable del plástico.

Es necesario tomar en cuenta que la distancia debe de permitir la recolección de la imagen de manera que se evite la mayor cantidad de aberraciones visuales posibles por lo que podría llegar a ser considerable y en la Extrusora 16 el espacio es limitado; para evitar este problema se podrían utilizar dos cámaras, sin embargo, como se discutió en las secciones anteriores, el costo puede ser muy alto y la carga computacional también.

Su precio ronda los \$180 [18].



Figura 42. Sensor matricial 11484. Fuente: adaptado de [18].

## **Sensor lineal CMOS**

De manera similar a una de las opciones para el sistema de velocidad, se puede utilizar este tipo de sensores para detectar bordes en una línea de píxeles, sin embargo, esta línea debe ser suficientemente clara como para que pueda ser detectada por el sensor.

Igualmente requiere de una posición fija y con una distancia que permita evitar cualquier tipo de aberración visual. Si embargo debido a su especialización, este sistema podría ser más conveniente que el anterior.

Su precio sobrepasa los \$1000 [14].

Ver Figura 32.

## **Sensor montado en carrito movido por motor a pasos**

Al tener control sobre la velocidad y los pasos del motor de una manera precisa, se puede utilizar la relación de conversión de revolución del eje a distancia lineal, para así obtener la magnitud del desplazamiento de un punto a otro con precisión milimétrica. Dicha propiedad de los motores a paso se ajusta de manera adecuada a los requerimientos del presente proyecto pues brinda la posibilidad de una adquisición precisa con un tamaño reducido.

En el caso de interés, no se consideró la utilización de un servomotor pues a pesar de que por lo general tienen más torque, la aplicación presente tiene como criterio definitivo la precisión y la exactitud. La implementación de un código que permita una medición precisa del movimiento del motor se beneficiará del conteo discreto de los pasos realizados y no de una conversión indirecta del tiempo de ejecución de un pulso de amplitud modulada a su equivalente en metros recorridos según la rapidez angular del motor pues las conversiones y los cálculos pueden introducir errores innecesarios.

Posteriormente, se requiere de algún sensor que permita al sistema de adquisición de datos el encontrar fielmente la presencia del plástico y así, el sistema de control del movimiento del motor puede contar los pasos entre los límites del material.

Según las opciones en las tiendas CrCibernética [15] y MicroJPM [16] el precio de los motores a pasos ronda los \$17 y \$23.



Figura 43. Motor a pasos Nema 17. Fuente: tomado de [19].

Tabla 3. Comparación entre propuestas para el sistema de adquisición del ancho. Fuente: elaboración propia.

Propuesta	Ajuste al proceso de producción	Precio	Carga computacional	Puntuación
Sistema matricial	1	2	1	1,333
Sensor lineal CMOS	2	1	2	1,667
Motor a pasos	3	3	3	3,000

Según el análisis anterior y la comparación en la Tabla 3. La propuesta que se ajusta mejor al presente proyecto es la del sensor montado en un carrito movido por un motor a pasos.

Una vez decidido el sistema que se utilizará, el siguiente paso lógico es decidir el tipo de sensor que marcará el límite de la película plástica.

### Criterios para la selección del sensor para el sistema de adquisición de ancho

- El sensor para el sistema de adquisición de datos del ancho debe de funcionar con el menor contacto posible con la película del producto.
- En caso de ser un sensor sin contacto, debe de ofrecer una incertidumbre menor a medio centímetro.
- La señal entregada debe de ser lo más discreta posible para evitar errores por comparación analógica.
- La alimentación del sensor no necesariamente estará restringida a un voltaje específico, sin embargo, es preferible que sea cercano a 5V estándar.

- La complejidad de la señal debe de ser lo más simple posible.
- La interpretación de la señal debe de ser lo más sencilla posible.

### **Búsqueda de información para la selección del sensor para el sistema de medición de ancho**

Para la selección del sensor que se utilizará en el sistema de medición del ancho se tomarán en cuenta las hojas de datos disponibles y preferiblemente la disponibilidad local de dichos sensores. Los datos que se tomarán en cuenta para la selección de dicho sensor serán:

- Tiempo de respuesta: la medida precisa y exacta del ancho es de vital importancia por lo que el menor tiempo de respuesta es deseable.
- Histéresis: la información adquirida puede estar en función al flanco medido de una señal cambiante por lo que preferiblemente se seleccionará un sensor con una magnitud de histéresis pequeña para evitar la necesidad de una corrección indirecta de esta propiedad.
- Ruido: la menor cantidad de ruido es deseable para poder utilizar fielmente la señal generada, tratando, así mismo, de minimizar fuentes de error.
- Propiedades eléctricas: serán preferibles los sensores que funcionen dentro de los voltajes estándares de tecnología TTL, es decir, 5 V y 3,3 V.
- Costo: se intentará disminuir el costo económico del sistema de adquisición sin que esto sacrifique el desempeño del sistema.
- Disponibilidad local: para optimizar la utilización del tiempo se preferirán aquellos sensores que estén disponibles localmente.

Si alguno o varios de estos criterios no se encuentran disponibles en las hojas de datos, se tomará una decisión previa y se procederá a la realización de pruebas experimentales.

### **Generación y descarte de la mayor cantidad de soluciones para el sensor en el sistema de medición de ancho**

Cualquier sensor de tamaño reducido que permita detectar una interrupción puede resultar útil para el sistema de medición del ancho.

Para la decisión sobre este sensor se utilizarán los siguientes criterios:

- Ajuste al proceso productivo: esta categoría abarca la forma en que mide, y la distancia posible entre el sensor y el objeto. Se le asignará un peso de 5 puntos sobre el ponderado.

- Totalmente inadecuado: se le asignará una nota de 0 puntos cuando requiera el contacto directo con la película plástica.
- Parcialmente inadecuado: cuando la distancia de medición sea fija. Se asignará una nota de 1 punto.
- Parcialmente adecuado: cuando la distancia de detección sea variable pero no se pueda tener control sobre ella. Se asignará una nota de 2 puntos.
- Totalmente adecuado: cuando la distancia sea variable y se pueda controlar de acuerdo con las propiedades eléctricas del sensor. Se asignará una nota de 3 puntos.
- Tamaño: es necesario mantener el tamaño más reducido posible sin afectar el funcionamiento. Se le asignará un peso de 4 puntos sobre el ponderado.
  - Se ordenará de menor a mayor.
- Precisión (e histéresis): algunos sensores tienen un cono de detección amplio debido a la naturaleza de su medición. Cuando no se tenga el detalle completo, se asumirá que el cono de medición a cualquier distancia es la medida mayor de la superficie del o de los elementos activos del sensor. La histéresis influye también en la precisión direccional que tendrá el sensor. Se le asignará un peso de 3 puntos.
  - Se ordenará de menor a mayor.
- Disponibilidad: se ordenará de menor a mayor de acuerdo con el catálogo en línea de los proveedores CrCibernética [15] y MicroJPM [16]. Se le asignará un peso de 2 puntos sobre el ponderado.
- Precio: Se ordenará de menor a mayor. Se le asignará un peso de 1 punto sobre el ponderado.

Nota: al no tener una medida oficial de la incertidumbre y del punto exacto donde el sensor detecta, se asume el peor de los casos donde la detección de la presencia del plástico puede ocurrir en cualquier punto sobre la superficie activa del sensor.

La fórmula para calcular el ponderado es similar a las utilizadas en apartados anteriores:

$$Ponderado = \frac{A \times 5 + T \times 4 + H \times 3 + D \times 2 + P \times 1}{5!} \quad (2.5)$$

Donde  $A$  es la puntuación según el ajuste al proceso productivo,  $T$  es la nota asignada al tamaño,  $H$  es la puntuación asignada a la precisión e histéresis,  $D$  es el puntaje asignado a la disponibilidad y  $P$  es la nota asignada al precio.

### Sensor capacitivo LJC18A3-B-Z/BX

Según su hoja de datos [20], el tamaño del sensor es de 68 mm de largo por un ancho mayor de 29 mm y con un área de sensado de 24mm. Su rango de detección es de 10 mm. No especifica los voltajes de operación ni la naturaleza de su histéresis o el cono de medición; de manera extraoficial, la página en línea de CrCibernética [15] reporta un voltaje de operación desde los 6 V a los 36 V.

- Su distancia de medición es variable pero no se especifica variabilidad por lo que el puntaje asignado será de 2 puntos.
- Por tamaño se asignará un puntaje de 1 punto.
- A su precisión se le dará una incertidumbre de  $\pm 24$  mm, por lo que su puntaje será de 1.
- Según [15] cuentan con 10 unidades y en [16] cuentan con 7 unidades. Su puntuación será de 1 punto.
- Precio: el precio es de \$17,95 según [16] y de \$9,95 según [16]. Su puntuación será de 1 punto.

Ver Tabla 4.



Figura 44. Sensor capacitivo LJC18A3-B-Z/BX. Fuente: adaptado de [16]

### Sensor resistivo FSR 402

Según [21]:

- Su diámetro es de 18,28 mm. Su puntuación será de 2 puntos.
- Su histéresis es del 10% y requiere de 0,1 N de fuerza para la detección. Esto es una presión de 381 Pa. Su puntaje de adaptabilidad será de 0.



- Debido a que el ancho del área activa del sensor es de 18,28 se asignará una incertidumbre de  $\pm 18,28$  mm por lo que su puntuación es de 1 punto.
- Su rango eléctrico se puede ajustar con el uso de un amplificador operacional. Su puntuación será de 2 puntos.
- El precio en [15] es de \$7,95 y en [16] de \$9,90. Su puntuación será de 2 puntos.
- Al momento de la selección en [15] se tienen 17 en existencias y en [16] se tienen 4 en existencias. Su puntuación será de 2 puntos.

Ver Tabla 4.



*Figura 45. Sensor resistivo FSR 402. Fuente: adaptado de [15].*

### **Sensor Reflectivo CNY70**

Este sensor consiste en un LED infrarrojo a un lado del centro homólogo a un fototransistor del lado contrario.

Según su hoja de datos [22] la corriente del colector varía según la distancia que puede ser entre 0 y 10 mm. Esta corriente puede ser modificada con el voltaje al con que se alimente. La hoja de datos también ofrece la especificación de su cono de detección de acuerdo con la corriente de colector. El máximo ángulo de detección es de 50°.

Sus dimensiones son de 7 x 7 x 12mm y el área activa es de 2,8mm x 1,8mm por lo que la incertidumbre máxima a los 0° de detección será de 1,8mm según la orientación escogida.

En la página de CrCibernética [15] se tienen 79 en existencias a un precio de \$1,79 y en la página de MicroJPM [16] se cuenta con 40 unidades a un precio de \$1,50.

- Su puntuación en adaptabilidad será de 3 puntos.
- Su puntuación en tamaño será de 3 puntos.
- Su puntuación en precisión será de 3 puntos.
- Su puntuación en tamaño será de 3 puntos.

- Su puntuación en disponibilidad será de 3 puntos.

Ver Tabla 4.

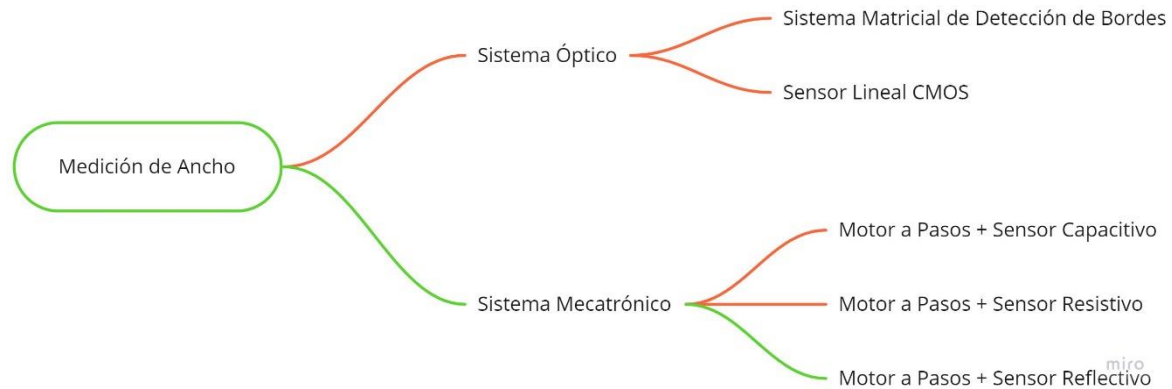


Figura 46. Sensor CNY70. Fuente: adaptado de [22].

Tabla 4. Comparación de los sensores posibles para el sistema de medición de ancho. Fuente: elaboración propia.

Propuesta	Ajuste al proceso de producción	Tamaño	Precisión	Disponibilidad	Precio	Puntuación
Sensor Capacitivo	2	1	1	1	1	1,333
Sensor Resistivo	0	2	1	2	2	1,133
Sensor Reflectivo	3	3	3	3	3	3,000

Según la discriminación anterior, se puede escoger con confianza el sensor reflectivo CNY70.



*Figura 47. Flujo de oportunidad para la selección del sistema de adquisición de datos del ancho. Fuente: propia.*

## Sobre los materiales

Debido a que la eventual construcción de este dispositivo está sujeta a la disposición de materiales de parte de proveeduría en la compañía, se dará prioridad a aquellos que se tengan disponibles en el momento del diseño siempre y cuando las propiedades mecánicas de dichos materiales no pongan en peligro la integridad del conjunto ni comprometan el desempeño del sistema y su funcionamiento.

## Búsqueda de información pertinente para la decisión de los materiales a utilizar para la construcción del montaje mecánico del sistema de adquisición de la velocidad y adquisición de ancho.

Las principales características para tomar en cuenta en los materiales son:

- Dureza: es importante evitar que se generen concentradores de esfuerzo en las piezas que se instalarán en la máquina pues el entorno puede ser hostil en algunas ocasiones y puede estar sujeto a golpes por parte de los operadores.
- Módulo de elasticidad: la resistencia a la compresión, a la tensión y al esfuerzo cortante asegurarán que los elementos mecánicos se mantengan íntegros y funcionales durante el tiempo de funcionamiento del sensor.
- Resistencia a la fatiga: en caso de aquellos elementos que puedan ser móviles la presencia de esfuerzos cíclicos y variables podría comprometer su funcionamiento por lo que puede ser útil tener esta información a la mano.
- Disposición del material: lógicamente se tomarán en cuenta aquellos materiales de los que se tenga disposición de manera inicial por parte de la empresa. En caso de que las especificaciones de dichos materiales no

cumplan con lo adecuado, se seguirá un proceso iterativo sobre el diseño para la selección de materiales distintos.

La selección de candidatos se hará para los elementos cruciales de los sistemas; en el caso del sistema de velocidad, para los brazos, y en el caso del sistema de ancho, para las vigas que soportan el carrito y el ensamblaje completo. Las presentaciones disponibles de barras metálicas en el inventario de artículos de Polymer que pueden ser utilizadas tanto para el sistema de adquisición de velocidad como para el sistema de medición de ancho son:

- Barra de acero 4140 de 1 pulgada de ancho: a un precio de 7118 colones por metro.
- Barra de acero 4140 de ½ pulgada de ancho: a un precio de 4049 colones por metro.
- Barra de aluminio de 1 pulgada: a un precio de 9124 colones por metro.
- Barra de aluminio de ½ pulgada: a un precio de 8600 colones por metro.

Según [42] las propiedades del acero son:

- Densidad:  $7,85 \text{ g/cm}^3$
- Módulo de fluencia: alrededor de 415 MPa.
- Dureza Brinell: 197.

Según [68] las propiedades del aluminio son:

- Densidad:  $2,7 \text{ g/cm}^3$
- Módulo elástico: alrededor de 290 MPa [69].
- Dureza Brinell: 96.

Si se aplica la teoría del Esfuerzo Cortante Máximo [5]:

En el caso del acero, una barra de 1/2 pulgada puede soportar un peso de hasta 2190 kg concentrados en un punto antes de la falla por cortante con un factor de seguridad de 1,2; y una barra de 1 pulgada puede soportar un peso de hasta 8761 kg concentrados en un punto antes de la falla por cortante con un factor de seguridad de 1,2.

En el caso del aluminio, una barra de 1/2 pulgada puede soportar un peso de hasta 1530 kg concentrados en un punto antes de la falla por cortante con un factor de seguridad de 1,2; y una barra de 1 pulgada puede soportar un peso de hasta 6123 kg concentrados en un punto antes de la falla por cortante con un factor de seguridad de 1,2.

Para ambos sistemas, se puede asumir con seguridad que ningún elemento alcanzará ese peso por lo que el módulo de fluencia no parece ser un factor determinante.

Para el sistema de adquisición de velocidad, es seguro asumir que el tamaño será muy pequeño por lo que el peso del ensamblaje no será de mucho cuidado, por lo tanto, el acero, al ser una opción más barata y con una dureza mayor, se convierte en el mejor candidato.

Para el sistema de adquisición del ancho, debido al comportamiento cíclico del sistema, es importante tomar en cuenta la dureza debido a que entre más dura sea la superficie de un material más frágil es este [5] por lo tanto; además, la complejidad del montaje seguramente será considerablemente mayor por lo que es conveniente ahorrar peso en el ensamblaje completo, por lo tanto, para las barras que soportan el carrito, el primer candidato será el aluminio y para las barras que soportan el ensamblaje, el primer candidato será el acero de 1 pulgada para dar mayor rigidez.

Es importante notar que aún no se han realizado cálculos de momento flector pues no se tienen las dimensiones finales de los componentes y esto podría llegar a ser un dato importante para reconsiderar la selección al igual que el posible pandeo de las barras y la interacción con los rodamientos.

### **Sobre la adquisición de datos**

Como se ha discutido hasta el momento, hay diferentes variables que se deben controlar y medir; los sensores por sí solos, únicamente proporcionan las señales eléctricas que pueden ser interpretadas y medidas, sin embargo, para que estas señales tengan sentido, es necesario una unidad de micro procesamiento y micro control.

Se podrá partir de las decisiones tomadas en las etapas anteriores para, asignar los requerimientos del sistema de adquisición de datos, así mismo para la observación de algún circuito en específico, necesario para el tratamiento de las señales, antes de que sea adquirida por el microprocesador.

Hasta el momento, los elementos del sistema de medición general que han sido seleccionados son:

- Sensor de efecto Hall [9]:
  - Este sensor puede ser alimentado con un voltaje desde los 5V hasta los 24V.
  - Se comporta como sensor de estado sólido (digital).
  - Tiene una frecuencia de operación máxima de 500 KHz.

La señal del sensor de efecto Hall A3144, al ser digital, no necesita ser filtrada o adecuada para su medición; puede proporcionar una señal de interrupción fiel.

- Motor a pasos Nema 17 [19]:
  - Tiene una resolución de 200 pasos por vuelta.

- Tiene 4 fases.
- La corriente máxima admitida es de 1,33 A.

Las señales para su control son digitales sin embargo la corriente necesaria es considerable por lo que es recomendable la utilización de un controlador para motor a pasos. En [15] se encuentra disponible el controlador digital para motor a pasos EasyDriver a un precio de \$9,95 que está basado en el circuito integrado A3967 y según su manual de usuario [23] sólo necesita de 2 entradas con voltaje a 5V para controlar el motor en su totalidad. Por lo que corresponde a una excelente opción para el control del motor.

En [15] también se encuentra la opción de Driver TB6560, sin embargo, el precio es de \$15 y su funcionamiento es muy similar.

- Sensor reflectivo con salida de transistor, CNY70 [22]:
  - Se puede alimentar con 5V.
  - Su señal es analógica.

Tomando en cuenta la naturaleza de las señales y los voltajes de alimentación de los elementos ya mencionados se puede sentar una base para la selección de los componentes de control.

### **Generación y descarte de la mayor cantidad de soluciones para el microcontrolador.**

Continuando con la metodología utilizada hasta el momento para la selección de los componentes anteriores, también se utilizarán condiciones y criterios definidos para la decisión definitiva:

- Ajuste a las condiciones preestablecidas por las selecciones anteriores (se asignará un peso de 5):
  - No se ajusta: cuando esta sea la condición se asignará 0 puntos.
  - Se ajusta parcialmente: cuando esta sea la condición se asignará 1 punto.
  - Se ajusta totalmente: cuando esta sea la condición se asignarán 2 puntos.
- Flexibilidad en el manejo del tiempo: este criterio determinará, qué tan favorable resulta su implementación si se toman en cuenta factores como la familiaridad del diseñador con dicho elemento, flexibilidad respecto a su programación y disposición de las herramientas para su programación. El puntaje se asignará de mayor a menor de acuerdo con cuál candidato sea el más adecuado para el cumplimiento oportuno de los objetivos.

Resulta de gran importancia señalar que esta consideración es necesaria pues, el tiempo de desarrollo disponible es limitado y todo aquello que facilite el diseño

rápido es preferible; siempre y cuando no se sacrifique el desempeño y el cumplimiento de los requisitos del sistema. A esta categoría se le asignará un peso de 4.

- Capacidad de procesamiento y funciones inherentes: los microprocesadores ofrecen funciones importantes tales como temporizadores, interrupciones, entre otros. La puntuación se asignará de mayor a menor donde la mayor puntuación será el número de candidatos menos uno y la menor puntuación será 0 puntos. A este criterio se le asignará un peso de 3.
- Cantidad de entradas disponibles: la consideración de esta condición favorece la flexibilidad del diseño en sus etapas futuras; se ordenarán de mayor a menor donde la mayor puntuación será el número de candidatos menos uno y la menor puntuación será 0 puntos. A este criterio se le asignará un peso de 2.
- Precio: finalmente se ordenarán las opciones de acuerdo con el precio. En esta categoría también se tomarán en cuenta aquellos equipos que deban ser adquiridos para su programación y funcionamiento siempre y cuando no se tengan previamente. A esta categoría se le asignará un peso de 1.

### **Candidatos para la unidad de micro procesamiento**

Debido a la prontitud y facilidad con la que pueden ser adquiridos estos elementos en sus plataformas, se le dará prioridad a aquellos que se encuentren disponibles en el catálogo de productos de CrCibernética [15] y MicroJPM [16].

#### **PIC4550**

Actualmente con 6 en existencia únicamente en MicroJPM, tiene un precio de \$12,95 [16].

Según su hoja de datos [24]:

- Su frecuencia de operación interna es variable y puede ir desde los 8 MHz hasta los 31kHz, sin embargo, cuenta con la posibilidad de adjuntar un oscilador externo para aumentarla hasta los 40 MHz.
- Cuenta con 4 temporizadores internos.
- Cuenta con 2 módulos de captura de interrupción.
- Tiene la opción de hasta 3 interrupciones externas.
- Soporta comunicación USB, SPI y I2C.
- Su encapsulado tiene 40 pines.
- Para su programación se requiere del MPLAB PICkit 3 que está disponible en CrCibernética [16] con 2 en existencia a un precio de \$29,95.



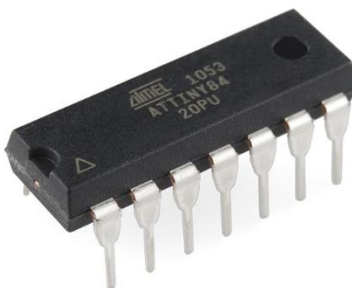
*Figura 48. PIC4550. Fuente: adaptado de [16]*

## **ATTINY84**

Únicamente ofrecido por la tienda MicroJPM [16] a un precio de \$8,25 sin embargo no está disponible al momento de la redacción.

Según su hoja de datos [25]:

- Cuenta con un temporizador de 8 bits y otro de 16 bits.
- Ofrece hasta 12 interrupciones externas y por cambio de estado.
- Ofrece comunicación USI (Interfaz Serial Universal por sus siglas en inglés). Esta interfaz es síncrona por lo que requiere de una señal de reloj común entre los elementos activos de la comunicación.
- Puede operar entre los 0 MHz y los 20 MHz.
- Se puede programar con la plataforma de desarrollo Arduino de la cual ya se cuenta con 2 unidades.
- Su encapsulado tiene 14 patillas.



*Figura 49. ATTINY84. Fuente: adaptado de [16]*

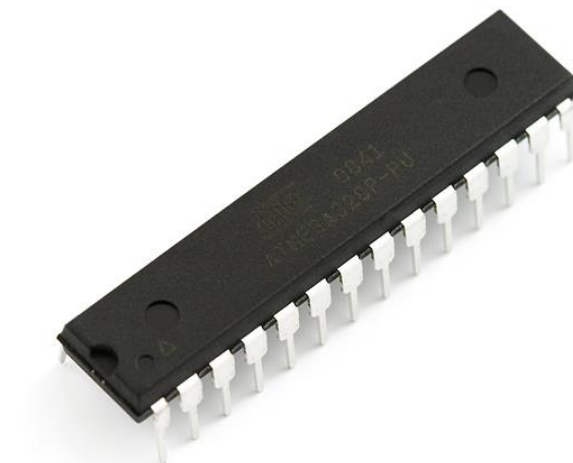


## ATmega328 PU

Está disponible en CrCibernética [15] a \$6,95 y en MicroJPM [16] a \$5,95.

Según su hoja de datos [26]:

- Cuenta con dos temporizadores de 8 bits y uno de 16 bits.
- Puede operar hasta los 20 MHz.
- Ofrece 2 pines con interrupción por estado externo y 23 por cambio de estado de pin.
- Ofrece comunicación USART (Receptor-Transmisor Sincrono-Asíncrono Universal). Al ser asíncrona no necesita una señal de reloj común entre los elementos que se comunican.
- Se puede programar con la plataforma de desarrollo Arduino de la cual ya se cuenta con 2 unidades.
- Su encapsulado tiene 28 patillas.



*Figura 50. ATmega 328 PU. Fuente: adaptado de [15]*

Adicional a lo mencionado anteriormente, ya se tiene amplia experiencia con la plataforma de desarrollo Arduino y con la operación del microprocesador ATmega 328 PU

Según los criterios:

- Los 3 candidatos se ajustan a las necesidades y a las condiciones preestablecidas por las selecciones anteriores pues los tres ofrecen

interrupciones externas, comunicación serial y temporizadores que permitirían la adquisición y utilización de las señales de los sensores.

- En el grado de familiaridad y flexibilidad con respecto al manejo del tiempo, se prefiere en primer lugar el ATmega 328PU pues ya se tiene amplia experiencia en el uso de este microprocesador y la comunicación USART resulta muy cómoda al ser asíncrona pues reduce la complejidad de esta. En segundo lugar, el ATtiny 84 pues comparte entorno de desarrollo con el ATmega 328 PU y por último el PIC4550.
- En capacidad de procesamiento en primer lugar se ubica el PIC4550 debido a que su frecuencia posible es mayor y ofrece 4 temporizadores, en segundo lugar, se ubica el ATmega328 PU con 20 MHz de frecuencia y 3 temporizadores y por último el ATtiny 84 pues, aunque tiene la misma capacidad en frecuencia sólo cuenta con 2 temporizadores.
- Para la cantidad de entradas disponibles, es de gran importancia la presencia de interrupciones pues de esta manera se puede llevar un control exacto de los eventos, en este caso, el mayor puesto lo tiene el ATmega 328 PU con 23 posibles interrupciones, el segundo puesto lo tiene ATtiny 84 con 12 interrupciones y por último el PIC4550 con 3 interrupciones.
- Respecto al precio, el primer lugar lo ocupa el ATmega 328 PU a \$6,95 y \$5,95, el segundo lugar lo ocupa el ATtiny 84 a un precio de \$8,25 y por último, el PIC4550 con un precio de \$12,95, a esto se le debe sumar el PICKit 3 por \$29,95.

La fórmula utilizada para el cálculo de la puntuación ponderada es la siguiente:

$$Ponderado = \frac{A \times 5 + F \times 4 + C \times 3 + E \times 2 + P \times 1}{5!} \quad (2.6)$$

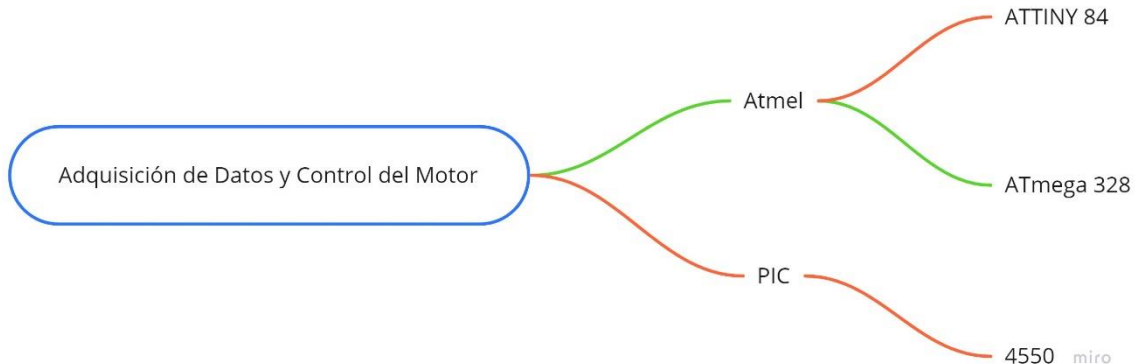
Donde *A* es el grado de ajuste a las condiciones preestablecidas, *F* es el grado de flexibilidad respecto al tiempo, *C* es la capacidad de procesamiento, *E* es la cantidad de entradas disponibles y *P* es el precio.

En la Tabla 5. se puede observar la comparación entre los candidatos para la recolección de datos y el control del motor.

*Tabla 5. Tabla de comparación de candidatos para la adquisición de las señales y el control del motor a pasos. Fuente: elaboración propia.*

Propuesta	Ajuste a las selecciones previas	Flexibilidad respecto tiempo	al Capacidad de Procesamiento	de Cantidad de entradas	de Precio	Puntuación
PIC4550	2	1	3	1	1	1,733
ATtiny 84	2	2	1	2	2	1,800
Atmega 328PU	2	3	2	3	3	2,467

Según los criterios de selección y la fórmula aplicada, la opción más oportuna es el ATmega 328PU



*Figura 51. Flujo de oportunidad para la selección del microprocesador. Fuente: elaboración propia.*

### **Candidatos para el Sistema de Alertas Visuales y la Comunicación de Datos en Línea**

Debido a la naturaleza visual de esta etapa, se requiere un computador que pueda soportar una interfaz gráfica y dé soporte a los distintos lenguajes de programación que ofrecen esta posibilidad.

Adicionalmente, la plataforma que se vaya a utilizar debe poder comunicarse a internet.

Debido a la gran variedad de computadores disponibles, con distintos sistemas operativos, los criterios para descartar la potencial utilización de cada uno serán:

- Precio: las ofertas de sistemas de computación embebidos varían grandemente entre ellas. A esta categoría se le asignará una puntuación de 5 sobre el ponderado.
- Tamaño: debido al espacio limitado disponible en las proximidades de la máquina, se debe implementar el sistema más pequeño posible. Se asignará un puntaje de 4 sobre el ponderado.
- Compatibilidad: esta categoría comprende asuntos de licencias como de compatibilidad nativa con controladores y software necesario.
- Flexibilidad de implementación: esta categoría está relacionada al tamaño del sistema como tal, sin embargo, contempla la cantidad de accesorios, equipos que se puedan adjuntar a su funcionamiento y entradas y salidas lógicas.
- Disponibilidad: a esta categoría no se le asignará puntaje por lo tanto no estará incluida en la tabla de comparación; si alguno de los candidatos no

está disponible para un tiempo de máximo un mes a partir del inicio de la ejecución del proyecto, se descartará.

## **Raspberry Pi 4**

La Raspberry Pi 4 [59] es una computadora de escritorio con sistema operativo Raspbian [60] basado en Linux [61], sus especificaciones técnicas son:

- Procesador de 4 núcleos a 1.5 GHz.
- 2 GB de memoria RAM.
- Ethernet Gigabit.
- Bluetooth 5.0.
- Wifi.2
- Alimentación 5 V, 3 A.
- No incluye monitor: 2 puertos HDMI para pantallas.
- 2 puertos USB 3.0 y 2 puertos USB 2.0
- No incluye disco duro: puede utilizar memoria microSD de hasta 32 GB.

Debido a que su sistema operativo está basado en Linux, ofrece soporte a gran cantidad de controladores para dispositivos tanto originales como genéricos.

Tiene dimensiones de 85,6 mm x 56,5 mm sin la caja protectora por lo que representa un sistema muy compacto.

A pesar de que no incluye monitor en su paquete, el cable HDMI permite utilizar cualquier tipo de monitor de cualquier tamaño disponible en el mercado.

El paquete que incluye el cable de alimentación, el cable HDMI, una tarjeta microSD de 16 GB y la caja protectora está a un precio de 109,95 USD en la tienda en línea CrCibernética [15].

## **ASUS X515MA**

Debido a la alta cantidad de propuestas en la plataforma Windows, se propondrá este modelo debido a ser uno de los ordenadores de precio más cómodo en el país.

Sus prestaciones son:

- Pantalla de 15 pulgadas.
- Memoria RAM de 4 GB.
- Unidad de estado sólido de 128 GB.
- Bluetooth.
- Wifi.

- Sistema operativo Windows 10.
- Cuenta con 3 puertos USB y un puerto HDMI.

En la página web de Extreme Tech [62] su precio es de 205000 colones.

Debido al tamaño de su pantalla, es lógico asegurar que su tamaño es un tanto mayor a las 15 pulgadas de diagonal.

Al tener sistema operativo Windows, la compatibilidad se puede ver comprometida con componentes electrónicos genéricos.

Cuenta con un solo puerto HDMI por lo que únicamente se puede adjuntar una pantalla adicional.

## **Chromebook**

En la página de la tienda Siman [63] se encuentra disponible a un precio de 169900 colones.

Sus especificaciones técnicas son:

- 32 GB de almacenamiento.
- 4 GB de memoria RAM.
- Pantalla de 12 pulgadas.
- Bluetooth y WiFi
- Sistema operativo: Chrome OS.
- Cuenta con 2 puertos USB y un puerto HDMI.
- Monitor de 12 pulgadas.

Sus dimensiones son 11,7 x 8,3 pulgadas.

Al tener sistema operativo Chromebook OS, la compatibilidad de controladores e implementación de distintas rutinas de programación puede resultar más complicada que con los otros candidatos.

## **Análisis de las Propuestas para el Sistema de Alertas Visuales y la Comunicación de Datos en Línea**

- Precio: según el orden de precios, de menor a mayor, los puntajes asignados serán:
  - Raspberry Pi 4, 3 puntos.
  - Chromebook, 2 puntos.
  - ASUS X515MA, 1 punto.
- Tamaño: según el orden de tamaños, de menor a mayor:
  - Raspberry Pi 4, 3 puntos.

- Chromebook, 2 puntos.
- ASUS X515MA, 1 punto.
- Compatibilidad: según su sistema operativo.
  - Raspberry Pi 4, 3 puntos.
  - ASUS X515MA, 2 puntos.
  - Chromebook, 1 punto.
- Flexibilidad: por sus entradas y salidas.
  - Raspberry Pi 4: 4 puertos USB, 2 puertos HDMI y 40 pines E/S; 3 puntos.
  - ASUS X515MA, 2 puntos.
  - Chromebook, 1 punto.
- Disponibilidad:
  - Según las páginas [15], [62] y [63] todos los equipos están disponibles.

Tabla 6. Cálculo de la puntuación ponderada para los candidatos del Sistema de Alertas Visuales y Comunicación de Datos en Línea. Fuente: elaboración propia.

Candidato	Precio	Tamaño	Compatibilidad	Flexibilidad	Ponderado
Raspberry Pi 4	3	3	3	3	3
Asus X515MA	2	1	2	2	1,714285714
Chromebook	1	2	1	1	1,285714286

Gracias a la comparación anterior, se puede entender que el candidato más apropiado para la ejecución de esta etapa es la Raspberry Pi 4.



Figura 52. Flujo de oportunidad para la selección del centro de la computadora. Fuente: Elaboración propia.

## **Sobre el Registro de Datos en Línea**

La última etapa crucial para el desarrollo del proyecto es el almacenamiento de los datos recopilados en una plataforma en línea, esta plataforma debe permitir la creación de tablas en libros de cálculo, preferiblemente Excel 365 (pues en la compañía ya se cuenta con las licencias necesarias); además debe de ser capaz de recibir los datos enviados por la interfaz al menos cada 5 minutos. Considerando el factor tiempo, debe de ser lo más amigable e intuitiva posible tanto para la implementación como para la utilización periódica por parte de la persona encargada.

Los criterios principales para la selección o el descarte de los candidatos son:

- Precio: será el factor principal para evaluar los candidatos propuestos. A este criterio se le asignará un peso de 3 en el ponderado.
- Facilidad de implementación: se asignará tomando en cuenta el tiempo estimado para su implementación y qué tan amigable sea la plataforma con el usuario. A este criterio se le asignará un peso de 2 en el ponderado.
- Funcionalidad: de acuerdo con las funciones nativas de la plataforma; se le asignará un peso de 1 sobre el ponderado.

## **Propuesta de Candidatos para el Registro de Datos en Línea**

### **Amazon Web Services**

Amazon Web Services es una plataforma completa de servicio en la nube que permite la implementación de bases de datos, seguridad cibernética, ofrece plataformas para creación de aplicaciones en la nube, etc. [64].

En cuanto al precio, su costo se calcula por el uso al mes y por el tamaño de las cargas utilizadas, una plataforma básica para aplicación puede tener un costo de hasta 800 USD [65] a esto debe sumarse el tiempo de capacitación de la persona que dará mantenimiento y el costo de implementación.

Debido a la alta flexibilidad de sus ofertas, la implementación de sus servicios requiere capacitación y una importante inversión en tiempo pues dicha implementación se realiza en varios niveles de servicio.

## **Google Cloud**

Es una plataforma en la nube que une todos los servicios que ofrece Google, desde flujo, seguridad y análisis de datos hasta soluciones de inteligencia artificial [66]

En lo que respecta a precios, la utilización de un solo servidor con un solo disco duro persistente puede tener un costo de hasta 98 USD [67].

Su implementación es de igual manera enfocada a plataformas a escala empresarial y no a escala local como lo demanda la presente aplicación; además, en el costo se debe tomar en cuenta el tiempo de implementación y la capacitación de los operadores de esta plataforma.

## **Integromat**

Integromat, es una plataforma simplificada de automatización enfocada en escenarios que utilizan API's para el cumplimiento de sus funciones. [29]

Su costo estimado al mes para la aplicación de interés es de aproximadamente 29 USD al mes.

Su implementación resulta intuitiva para el usuario por lo que no requiere capacitación y además ofrece implementación rápida con Excel 365 y otras aplicaciones como Telegram.

## **Descarte de Candidatos para el Registro de Datos en Línea**

Respecto al precio, la opción de mayor costo es Amazon Web Services, seguida por Google Cloud, por último, la de menor precio es Integromat por lo que sus puntuaciones son:

- Integromat: 3 puntos.
- Google Cloud: 2 puntos.
- Amazon Web Services: 1 punto.

En facilidad de implementación, la plataforma que resulta más amigable es Integromat, seguida por Google Cloud y por último Amazon Web Services. Sus puntuaciones son:

- Integromat: 3 puntos.
- Google Cloud: 2 puntos.
- Amazon Web Services: 1 punto.

La funcionalidad de Google Cloud y Amazon Web Services es claramente superior a la de Integromat, sin embargo, Integromat ofrece la integración de API's para uso fácil. Sus puntuaciones son las siguientes:



- Amazon Web Services: 3 puntos.
- Google Cloud: 3 puntos.
- Integromat: 2 puntos.

Tabla 7. Tabla de calificación ponderada para la selección de la plataforma de registro de datos. Fuente: elaboración propia.

Propuesta	Precio	Facilidad	Funcionalidad	Puntuación
Amazon Web Services	1	1	3	1,333
Google Cloud	2	2	3	2,167
Integromat	3	3	2	2,833

Por las razones anteriores, la opción que mejor se ajusta al presente proyecto es Integromat.

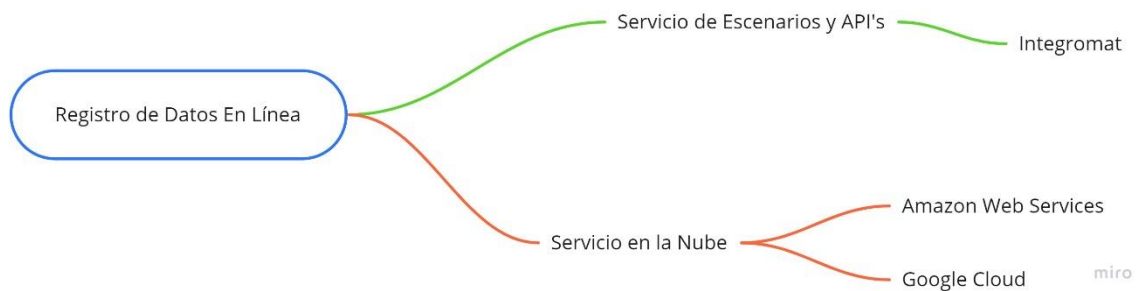


Figura 53. Flujo de oportunidad para la selección de la plataforma de registro de datos en línea. Fuente: elaboración propia.

## Criterios de Validación

En el análisis de resultados se debe poner a prueba la propuesta de diseño según criterios medibles que pongan en evidencia que las decisiones tomadas fueron las correctas.

## Sistema de Adquisición de Velocidad

Los criterios de validación para el sistema de velocidad serán:

- Que sea capaz de medir velocidades desde los 10 m/min, o a la velocidad mínima disponible al momento de la prueba, hasta los 150 m/min o hasta la velocidad máxima al momento de la prueba.

- Que sea capaz de detectar si la rueda se ha detenido por más de 4 segundos.
- Que sea capaz de transmitir los datos de velocidad a una frecuencia suficiente para que cada pulso recibido sea adecuadamente medido.
- Que su montaje no requiera la detención de la operación de la máquina.
- Que el funcionamiento de sus partes tenga una vida útil superior al tiempo en el que se recupera la inversión inicial.

### **Sistema de Adquisición de Ancho**

Los criterios para validar el funcionamiento del sistema de adquisición de ancho serán:

- Que sea capaz de medir filminas de hasta 150 cm de ancho.
- Que la desviación estándar sea menor a los 2 mm.
- Que sea capaz de detectar la presencia de todos los productos extruidos en la máquina E16.
- Que sea capaz de detectar la presencia del plástico a cualquier hora del día.

### **Sistema de Visualización, Alarma y Comunicación de Datos en Línea**

Los criterios de validación para el sistema de visualización, alarma y comunicación de datos en línea son:

- Que sea capaz de mostrar la información recibida de los sistemas de adquisición de datos con una resolución de al menos 2 dígitos decimales.
- Que sea capaz de alertar cuando el rodillo de prensa haya estado detenido por al menos 4 segundos.
- Que sea capaz de alertar cuando la medición del ancho ha salido por 0,5 cm o más del valor predeterminado.
- Que sea capaz de enviar los datos requeridos con una frecuencia de al menos cada 5 minutos.

## Capítulo 4: Propuesta de Diseño

### Diseño de un Sistema de Alertas Visuales y de Adquisición de Datos

La interfaz gráfica de la Extrusora 16 permite la visualización en tiempo real de los datos adquiridos por el sistema de adquisición de velocidad ubicado sobre el rodillo de presión, ver Figura 4.; y los tres sistemas de adquisición de datos de ancho de producto extruido, ubicados: el primero en el espacio previo al rodillo de presión, ver Figura 5., el segundo y tercero en las líneas de extrusión A y B respectivamente, ver Figura 6. Además, permite la introducción de los parámetros de producción: ancho inicial esperado (“Seteado”), ancho final esperado en A (“Seteado en A”), ancho esperado en B (“Seteado en B”), producción esperada (“Seteada”), densidad, espesor y presentación del producto.

También, permite la introducción de elementos logísticos como: nombre de operador, número de operador, nombre del producto, tipo de hora y la razón de detención de la máquina; también cuenta con la opción de la modificación de la configuración predeterminada de los márgenes de tolerancia para las medidas de los anchos y la producción esperada; calcula la cantidad de desperdicio esperado y real, la velocidad esperada con base a los parámetros esperados y la producción real. Además, representa visualmente con una alerta cuando alguno de los parámetros excede el margen de tolerancia.

Internamente, las funciones que controlan la interfaz se encargan de la recopilación real de toda la información disponible, del almacenamiento de esta información en la forma de valores CSV o valores separados por coma (por sus siglas en inglés) y de un registro textual de los eventos relacionados a las variables mencionadas anteriormente que toman lugar en la operación de la máquina. Adicionalmente, se encarga del envío de estos datos y alertas a través de internet hacia la plataforma Integromat para su posterior almacenamiento en un documento de Excel 365 y para las alertas de interés.

El entorno de desarrollo es Python (versión 3) orientado a su programación en Linux/Raspbian para Raspberry Pi 4 con la utilización de la biblioteca **guizero** [27] y **tkinter** [28].

### Interfaz

El manual técnico se redactará según 2 etapas principales: La interfaz y la programación. La presente sección es sobre la interfaz de usuario.

## Pantalla Inicial

En la Figura 54. se puede observar la pantalla inicial. Esta pantalla muestra toda la información que está siendo recopilada por el sistema de adquisición de datos además de las alertas por salida de márgenes o detención; adicionalmente muestra información logística relacionada a la naturaleza del producto, al desperdicio esperado en términos de centímetros de producto y el cálculo de la producción real.


EXTRUSORA E16					
	VARIABLES	SETEADO	REAL	DIFERENCIA	ALERTA
CONTROLADAS	ANCHO INICIAL (cm)	14	14.02	0.02	
	ANCHO FINAL A (cm)	13.5	14.02	0.52	
	ANCHO FINAL B (cm)	13.5	14.0	0.5	
	VELOCIDAD (m/min)	5.06	6.66	1.6	
NO CONTROLADAS	ESPESOR (mil/in)	0.52	DESPERDICIO AUTOMATICO	ESPERADO	REAL
	DENSIDAD (g/cm3)	0.923		1.0	0.02
PRODUCTO	DIGITE DATO		LAMINA	TUBO	
OPERADOR	Nombre:	**	kg/h	ESPERADO	REAL
	Numero:	**		1.0	1.36
TIPO DE HORA:		RAZON DE PARO:			09:34:57 AM

Figura 54. Pantalla Inicial de Interfaz de Usuario. Fuente: elaboración propia.

Los apartados de la Interfaz de Usuario de izquierda a derecha y de arriba a abajo son:

- El nombre de la máquina.
- El logo de Polymer S.A. que dobla su función como botón de acceso a la ventana de autorización para cambiar parámetros de tolerancia y valor esperado de la producción.
- El nombre de las columnas relacionadas a las 4 variables medidas:
  - Variables: nombre de la variable medida.
  - 'Seteado' [sic]: Es el valor de producción esperado. Para facilidad de comprensión, el supervisor de producción solicitó este anglicismo.
  - Real: En esta columna se encuentran los valores que están siendo medidos por los 4 sistemas de adquisición de datos.
  - Diferencia: Es la diferencia aritmética entre el valor real sustraído del valor esperado, en el caso de la velocidad, este valor es absoluto.

- Alerta: cuando los valores de diferencia estén por fuera del margen aceptable la celda se muestra roja sin parpadear.
- Leyendas de tipo:
  - Controladas: se refiere a las variables que están siendo monitorizadas periódicamente por el sistema de adquisición de datos. Vale aclarar que no existe un sistema de realimentación hacia la máquina extrusora por lo tanto no es un sistema de control automático.
  - No controladas: son los valores constantes que ha determinado el departamento de producción, el valor de espesor puede ser cambiado por el usuario y el valor de densidad puede ser cambiado en la ventana de configuración sin embargo su monitorización no está dentro del alcance del presente proyecto.
  - Producto: nombre del producto que está siendo extruido en el momento. Y si su presentación es laminar o tubular (esta selección influye directamente en la fórmula aplicada para el cálculo de la velocidad teórica). Cuando se selecciona la presentación tubular, los valores esperados del ancho final A y el ancho final B se configuran en 0 cm. Ver Figura 60.
  - Operador: en estas dos filas se introducen los datos relacionados al operador de turno.
- Fila de desperdicio: En esta fila se pueden leer los valores de desperdicio esperado y de desperdicio real. Estos dos valores son calculados internamente por el algoritmo.
- 'kg/h': ver Figura 57., se refiere a la fila de celdas relacionada a la producción esperada y a la producción calculada. La producción esperada se puede cambiar manualmente en la pantalla de configuración y la producción real es calculada automáticamente con la siguiente fórmula:

$$Producción \left[ \frac{kg}{h} \right] = \frac{Densidad \times Espesor \times Velocidad \times 1,524 \times [Ancho A + Ancho B]}{100} \quad (3.1)$$

- Tipo de Hora: ver Figura 55., esta selección solo estará disponible cuando el sistema de adquisición de velocidad esté recibiendo información relacionada al movimiento del rodillo y no esté detenido. Las dos distintas categorías disponibles son:
  - 'Set Up': cuando el operador se encuentra ajustando la extrusora.
  - 'Produccion' [sic]: cuando la máquina se encuentra en producción.
  - Improductiva: esta categoría está oculta para el usuario y es un valor implícito; se asigna automáticamente como "improductiva" al tiempo en que la máquina esté en *set up* o cuando esté detenida.

TIPO DE HORA:

Set Up

Figura 55. Selección del tipo de hora. Fuente: Elaboración propia.

- Razón de paro: ver Figura 56., se muestra en letras negras sobre fondo amarillo y se selecciona en la pantalla de alerta de detención.

RAZON DE PARO:

Choque de mangas programado

Figura 56. Información sobre el paro. Fuente: Elaboración propia.

- Reloj: muestra la hora en formato AM-PM según la zona horaria GMT +6

kg/h	ESPERADO	REAL
	2.0	0.0

Figura 57. El valor de producción real parpadea en rojo cuando sale del margen. Fuente: Elaboración propia.

## Configuración

La pantalla de configuración puede ser accedida por cualquier persona autorizada que tenga acceso al nombre de usuario y contraseña. Este nombre de usuario y contraseña se solicitan en la ventana de autorización (ver Figura 58.) al presionar con el puntero sobre el logo de Polymer S.A; una vez se cierre la pantalla de configuración, el valor introducido en el espacio de contraseña se reinicia.

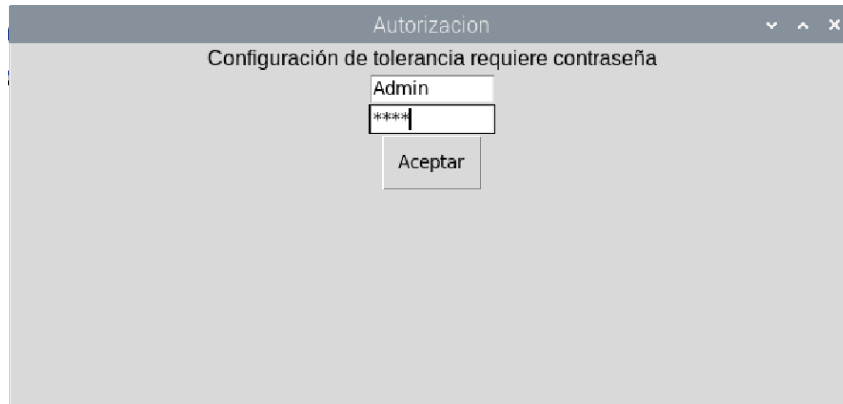


Figura 58. Pantalla de Autorización. Fuente: Elaboración propia.

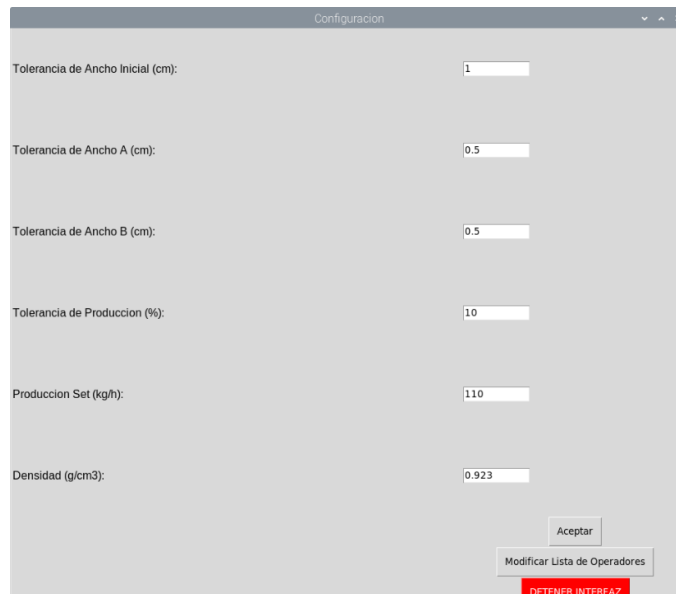


Figura 59. Pantalla de configuración con los valores predeterminados. Fuente: Elaboración propia.

Los valores de tolerancia dictan los límites inferiores y superiores tolerados para los valores de ancho, delimita asimismo el porcentaje máximo en el que puede variar la producción de forma negativa y este porcentaje también se utiliza para definir el límite inferior de la velocidad.

Cuando los valores reales y calculados se salen de los límites tolerados, las celdas de alarma se muestran en rojo, ver Figura 60.

EXTRUSORA E16


	VARIABLES	SETEADO	REAL	DIFERENCIA	ALERTA
CONTROLADAS	ANCHO INICIAL (cm)	10	9.15	-0.85	
	ANCHO FINAL A (cm)	0	9.15	9.15	
	ANCHO FINAL B (cm)	0	9.15	9.15	
	VELOCIDAD (m/min)	835.47	0 m/min	835.47	
NO CONTROLADAS	ESPESOR (mil/in)	0.52	DESPERDICIO AUTOMATICO	ESPERADO	REAL
	DENSIDAD (g/cm3)	0.923		20.0	0.0
PRODUCTO	DIGITE DATO		LAMINA	TUBO	
OPERADOR	Nombre:	**	kg/h	ESPERADO	REAL
	Numero:	**		110.0	0.0
TIPO DE HORA:		RAZON DE PARO:		Cambio de Rollo	08:41:24 AM

Figura 60. Celdas de alerta en rojo debido a que los límites de tolerancia han sido sobrepasados. Fuente: Elaboración propia.

En el caso de la Figura 60., se ha seleccionado la presentación tubular por lo que el ancho final A y ancho final B se han predeterminado en 0 cm, el valor de tolerancia en A y B es de 0,5 cm y en la etapa inicial es de 1 cm y su valor esperado se ha cambiado a 10 cm; por esta razón la celda de alerta para el ancho inicial no está encendida.

Se puede observar también que los valores numéricos cambian su formato a letras blancas con fondo rojo como representación complementaria que facilite la identificación del problema por parte del operador.

Desde la pantalla de configuración se puede modificar la lista de operadores y salir de la interfaz.

### Modificación de Lista de Operadores

Si se presiona con el cursor sobre el botón “Modificar Lista de Operadores” en la Figura 59., se abrirá la ventana de modificación como la siguiente:

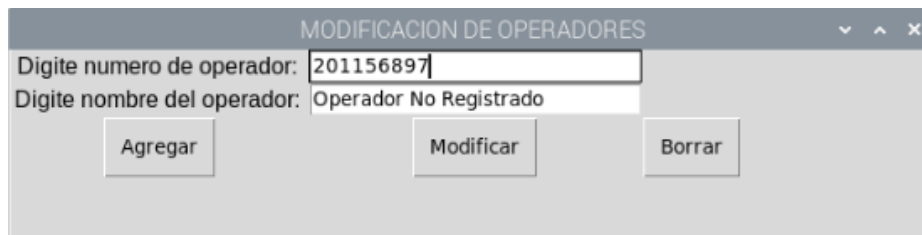




Figura 61. Ventana de modificación de la lista de operadores. Fuente: Elaboración propia.



En ella se puede agregar un nuevo operador, modificar un operador existente por número o borrar un operador existente por número. Los cambios se salvan con el toque del botón por lo que no es necesario ninguna tarea extra al terminar.

### Selección de Operador

Actualmente existe una lista precargada con los operadores asignados a la Extrusora 16, dentro de esta lista se puede seleccionar el nombre del operador al digitar el número de operador en la sección correspondiente dentro de la interfaz de usuario. Ver Figura 62.

<b>OPERADOR</b>	Nombre:	 PABLO JOSE
	Numero:	

*Figura 62. Sección de selección de operador, nombre y número ocultos por confidencialidad. Fuente: Elaboración propia.*

Si el número de operador no se encuentra dentro de la lista persistente de operadores, se observará el siguiente aviso:

<b>OPERADOR</b>	Nombre:	Operador No Registrado
	Numero:	201156897

*Figura 63. Aviso sobre número de operador desconocido. Fuente: Elaboración propia.*

Para la selección del operador no es necesario presionar ninguna tecla adicional pues se actualiza con cada tecleo.

### Ventana de Alerta por Detención

En la Figura 60., se puede observar que el valor medido de la velocidad es 0 m/min; cuando esto pasa, además del cambio en la celda de alerta y el formato del texto, aparecerá en pantalla la ventana de alerta, ver Figura 64.



*Figura 64. Ventana de alerta por detención. Fuente: Elaboración propia.*

Al presionar con el cursor sobre el elemento superior en la pantalla de alerta por detención, se puede acceder a la lista completa de razones de detención, una vez se haya seleccionado y se presione sobre el botón Registrar, la ventana de alerta por detención se ocultará automáticamente y la razón de detención se mostrará en la pantalla inicial, ver Figura 56.

La lista completa de razones de detención no puede ser modificada por el usuario y es la siguiente:

- Paro no Justificado
- Ajustes máquina en Set Up
- Camb/Ajuste Matr/ Dado Set Up
- Entonación de tintas (Set Up)
- Inspección de calidad (Set Up)
- Montaje Mecánico
- Ajuste de Imprenta en Proceso
- Ajuste de maquina [sic] por pruebas
- Aprobación del cliente
- Arranque de máquina
- Calentamiento de Máquina
- Cambio de Dado
- Cambio de Filtros
- Cambio de Rollo
- Cambio de Tesa
- Choque de mangas programado
- Comidas
- Corte electricidad

- Cuadre de Horas Improductivas
- Desmontar rollos
- Error Configuración Producto
- Error en Configuración (I&D)
- Espera Entonación de Tintas
- Falla- Falta consumibles, pist
- Falta Materia Prima/Insumos
- Falta Equipo Alisto
- Falta de Coproducto
- Falta de Jumbos
- Falta de Personal
- Falta de Repuesto
- Falta de Tintas
- Falta o Espera de Montajes
- Foto Polímero Dañado (Produc)
- Foto Polímero defectuoso (Arte)
- Limpieza de Dado
- Limpieza de equipo
- Mal Montaje de Planchas
- Mant Correctivo Eléctrico/Elec
- Mant Correctivo Mecánico
- Paro no programación
- Paro por Control de Calidad
- Problema Materia Prima/Insumo
- Problema con Tintas
- Problemas con Rollo
- Problemas con adhesivo
- Reuniones

Las razones de detención han sido añadidas textualmente a como fueron solicitadas y pueden ser modificadas por el programador únicamente. Se ahondará en el tema en la sección de programación.

Cada vez que se dé una detención, la opción que se selecciona automáticamente como la razón de detención es 'Paro no Justificado' y el tipo de hora se determina automáticamente como 'Set Up' en la pantalla de inicio (ver Figura 55.) y como 'Improductiva' en el registro automático.

La ventana de alerta por detención también aparecerá cuando el tiempo de 'Set Up' haya excedido el límite preestablecido, este límite es de 20 minutos, sin embargo, puede ser modificado por el programador. Se ahondará en el tema en la sección de programación.

Adicional a las condiciones anteriores, la ventana de alerta de detención no se podrá cerrar si no se selecciona una razón diferente a 'Paro no Justificado', en caso de que se intente seleccionar esta razón, aparecerá una alerta con letras amarillas en la pantalla con la leyenda: "Debe seleccionar una opción distinta!", ver Figura 65.



Figura 65. Restricción para selección de razón de detención. Fuente: Elaboración propia.

## Datos

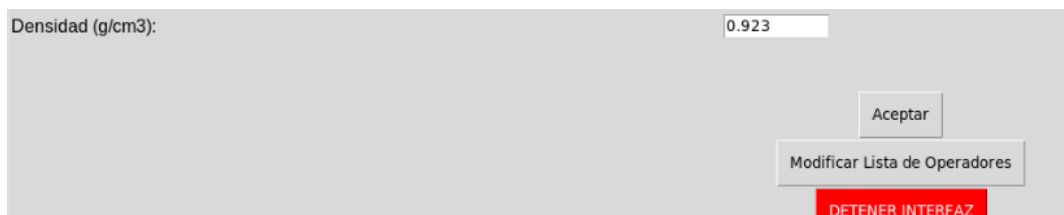
La información recopilada por los sistemas de adquisición de datos, la información introducida en la interfaz de usuario y los valores calculados por el algoritmo, serán utilizados de 3 formas distintas:

- Gráfica: mostrándolos en la interfaz de usuario.
- En línea: el algoritmo de la interfaz de usuario se encarga de enviar todos los valores a la plataforma Integromat [3] cada 5 minutos. Además, se encarga de enviar una alerta para que dicha plataforma envíe un mensaje de texto personalizado por la aplicación Telegram y por email de Outlook.
- Como registro textual: el algoritmo de la interfaz de usuario se encarga de mantener un registro en formato CSV (valores separados por coma, según siglas en inglés) de todos los valores cada 5 segundos. Además, a la media noche todos los días, envía este archivo de texto a la plataforma Integromat [5] para ser enviados por correo por dicha plataforma.

Los dos últimos elementos serán descritos a fondo en la sección de programación.

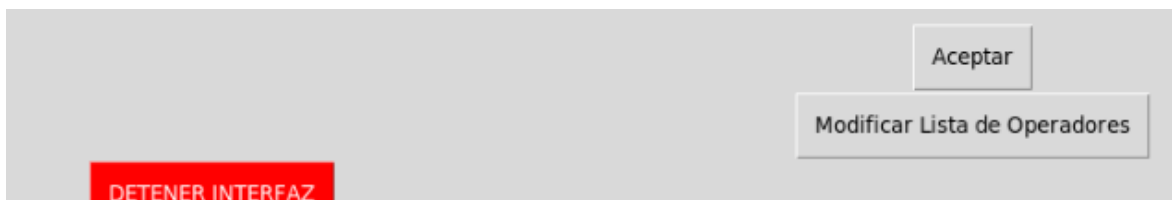
## Salida de la Interfaz

El cierre de la interfaz de usuario se puede realizar únicamente desde la ventana de configuración con el botón en rojo “DETENER INTERFAZ”. Ver Figura 66.



*Figura 66. Botón primario de salida. Fuente: Elaboración propia.*

Como medida de seguridad y para evitar salidas indeseadas, cuando el puntero entra en los límites del primer botón, este último se oculta y se muestra el botón secundario que es exactamente igual al primario con el cual se puede salir de manera definitiva. Ver Figura 67.



*Figura 67. Botón secundario de salida. Fuente: Elaboración propia.*

## Ocultar Pantalla de Inicio

La pantalla de inicio está programada de tal manera que no se pueda salir de ella sin embargo puede resultar necesario para la adquisición de los archivos de registro, para el cambio de contraseña, para resolución de problemas, etc.

Si este es el caso, al acceder a la ventana de configuración, también se mostrará una pequeña ventana que permite ocultar la pantalla inicial al presionar con el cursor sobre el botón “Ocultar Interfaz” y desbloqueará el escritorio a la vez que se oculta el primer botón. Ver Figura 68.

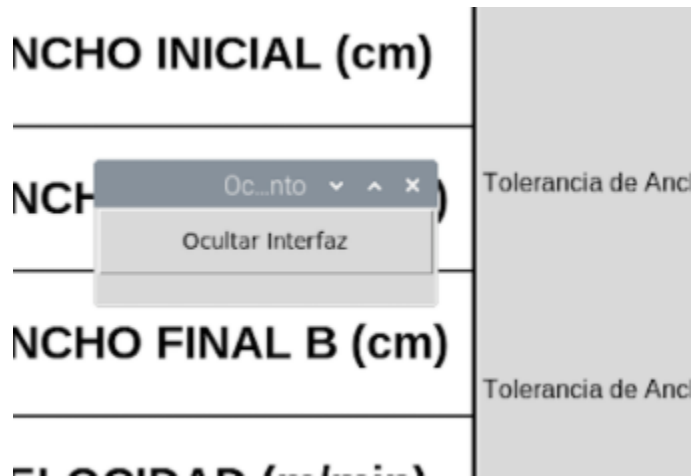


Figura 68. Botón de ocultamiento. Fuente: elaboración propia.

El segundo botón "Mostrar Interfaz" estará disponible cuando se oculte el primero y este permitirá mostrar la interfaz nuevamente.

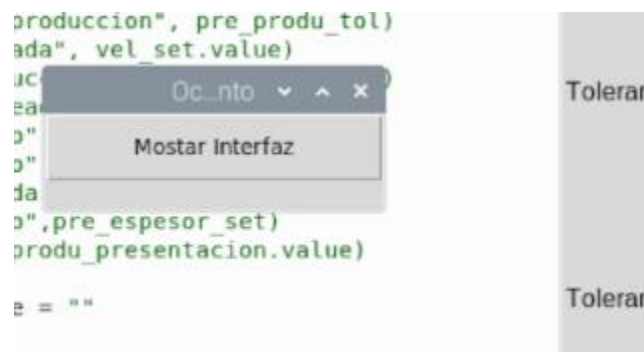


Figura 69. Botón "Mostrar Interfaz". Fuente: Elaboración propia.

## Programación de la Interfaz

Como anteriormente se mencionó, las bibliotecas que se están utilizando para programar la interfaz en Python, se llaman **guizero** [1] y **tkinter** [2]. **guizero** es basada en **tkinter** por lo que ambas son mutuamente compatibles.

## Creación de Aplicación

La aplicación es el elemento principal en la Interfaz de Usuario de la Extrusora 16 y son creadas de la siguiente manera:

- `app = App(title="Extrusora E16", width = t_width, height = t_height, layout = "grid")`

La función 'App' es propia de **guizero** y recibe los argumentos **title** (título), **width** (ancho), **height** (alto) y **layout** (organización interna de los elementos dentro de la ventana). Los valores para el ancho y el alto tienen que ser números enteros.

La clase que la función App retorna, se le asigna a la variable llamada **app** la cual en este caso corresponde a la aplicación desarrollada.

## Creación de Ventanas

En caso de ser necesaria la creación de ventanas adicionales a la principal, se utiliza una línea de código como la siguiente:

- **`pass_window = Window(app, title = "Autorizacion", width = int(p_width), height = int(p_height))`**

La función 'Window' es propia de **guizero** y se encarga de la creación de ventanas adjuntas al contenedor que se asigna en el primer argumento, en este caso sería la aplicación **app**; con los argumentos de título, ancho y alto.

Para la manipulación de dichas ventanas se utilizan algunos métodos tales como:

- **`pass_window.hide()`**: este método se encarga de ocultar la ventana.
- **`pass_window.show()`**: este método se encarga de mostrar la ventana.
- **`pass_window.tk.config(relief= "sunken")`**: el sufijo **`.tk.config()`** permite acceder a la configuración del elemento base en **tkinter** contenido en la variable **pass\_window**.
- **`pass_window.tk.attributes('-topmost', True)`**: el sufijo **`.tk.attributes()`** puede modificar atributos inherentes de las ventanas en el sistema operativo. En este caso en particular, esta línea asegura que cada vez que se acceda a la ventana, esta se muestre por encima de la principal y no pueda perderse al presionar con el cursor en cualquier otro lugar.

La Interfaz de Usuario de la Extrusora 16 tiene las siguientes ventanas:

- **`config_window`**: esta es la ventana de configuración.
- **`pass_window`**: esta es la ventana de autorización.
- **`detencion_window`**: esta es la ventana de alerta por detención.

## Creación de Cajas

Las cajas son contenedores gráficos rectangulares que permiten organizar los distintos elementos de la interfaz para mantener el orden y la configuración deseada.

Las cajas se crean de la siguiente manera:

- $tabla\_sup = \text{Box}(app, width = t\_width, height = \text{int}(h\_celda), border = True, grid = [0,1])$

La función 'Box' se encarga de la asignación de un objeto caja a una variable, en este caso **tabla\_sup**, con los argumentos de: contenedor, en este caso **app**, ancho, alto, **border** (borde) y **grid**, en este caso es necesario especificar pues **app** tiene una configuración tipo **grid**, y las coordenadas **[0,1]** significan que **tabla\_sup** ocupa el primer lugar de forma horizontal y el segundo lugar de forma vertical.

En la Figura 70., se puede observar la configuración espacial de las principales cajas que conforman la interfaz de usuario. Dentro de cada caja puede haber una o varias cajas más de acuerdo con la necesidad.

EXTRUSORA E16					
Polymer	VARIABLES	SETEADO	REAL	DIFERENCIA	ALERTA
CONTROLADAS	ANCHO INICIAL (cm)	10	9.15	-0.85	ALERTA
	ANCHO FINAL A (cm)	0	9.15	9.15	
	ANCHO FINAL B (cm)	0	9.15	9.15	
	VELOCIDAD (m/min)	835.47	0 m/min	835.47	
NO CONTROLADAS	ESPESOR (mil/in)	0.52	DESPERDICIO AUTOMATICO	ESPERADO	REAL
	DENSIDAD (g/cm3)	0.923		20.0	0.0
PRODUCTO	DIGITE DATO		LAMINA	TUBO	
OPERADOR	Nombre:	**	kg/h	ESPERADO	REAL
	Numero:	**		110.0	0.0
TIPO DE HORA:		RAZON DE PARO:		Cambio de Rollo	08:41:24 AM

Figura 70. Distribución de las cajas principales de la interfaz. Fuente: Elaboración propia.

## Creación de Texto

Cualquier elemento de texto que no vaya a ser utilizado para introducir datos o seleccionar opciones, se programa de la siguiente manera:

- $produ\_set = \text{Text}(produ\_set\_box, text = "***", align = "top", color = "#008000")$

La función **Text()** es propia de **guizero** y asigna un objeto de texto a la variable **produ\_set** con los parámetros de: contenedor, en este caso **produ\_set\_box**, **text**, en este caso **"\*\*"**, **align** (alineación) que es la justificación general del texto dentro del contenedor y por último el color.

La creación de un texto no requiere la asignación del objeto a una variable, puede crearse utilizando únicamente los parámetros de entrada.



Los métodos más frecuentes para su modificación en el código fuente de la Interfaz de Usuario de la Extrusora 16 son:

- **`produ_set.tk.config(font='Helvetica 44 bold')`**: con el sufijo **`.tk.config()`** se cambia el tipo de letra a Helvética en negrita de tamaño 44.
- **`produ_set.value = pre_produ_set`**: con el sufijo **`.value`** y el operador **`=`** se le asigna al texto el valor de la variable **`pre_produ_set`**

ESPERADO
110.0

Figura 71. `produ_set` corresponde al texto inferior "110". Fuente: Elaboración propia.

## Creación de Cuadros de Texto

Los cuadros de texto se utilizan cuando el usuario necesita introducir un dato en la interfaz. Estos cuadros se crean de la siguiente manera:

- **`nombre_operador = TextBox(nombre_operador_box, text= "**", width = "fill", height = "fill")`**

La naturaleza de la creación de los cuadros de texto es muy similar a la de los objetos anteriores donde se requieren los parámetros: contenedor, texto, ancho y alto; en el caso de la línea anterior, el comando **`fill`** provoca que el cuadro de texto llene el espacio provisto por el contenedor.

El tamaño de letra del texto dentro del cuadro de texto se puede modificar con el siguiente método:

- **`nombre_operador.text_size = ("24")`**

El sufijo **`.text_size`** junto a los operadores **`= (" ")`** permiten asignar cualquier tamaño de letra.

Nombre:	**
Numero:	**

Figura 72. `nombre_operador` corresponde a la primera casilla de la segunda columna. Fuente: propia.

## Creación de un Grupo de Botones

Para la Interfaz de Usuario de la Extrusora 16 solo se utiliza un grupo de botones que permite la selección de la presentación del producto según sea tubular o laminar.

- **`produ_presentacion = ButtonGroup(select_presentacion_box, options = ["LAMINA", "TUBO"], selected = "top", align = "left", command = actualizar_presentacion, horizontal = True)`**

Los parámetros son iguales a los anteriores sin embargo se introducen los parámetros: **`options`**, en este caso LAMINA y TUBO, **`selected`**, en este caso el comando **`"top"`** selecciona la primera opción, **`command`**, que asigna una función de acción a la selección, en este caso **`actualizar_presentacion`** y **`horizontal`** para definir si se extenderá de forma vertical u horizontal.

El color se asigna con la iteración representada en la Figura 73.

```
cont_button = 0
for radio_button in produ_presentacion.children:
    tk_radio_button = radio_button.tk
    if cont_button == 0:
        radio_button.tk.config(bg = "#06AED5")
    elif cont_button == 1:
        radio_button.tk.config(bg = "#808080")
    cont_button +=1
```

Figura 73. Iteración para la alternancia del color en las opciones de `produ_presentacion`. Fuente: Elaboración propia



Figura 74. Grupo de botones de nombre `produ_presentacion`. Fuente: Elaboración propia.

## Creación de un Botón

Los botones son utilizados para que la interfaz reacciones a distintas necesidades y comandos que el usuario puede requerir.

- **`pass_button = Button(empty_sup.tk, command = ventana_contrasena, text= "Config", image = img, cursor = "heart").pack()`**

‘Button’ es una función de **`tkinter`** que retorna un objeto de tipo **`Button`** que se asigna a **`pass_button`** dentro del contenedor **`empty_sup.tk`** que corresponde a un objeto tipo caja, sin embargo, debido a que la función es propia de **`tkinter`** y no

**guizero** es necesario especificar que se utilizará como objeto perteneciente a **tkinter** y no a **guizero**. Al igual que para la creación de un grupo de botones, a cada botón se le puede asignar una función **command**, en este caso es la función **ventana\_contrasena** la cual deja mostrar la ventana de autorización. En el caso de este botón en particular, se le asigna una imagen que existe dentro de la variable **img**.



Figura 75. Botón para acceder a la ventana de autorización. Fuente: Elaboración propia.

### Creación de una Lista de Opciones

Las listas de opciones permiten asignar un valor a una variable entre una lista predeterminada.

- **`tipo_hora = Combo(tipo_hora_box, options = ["Set Up", "Produccion"])`**

De manera similar a los casos anteriores, 'Combo' es una función de Guizero que asigna un objeto de tipo lista de opciones a la variable **tipo\_hora** dentro del contenedor **tipo\_hora\_box** con las opciones "Set Up" y "Produccion".

Con el siguiente método se puede cambiar el color:

- **`tipo_hora.tk.config(bg = "GREEN")`**

Donde los operadores (**bg = " "**) permiten asignar un color de fondo determinado.

---

TIPO DE HORA:

Set Up –

Figura 76. La variable **tipo\_hora** está a la derecha de color verde. Fuente: Elaboración propia.

## Adquisición de los Datos en Serie desde los Microprocesadores

Para habilitar la comunicación serial en Python se utiliza la biblioteca pySerial [4], los puertos USB para la comunicación serial se inicializan de la siguiente manera:

```
"INICIALIZACION DE COMUNICACION SERIAL"
#""
vel_serial =serial.Serial('/dev/ttyUSB3', 9600)
a_ini_serial= serial.Serial('/dev/ttyUSB2', 9600)
a_fin_a_serial= serial.Serial('/dev/ttyUSB1',9600)
a_fin_b_serial= serial.Serial('/dev/ttyUSB0',9600)

vel_serial.flushInput()
a_ini_serial.flushInput()
a_fin_a_serial.flushInput()
a_fin_b_serial.flushInput()
#""
```

*Figura 77. Inicialización de la comunicación serial. Fuente: Elaboración propia.*

Para asignar una dirección serial específica se utiliza el método **Serial (' ', )** donde el primer argumento es el nombre del puerto habilitado y el segundo argumento es la frecuencia de comunicación en baudios. En el caso actual **'/dev/ttyUSB3'** es el nombre que la Raspberry Pi 4 le asigna al puerto al que está conectado el sistema de adquisición de datos de la velocidad, **'/dev/ttyUSB2'** es el nombre asignado al puerto para el sistema de adquisición de datos del ancho inicial y así sucesivamente.

Una vez asignados los puertos, se limpia la entrada de cualquier dato residual con **.flushInput()**

Los datos son enviados desde los microprocesadores con un formato de línea en hexadecimal, por ejemplo:

- **4B1\n**: este número corresponde a 1201 en base decimal con un salto de línea.

Debido a la razón anterior, antes de poder ser utilizados, la información debe de ser interpretada y convertida; las distintas funciones para la adquisición de estos datos lo logran con líneas que siguen el estilo siguiente:

```
ini_Bytes = puerto_ancho.readline()
try:
    ini_line = ini_Bytes.decode('utf-8').strip()
    try:
        ini_line = int(ini_line,16)
```

*Figura 78. Adecuación de la información serial entrante. Fuente: Elaboración propia.*

En la Figura 78., la primera instrucción se encarga de leer la entrada del puerto serial en formato de línea, la tercera instrucción se encarga de separar el número del salto de línea **\n** y la última instrucción se encarga de convertir el número de hexadecimal a decimal.

## Creación de la Lista Persistente de Operadores

La creación de la lista persistente de operadores se realiza con ayuda de la biblioteca **shelve** [13] en las siguientes líneas:

```
s = shelve.open('ops')
operadores =s['ops']
```

*Figura 79. Lista persistente de operadores. Fuente: propia.*

El archivo **ops** es en el que se almacena la lista de operadores actual, con el método **shelve.open()** este se asigna a la variable **s** y posteriormente el diccionario **ops**, que contiene la lista de operadores, se asigna al diccionario **operadores**.

Una vez se haya terminado de utilizar **s** es necesario almacenar los cambios en el diccionario **operadores**, almacenarlos y cerrar el archivo para que pueda ser utilizado por otras funciones. Ver Figura 80.

```
try:
    s['ops'] = operadores
finally:
    s.close
```

*Figura 80. Almacenamiento de cambios y cierre de archivo. Fuente: Elaboración propia.*

Las funciones que utilizan este diccionario en el código fuente de la Interfaz de Usuario para la Extrusora 16 estan en la biblioteca **interfaz\_E16.py** y son:

- **actualizar\_op(event)**: esta se encarga de la actualización del nombre de operador según el número de operador en la pantalla principal. Se activa a partir de un evento en dicho elemento gráfico.
- **buscar\_op(event)**: se encarga de actualizar el nombre del operador según el número en la ventana de actualización de la lista de operadores.
- **agregar()**: se encarga de agregar un operador nuevo.
- **modificar()**: se encarga de modificar un operador existente.
- **borrar()**: se encarga de borrar un operador existente.

## Funciones Activadas por Evento

La biblioteca *tkinter* permite actualizar y controlar elementos gráficos al definir eventos que llamen funciones según la necesidad. En el código fuente de la Interfaz de Usuario para la Extrusora 16 se utilizan de la siguiente manera:

- ***kill\_button.tk.bind('<Enter>',careful)***

La línea anterior enlaza el evento '**<Enter>**' a la función ***careful()*** para el botón ***kill\_button***. El evento '**<Enter>**' corresponde a la entrada del cursor dentro de los límites gráficos del botón ***kill\_button*** de la ventana de configuración.

Las funciones activadas por evento en el código fuente de la interfaz son:

- ***caps\_produ(event)***: se encarga de cambiar el formato de entrada del texto a mayúscula en el espacio de entrada del tipo de producto en la pantalla de inicio.
- ***caps(event)***: se encarga de cambiar el formato de entrada del texto a mayúscula en el espacio de entrada del nombre de operador en la ventana de modificación de la lista de operadores.
- ***actualizar\_op(event)***: se encarga de mostrar el nombre del operador según el número de operador. Ver Figuras 9., y 10.
- ***careful(event)***: Se encarga de ocultar el botón de salida inicial y mostrar el secundario. Ver Figuras

## Funciones de Adquisición de Datos

Las funciones de adquisición de datos de la Interfaz de Usuario de la Extrusora 16 son las siguientes:

### Velocidad

- ***velocidad (produ\_presentacion, pre\_produ\_factor, vel\_real,vel\_set,vel\_serial,vel\_error, data\_to\_send, vel\_alarma\_activa, tipo\_paro, detencion\_window, tipo\_hora, config\_produ\_tol, razon\_detencion, stream)***

Esta función se encarga de recibir los datos desde el puerto serial asignado al sistema de adquisición de datos de velocidad, de calcular la velocidad real según la información proveniente del microprocesador, de decidir si la velocidad medida está por debajo o por encima del margen de tolerancia, de decidir y alertar si el rodillo del sistema de adquisición de datos de velocidad se ha detenido, de activar la ventana de alerta de detención, de adjuntar al registro textual todos los eventos relacionados a la velocidad y de adjuntar los datos relacionados a la velocidad al diccionario ***data\_to\_send***. La función ***velocidad()*** también confirma si el

microprocesador detecta que la rueda se detiene y envía el dato con la hora exacta a Integromat [3] para activar una alerta por Telegram y Outlook.

Las variables que necesita para su funcionamiento son:

- ***produ\_presentacion***: esta variable indica si la presentación es tubular o laminar.
- ***pre\_produ\_factor***: es el factor relacionado a la presentación tubular o laminar.
- ***vel\_real***: es el objeto de texto que muestra el valor medido de velocidad.
- ***vel\_set***: es el objeto de texto con el valor teórico de la velocidad.
- ***vel\_error***: es el objeto de texto con el valor de la diferencia entre el valor medido y el valor teórico de la velocidad.
- ***data\_to\_send***: es un diccionario compartido entre todas las funciones para mantener el registro de la información entre todos los hilos de programación.
- ***vel\_alarma\_activa***: es el objeto de tipo caja que representa la alarma relacionada a la velocidad.
- ***tipo\_paro***: es el objeto con la información de la razón de detención.
- ***detencion\_window***: es la ventana de alerta de detención.
- ***tipo\_hora***: es el objeto con la información del tipo de hora.
- ***config\_produ\_tol***: es el objeto con la información sobre el margen de tolerancia de producción.
- ***razon\_detencion***: objeto con la lista de selección de las razones de detención.
- ***stream***: nombre del archivo actual de registro textual.

### **Ancho Inicial, Ancho Final en A y Ancho Final en B**

Entre estas funciones no cambia el algoritmo, únicamente el nombre de las variables y el valor numérico contenido en ellas.

- ***ancho\_ini(a\_ini\_real, a\_ini\_serial, a\_ini\_set, a\_ini\_error, a\_ini\_alarma\_activa, config\_a\_ini\_tol, data\_to\_send, stream)***
- ***ancho\_fin\_a(a\_fin\_a\_real, a\_fin\_a\_serial, a\_fin\_a\_set, a\_fin\_a\_error, a\_fin\_a\_alarma\_activa, config\_a\_fin\_a\_tol, data\_to\_send, stream)***
- ***ancho\_fin\_b(a\_fin\_b\_real, a\_fin\_b\_serial, a\_fin\_b\_set, a\_fin\_b\_error, a\_fin\_b\_alarma\_activa, config\_a\_fin\_b\_tol, data\_to\_send, stream)***:

Estas funciones se encargan de recibir los datos desde el puerto serial asignado al sistema de adquisición de datos del ancho inicial, final en A y final en B, de calcular la diferencia entre el ancho esperado y el ancho medido por el sistema de adquisición de datos, de activar la alerta en caso de que el ancho se haya salido de los márgenes tolerables, de adjuntar los datos al diccionario compartido entre hilos y de adjuntar los eventos relacionados a los anchos en el registro de texto.

Las variables presentan una convención de nombre y están homologadas entre ellas de la siguiente manera: en cualquiera de las variables utilizadas para el ancho inicial se puede suplantar el componente *\_ini\_* por *\_fin\_a\_* o *\_fin\_b\_* para obtener la variable homologada en la otra función; el algoritmo utilizado es exactamente igual entre las tres funciones.

Tomando en cuenta la información del párrafo anterior, las variables son:

- ***a\_ini\_real***: objeto de texto con el valor del ancho medido.
- ***a\_ini\_serial***: objeto serial con el puerto asignado a cada sistema de adquisición de datos de ancho.
- ***a\_ini\_set***: objeto de texto con el valor del ancho esperado.
- ***a\_ini\_error***: objeto de texto con el valor de la diferencia entre el ancho medido y el esperado.
- ***a\_ini\_alarma\_activa***: objeto de tipo caja para la representación de la alarma.
- ***config\_a\_ini\_tol***: objeto con el valor de la tolerancia permitida.
- ***data\_to\_send***: ver sección anterior.
- ***stream***: ver sección anterior.

## Funciones de Cálculo de Valores

Adicional a las funciones de adquisición de datos, otras funciones se encargan de utilizar la información adquirida para calcular valores de interés.

### Cálculo del Desperdicio

La función para el cálculo del desperdicio se define de la siguiente manera:

- ***calculo\_desperdicio( a\_ini\_set, a\_fin\_a\_set, a\_fin\_b\_set, a\_ini\_real, a\_fin\_a\_real, a\_fin\_b\_real, desperdicio\_esperado, desperdicio\_real, data\_to\_send, config\_a\_ini\_tol, config\_a\_fin\_a\_tol, config\_a\_fin\_b\_tol, presentacion\_produ, desperdicio\_real\_inner\_box)***

Esta función se encarga de calcular el desperdicio esperado en centímetros según los valores esperados de los anchos, de calcular el desperdicio real según los datos medidos del ancho, de representar gráficamente una alerta en caso de que el



desperdicio real se salga del margen esperado y de adjuntar los datos al diccionario compartido entre hilos.

La fórmula utilizada para calcular el desperdicio esperado es:

$$\text{Desperdicio esperado} = 2 \times AIT - (AAT + ABT) \quad (3.2)$$

Donde *AIT* es el ancho inicial teórico, *AAT* es el ancho en A teórico y *ABT* es el ancho en B teórico.

Adicionalmente, el cálculo del margen para el desperdicio real es la suma del desperdicio esperado más las tolerancias para los anchos.

$$\text{Margen} = D.E. + Tol\ Ini + Tol\ A + Tol\ B \quad (3.3)$$

Donde *D.E.* es el desperdicio esperado, *Tol Ini* es la tolerancia para el ancho inicial en centímetros, *Tol A* es la tolerancia en centímetros para el ancho en A y *Tol B* es la tolerancia en centímetros para el ancho en B.

- ***a\_ini\_set***: objeto de texto con el valor esperado para el ancho inicial.
- ***a\_fin\_a\_set***: objeto de texto con el valor esperado para el ancho en A.
- ***a\_fin\_b\_set***: objeto de texto con el valor esperado para el ancho en B.
- ***a\_ini\_real***: objeto de texto con el valor medido del ancho inicial.
- ***a\_fin\_a\_real***: objeto de texto con el valor medido del ancho en A.
- ***a\_fin\_b\_real***: objeto de texto con el valor medido del ancho en B.
- ***desperdicio\_esperado***: objeto de texto con el valor del desperdicio esperado.
- ***desperdicio\_real***: objeto de texto con el valor del desperdicio esperado.
- ***data\_to\_send***: ver secciones anteriores.
- ***config\_a\_ini\_tol***: objeto de texto con el valor de la tolerancia para el ancho inicial.
- ***config\_a\_fin\_a\_tol***: objeto de texto con el valor de la tolerancia para el ancho en A.
- ***config\_a\_fin\_b\_tol***: objeto de texto con el valor de la tolerancia para el ancho en B.
- ***presentacion\_produ***: objeto con el formato de presentación del producto.
- ***desperdicio\_real\_inner\_box***: objeto tipo caja en el que está contenido el valor del desperdicio real. Este objeto parpadea si el desperdicio se sale del margen.

## Cálculo de Producción

La función para el cálculo de la producción se define de la siguiente manera:

- ***calculo\_produccion***(*produ\_set*, *produ\_calc*, *a\_fin\_a\_set*, *a\_fin\_b\_set*, *vel\_set*, *a\_fin\_a\_real*, *a\_fin\_b\_real*, *presentacion\_produ*, *vel\_real*, *densidad\_set*, *espesor\_set*, *produ\_calc\_box*, *config\_produ\_tol*, *data\_to\_send*)

Esta función se encarga del cálculo de la producción real, del cálculo de la velocidad esperada, de la alerta visual en caso de que la producción real se salga del margen y de adjuntar los datos al diccionario compartido entre hilos.

La fórmula que utiliza para calcular la producción real es:

$$\text{Producción} \left[ \frac{\text{kg}}{\text{h}} \right] = \frac{\text{Densidad} \times \text{Espesor} \times \text{Velocidad} \times 1,524 \times [\text{Ancho A Real} + \text{Ancho B Real}]}{100} \quad (3.4)$$

Y la fórmula que utiliza para calcular el valor esperado de la velocidad en metros por minuto es:

$$\text{Velocidad esperada} = 65,617 \times \frac{P \text{ set}}{D \times E \times F.P. \times (AAE + AAB)} \quad (3.5)$$

Donde *D* es densidad, *E* es espesor, *F.P.* es el factor de producción el cual es 1 si la presentación es laminar y 2 si la presentación es tubular, *AAE* es el ancho en A esperado y *AAB* es el ancho en B esperado.

Las variables que necesita esta función son:

- ***produ\_set***: objeto con el valor de producción esperado.
- ***produ\_calc***: objeto con el valor de producción real.
- ***a\_fin\_a\_set***: ver apartados anteriores.
- ***a\_fin\_b\_set***: ver apartados anteriores.
- ***vel\_set***: ver apartados anteriores.
- ***vel\_real***: ver apartados anteriores.
- ***a\_fin\_a\_real***: ver apartados anteriores.
- ***a\_fin\_b\_real***: ver apartados anteriores.
- ***densidad\_set***: objeto con el valor de la densidad.
- ***espesor\_set***: objeto con el valor del espesor.
- ***produ\_calc\_box***: objeto tipo caja. Parpadea si el valor de la producción real se sale del margen inferior.
- ***config\_produ\_tol***: objeto con el valor de la tolerancia de producción.
- ***data\_to\_send***: ver apartados anteriores.

## Comunicación y Almacenamiento de Datos

El almacenamiento de los datos a través de la interfaz se realiza en varias etapas o niveles, algunas etapas están construidas como funciones, otras están repartidas a lo largo de las funciones ya explicadas.

### Registro Textual de Datos

La Interfaz de Usuario de la Extrusora 16 mantiene el registro textual de datos de dos maneras distintas:

- Archivo de texto con oraciones generales sobre los principales eventos que suceden en las funciones de adquisición de datos: este archivo es modificado por todas aquellas funciones que en sus argumentos de entrada se encuentre **stream**.

A este archivo de texto se le asigna su nombre con la siguiente línea:

```
o stream = name("LOG %s" %  
datetime.datetime.now().strftime("%Y-%m-%d_%H-%M-%S"))
```

Donde **name()** está definida por la función en la Figura 81.,

```
def name(s):  
    return s + ".txt"
```

*Figura 81. Definición de la función name(). Fuente: Elaboración propia.*

La combinación de la instrucción inicial y la llamada de la función **name()** generan un nombre del formato "LOG 2021-04-29\_09-14-49.txt"; este nombre se genera en el momento en que se inicializa la interfaz utilizando la biblioteca **datetime** [8] y el archivo es creado aleatoriamente por el primer hilo que se ejecute.

```
file = open(stream, "a+")  
  
sleep(minutos*60)  
detencion_window.show()  
  
file.write("\n")  
file.write("El setup por el operador %s " % str(nombre_operador.value))  
file.write("numero %s " % str(numero_operador.value))  
file.write("debia finalizar a las %s. \n" % str(datetime.datetime.now().strftime("%Y-%m-%d %H:%M")))  
file.write("Se contaba con %s minutos para finalizar \n" % str(minutos))  
file.write("\n")
```

*Figura 82. Ejemplo de código para adjuntar datos en el archivo stream. Fuente: Elaboración propia.*

El algoritmo de la Figura 82., permite escribir sobre el mismo archivo sin importar cuál de las llamadas lo haya creado sin crear duplicados.

El numero de pasos para el ANCHO INICIAL es 539  
 La medida del ANCHO INICIAL es de 11.049499999999998 cm  
 El promedio de las ultimas 1 medidas para el ANCHO INICIAL es 11.05 cm  
 El numero de pasos para el ANCHO FINAL A es 539  
 La medida del ANCHO FINAL A es de 11.049499999999998 cm  
 El promedio de las ultimas 1 medidas para el ANCHO FINAL A es 11.05 cm  
 El numero de pasos para el ANCHO FINAL B es 539  
 La medida del ANCHO FINAL B es de 11.049499999999998 cm  
 El promedio de las ultimas 1 medidas para el ANCHO FINAL B es 11.05 cm  
 LA RUEDA SE DETUVO A LAS 2021-04-29 09:14:51.

Figura 83. Registro textual de datos. Fuente: Elaboración propia.

- Archivo de texto con valores separados por coma: Este archivo es generado dentro de la función encargada del envío de datos. Ver Figura 84.

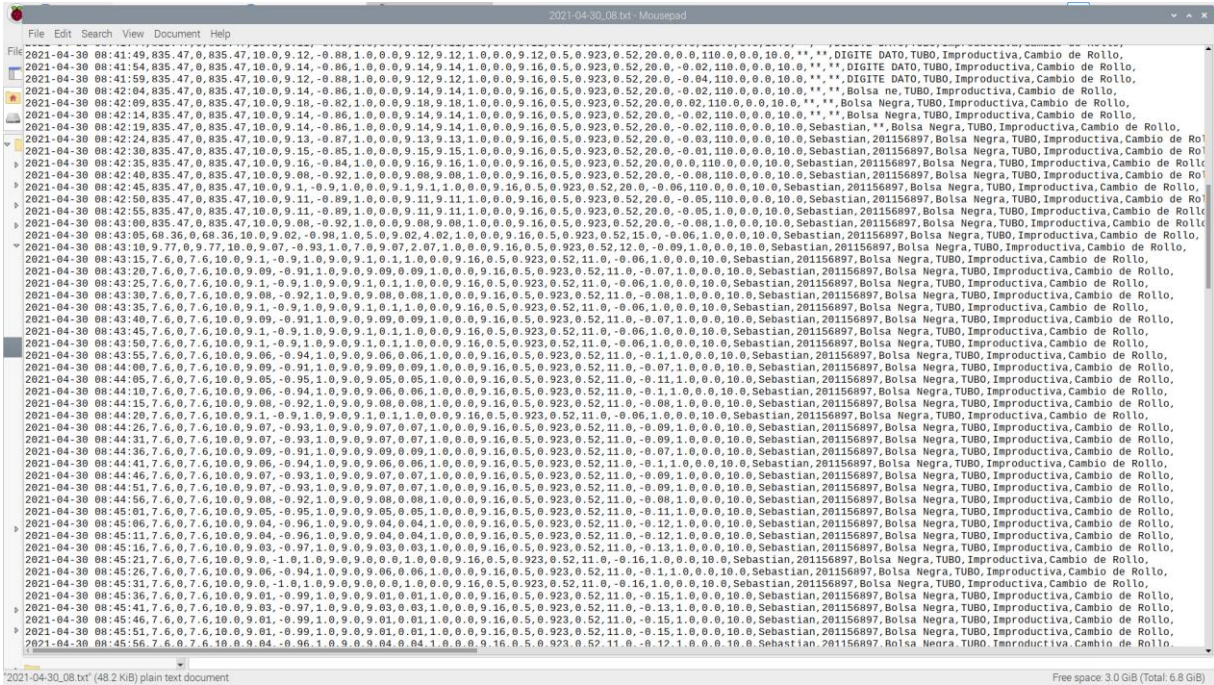


Figura 84. Archivo de valores separados por comas. Fuente: Elaboración propia.

## Función para el Envío de Datos

Esta función se encarga de la creación del archivo de texto de valores separados por comas, del envío de los datos a Integromat [3] cada 5 minutos, del almacenamiento de los valores en el archivo de texto de valores separados por comas cada 30 segundos y del envío de este registro a Integromat para que sea enviado por correo después de un tiempo determinado.

La función está definida así:

- `envio_datos(produ_tipo, nombre_operador, numero_operador, data_to_send, variables, stream, tipo_paro)`

Esta función también adjunta datos al diccionario compartido entre hilos que no fueron utilizados por otras funciones para que no sean excluidos del registro.

Las variables que requiere son:

- **produ\_tipo**: objeto de texto con el nombre del producto.
- **nombre\_operador**: objeto de texto con el nombre del operador.
- **numero\_operador**: objeto de texto con el número del operador.
- **data\_to\_send**: diccionario compartido entre hilos.
- **variables**: lista de variables de interés.
- **stream**: ver secciones anteriores.
- **tipo\_paro**: ver secciones anteriores.

La función **envio\_datos()** genera continuamente el nombre del archivo de registro de valores separados por comas con el día actual lo que permite compararlo consigo mismo y enviarlo por Internet cada cambio de hora o día. Ver Anexo.

### Temporizador del ‘Set Up’

Esta función se encarga de definir el tiempo total que se le permite al operador mantener la máquina en ‘Set Up’ y si este tiempo se excede, se muestra la ventana de alerta de detención.

1. **temp\_setup (tipo\_hora, detencion\_window, nombre\_operador, numero\_operador, stream)**

Para cambiar el tiempo disponible se debe cambiar la asignación después del operador = para la variable minutos. Ver Figura 85.,

```
def temp_setup (tipo_hora, detencion_window, nombre_operador, numero_operador, stream):  
    cont_setup = 0  
    minutos = 1
```

*Figura 85. Definición y configuración de la función temp\_setup(). Fuente: Elaboración propia.*

### Sobre los Hilos para las Funciones

Debido a la forma en la que se ejecuta la interfaz gráfica en **guizero** y **tkinter**, la ejecución de las funciones no se puede realizar en bucle por lo que la creación de ciclos infinitos y llamadas directas a funciones no tendrá el resultado esperado, sino que detendrá o ralentizará el funcionamiento de la interfaz gráfica.

Para solucionar este problema se utiliza la programación de hilos utilizando la biblioteca **threading** [5]. Los hilos presentes en la Interfaz de Usuario para la Extrusora 16 tienen la siguiente forma:

- ***threading.Thread(target = temp\_setup, args = (tipo\_hora, detencion\_window, nombre\_operador, numero\_operador, stream)).start()***

La creación del hilo se hace a partir del método ***threading.Thread()*** con los argumentos ***target*** que corresponde al nombre de la función que se ejecutará dentro del hilo y ***args*** que corresponde a los argumentos de entrada de la función. Una vez creado el hilo, se inicializa con ***.start()***

Las funciones de la Interfaz de Usuario para la Extrusora 16 son:

- ***velocidad***
- ***ancho\_ini***
- ***ancho\_fin\_a***
- ***ancho\_fin\_b***
- ***envio\_datos***
- ***temp\_setup***
- ***calculo\_produccion***
- ***calculo\_desperdicio***

Es necesario tener en consideración que la creación de hilos tiene un costo computacional que puede llegar a ser alto si dicho hilo está encargado de muchas acciones gráficas y esto mismo puede provocar que la interfaz se ejecute de manera lenta.

## **Sobre Integromat**

Integromat [3] es una plataforma de integración de tareas computacionales para la automatización de procesos. Funciona a base de escenarios que se ejecutan cada cierto tiempo o cuando ocurra un evento disparador, en este caso Webhooks.

La plataforma Integromat se está utilizando en el presente proyecto para el manejo y transporte de información de interés, y para la creación de alertas según sea la necesidad.

Dentro del código fuente de la interfaz se utiliza la biblioteca ***requests*** [6] que permite el envío de datos y archivos a través de internet. El envío de datos se realiza con instrucciones semejantes a la siguiente:

1. ***r= requests.post("https://hook.integromat.com/a6tmoarb9bimlgl91nj3nyfrqme25t5e",data\_to\_send)***

La función ***requests.post()*** envía el diccionario compartido entre hilos ***data\_to\_send*** a la dirección ***https://hook.integromat.com/a6tmoarb9bimlgl91nj3nyfrqme25t5e*** esta dirección web es generada por uno de los módulos dentro de uno de los escenarios

de Integromat; la respuesta es asignada a un objeto **r** y con la siguiente línea se puede verificar si se envió correctamente: \

## 2. *if (r.status\_code == 200):*

Los escenarios de Integromat creados para el funcionamiento de la Interfaz de Usuario para la Extrusora 16 son los siguientes:

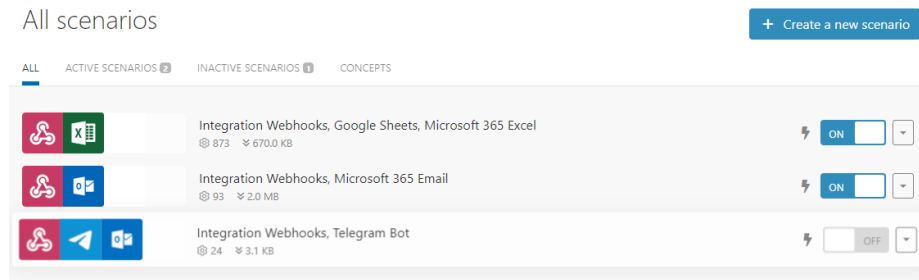


Figura 86. Escenarios de Integromat creados para la interfaz. Fuente: adaptado de [29].

El primer escenario se encarga de la recepción del diccionario **data\_to\_send** y de la inclusión de estos datos dentro de un documento de Excel 365. Ver Figura 87.

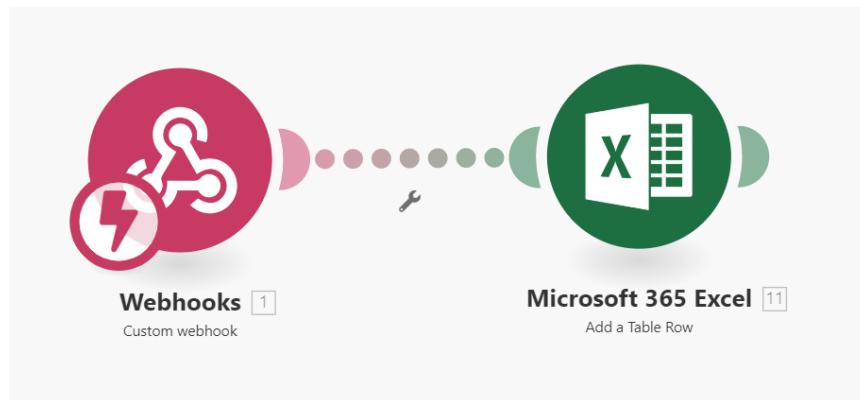


Figura 87. Representación gráfica del escenario 1. Fuente: adaptado de [29].

La configuración de los módulos se realiza presionando con el cursor sobre cada módulo y se mostrará cómo se observa en las Figura 88., y Figura 89., respectivamente.

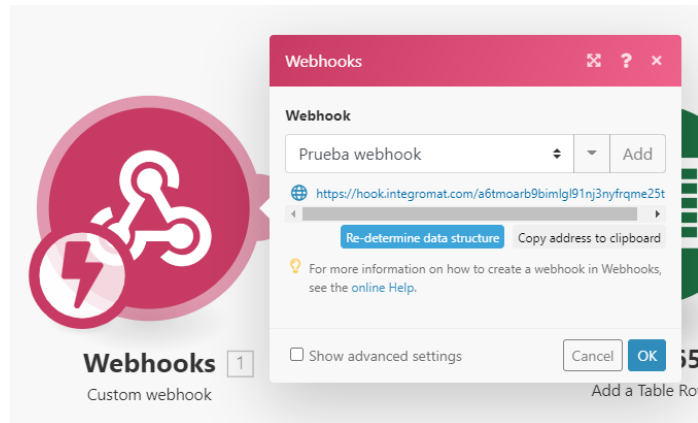


Figura 88. Modulo disparador Webhooks para el escenario 1. Fuente: adaptado de [29].

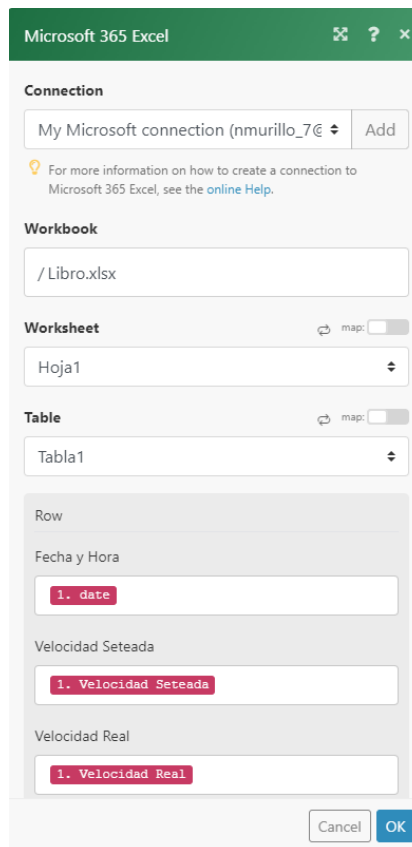


Figura 89. Modulo Microsoft 365 Excel encargado de agregar la información al documento. Fuente: adaptado de [29].

La ejecución de este escenario se llama dentro de la función **envio\_datos** con la instrucción que se utilizó de ejemplo anteriormente.

El segundo escenario se encarga de enviar el archivo de valores separados por comas por correo electrónico al final de cada día. Ver Figura 90.



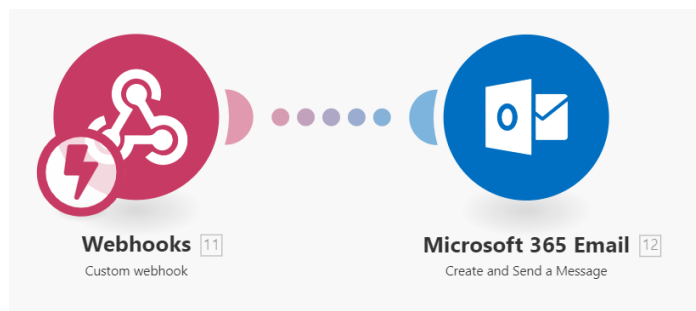


Figura 90. Escenario 2. Fuente: adaptado de [29].

De igual manera los módulos se configuran o modifican presionando con el cursor sobre ellos; el módulo de Microsoft 365 Email se observa de la siguiente manera:

The screenshot shows the configuration window for the 'Microsoft 365 Email' module. The window has a blue title bar with the text 'Microsoft 365 Email' and standard window controls (maximize, help, close). The main content area is divided into several sections:

- Connection:** A dropdown menu shows 'My Microsoft connection (Sebastián N)' with an 'Add' button. Below it is a help icon and text: 'For more information on how to create a connection to Microsoft 365 Email, see the [online Help](#).'
- Subject:** A text input field contains 'Datos' with a 'now' button next to it.
- Body Content:** A text area contains the text: 'Abrir como archivo de texto. Los datos estan en formato csv.'
- Importance:** A dropdown menu is set to 'Normal'. A 'map:' toggle switch is visible to the right.
- To Recipients:** A list shows one recipient: 'Email Address: snmurillo.int@gmail.com' with edit and delete icons. An 'Add item' button is below. A 'map:' toggle switch is to the right.
- Attachments:** A list shows one attachment: 'Data: 11. value' (highlighted in red) with 'File Name: Archivo' and edit/delete icons. An 'Add item' button is below. A 'map:' toggle switch is to the right.

At the bottom, there is a note: 'The maximum size of each file is 3MB.' and a 'Show advanced settings' checkbox. On the far right are 'Cancel' and 'OK' buttons.

Figura 91. Módulo Microsoft 365 Email para el escenario 2. Fuente: adaptado de [29].

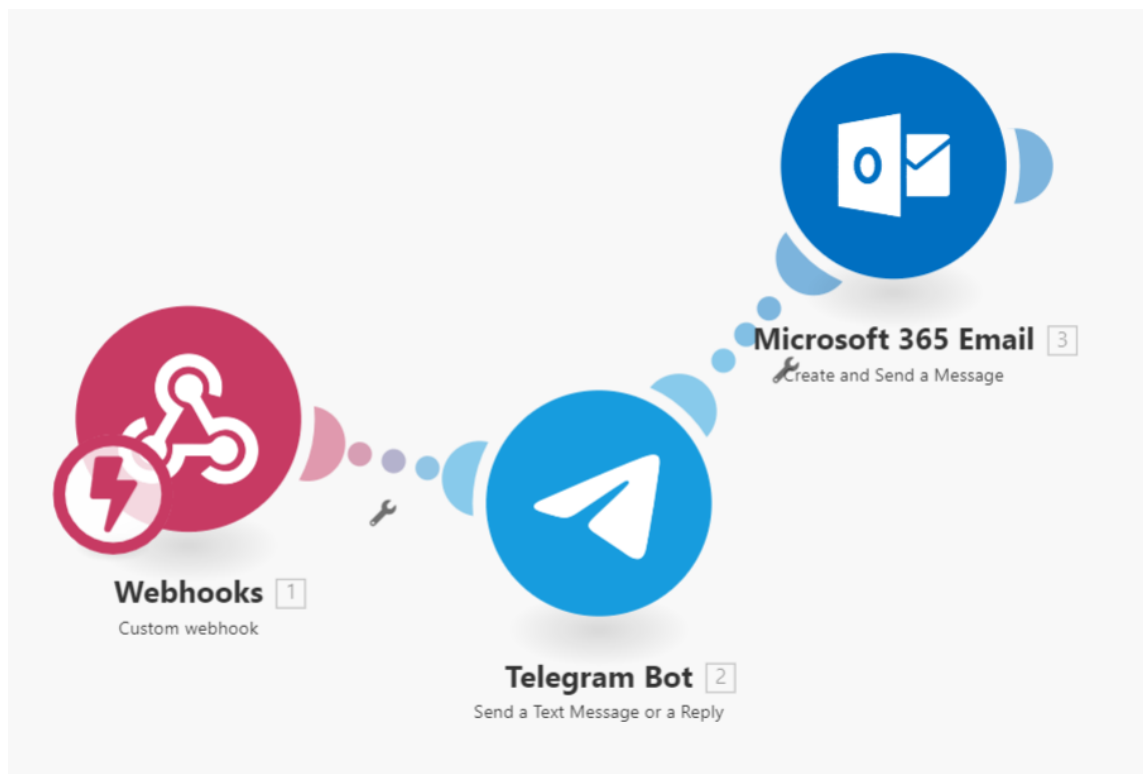
El escenario 2 se llama dentro del código fuente en la función **envio\_datos** con la instrucción:

**3. `r=requests.post("https://hook.integromat.com/t9i17x83fnu5t8oegfightrw4e1vk5p",attachment)`**

Donde la variable **attachment** es el archivo de texto que contiene los valores separados por comas, y se crea en la línea:

**4. `attachment = open(nombre_anterior, "rb")`**

El escenario 3 es el encargado de enviar la alerta escrita a través de un Bot de Telegram y un correo electrónico Microsoft 365 [12]



*Figura 92. Escenario 3. Fuente: adaptado de [29].*

Para el funcionamiento del segundo módulo es necesario la creación de un Bot de Telegram [7] que debe ser incluido dentro de un chat para que envíe los mensajes necesarios.

La configuración de este módulo se observa de la siguiente manera:

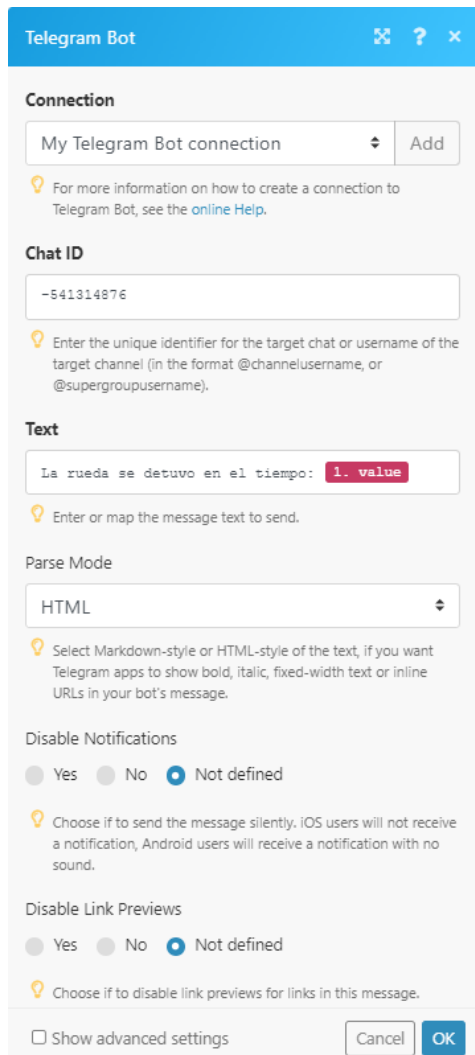


Figura 93. Configuración del módulo de Telegram. Fuente: adaptado de [29].

El mensaje de alerta enviado por el Bot se observa dentro del chat de Telegram de la siguiente manera:

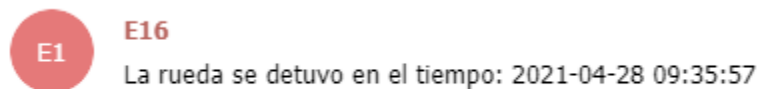


Figura 94. Mensaje de alerta enviado por el Bot de Telegram. Fuente: propia.

El escenario número 3 se activa desde la función **velocidad()** con la instrucción:

```
5. r=
requests.post("https://hook.integromat.com/xwymmmjgdovqwlvygq14
kyeuajdi4383",datetime.datetime.now().strftime("%Y-%m-%d
%H:%M:%S"))
```

El segundo argumento corresponde al momento exacto en el que se detuvo la rueda del sistema de adquisición de la velocidad.

## Sobre Alarma Visual

La torreta visual auditiva se controla directamente desde la interfaz de usuario con la ayuda de una compuerta CD4081B [10] y un *relay* a 5V. Esta torreta se enciende cuando cualquiera de los objetos visuales de alarma se encuentra activado en la interfaz (ver Figura 7).

La programación de dicha alarma se puede ver en la biblioteca ***funciones.py*** y se realiza con la ayuda de la biblioteca ***gpiozero*** [11] con las siguientes líneas:

```
led_1 = LED(2)
led_2 = LED(3)
led_3 = LED(14)
led_4 = LED(15)

led_1.off()
led_2.off()
led_3.off()
led_4.off()
```

*Figura 95. Selección de pines GPIO para la torreta. Fuente: Elaboración propia.*

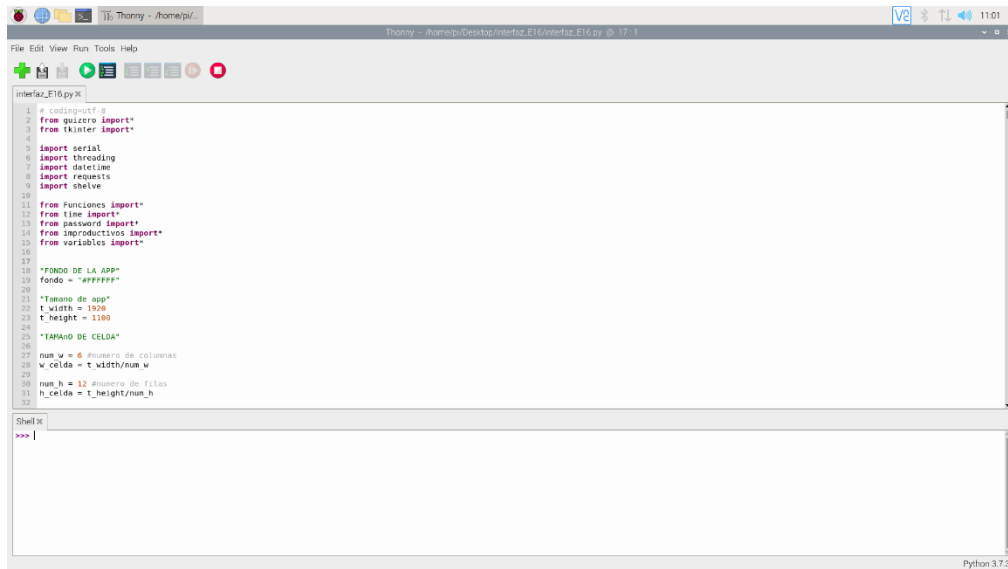
El número que se encuentra dentro de la función ***LED()*** corresponde al pin GPIO del número correspondiente. Y las variables ***led\_1***, ***led\_2***, ***led\_3*** y ***led\_4*** se activan junto a las variables:

- ***led\_1***: ***vel\_alarma\_activa*** para la velocidad.
- ***led\_2***: ***a\_ini\_alarma\_activa*** para el ancho inicial.
- ***led\_3***: ***a\_fin\_a\_alarma\_activa*** para el ancho final en A.
- ***led\_4***: ***a\_fin\_b\_alarma\_activa*** para el ancho final en B.

La alarma visual se activa con la instrucción ***led\_1.on()*** y se desactiva con ***led\_1.off()*** según corresponda

## Primera Ejecución

Esta Raspberry Pi 4 está programada específicamente para ejecutar el código fuente de la interfaz desde Thonny, al iniciar por primera vez o reiniciar la Raspberry Pi 4, ella iniciará de manera normal, inmediatamente se haya terminado de cargar el escritorio por completo, se abrirá la ventana con el código. Ver Figura 96.



```
1 # codinputT s
2 from tkinter import*
3 from tkinter import*
4
5
6 import serial
7 import threading
8 import datetime
9 import requests
10 import shelve
11
12 from Funciones import*
13 from time import*
14 from password import*
15 from impróductivos import*
16 from variables import*
17
18 *FONDO DE LA APP*
19 fondo = "FFFFFF"
20
21 *Tamaño de app*
22 t_width = 1600
23 t_height = 1180
24
25 *TAMAÑO DE CELDA*
26
27 num_w = 6 #numero de columnas
28 w_celda = t_width/num_w
29
30 num_h = 12 #numero de filas
31 h_celda = t_height/num_h
32
```

Figura 96. Código fuente en primera ejecución. Fuente: Elaboración propia.

Es de vital importancia que en este momento **NO SE MANIPULE EL TECLADO O EL CURSOR DE NINGUNA MANERA** pues los pasos que sigue automáticamente la Raspberry Pi 4 para el inicio son muy específicos.

Aproximadamente 10 segundos después, la ejecución del programa se realizará de manera automática.

Si se presenta algún inconveniente durante el inicio, se puede presionar manualmente el botón verde en forma de flecha (ver Figura 97.) o se puede presionar la tecla F5 en la ventanilla de la Figura 96. Si no es posible hacerlo de esta manera, se pueden seguir los pasos siguientes.

El script principal se encuentra en la carpeta llamada **interfaz\_E16** en el escritorio de la Raspberry. Los pasos por seguir para iniciar la Interfaz de Usuario de la Extrusora 16 son:

1. En el escritorio de la Raspberry Pi 4, acceder a la carpeta **interfaz\_E16**.
2. Dentro de la carpeta, se encuentran las bibliotecas de funciones y el archivo **interfaz\_E16.py**, se debe presionar 2 veces con el cursor sobre ese archivo.
3. Esta acción abrirá el programa Thonny, una vez abierto con el código fuente, se debe presionar con el cursor sobre la flecha verde. Ver Figura 97.

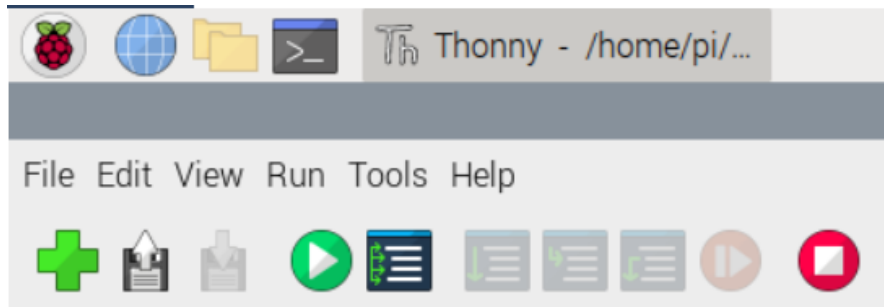


Figura 97. Panel de instrucciones del compilador Thonny. Fuente: adaptado de [79].

4. Una vez se haya presionado con el cursor sobre la flecha verde, el código fuente de la interfaz empezará a ejecutarse, la ventana puede parpadear un par de veces mientras la aplicación genera todos los elementos gráficos.

## Resolución de Problemas

A pesar de que el funcionamiento de la interfaz de usuario se probó de manera extensiva en conjunto con los sistemas de adquisición de datos, el error humano nunca deja de ser una posibilidad y el programador tampoco puede tener absoluto control sobre las aplicaciones o el hardware diseñado por terceros. Las principales fuentes de problemas posibles para la Interfaz de Usuario para la Extrusora 16 pueden ser:

### Problemas de Alimentación

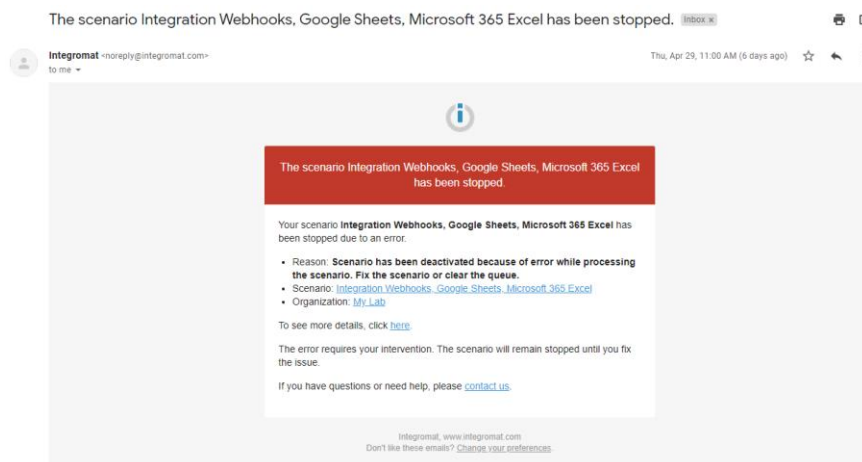
Se sugiere contar con una fuente de respaldo para evitar la interrupción inesperada de la ejecución de la interfaz gráfica. Sin embargo, siempre existe la posibilidad de una interrupción fortuita.

Si una interrupción de la alimentación llega a suceder, se deben llevar a cabo los siguientes pasos:

1. Desconectar el cable de alimentación de la Raspberry Pi 4, de preferencia en el lugar del tomacorriente en vez del conector propio de la computadora.
2. Esperar a que se normalice el suplemento de corriente.
3. Reconectar la alimentación de la Raspberry.
4. Esperar a que se inicie la interfaz de manera usual.

## Problemas con los Escenarios de Integromat

La activación y desactivación de los escenarios puede ser controlada manualmente (ver Figura 86.), sin embargo, en ocasiones los escenarios pueden recibir más información de la que están programados para procesar o el formato puede resultar confuso esporádicamente por lo que el escenario se detiene de manera automática para evitar problemas. Si esto llega a suceder, una notificación se envía al correo electrónico asociado con la cuenta de Integromat; esta notificación es similar a la siguiente:



*Figura 98. Correo de aviso de detención del escenario de Integromat. Fuente: Elaboración propia.*

Si esto sucede, la cola del escenario debe ser limpiada y el escenario debe de ser reactivado. Para realizarlo se deben seguir los siguientes pasos:

1. Presionar con el ratón sobre la pestaña **Scenarios** del panel de herramientas en Integromat. Ver Figura 99.

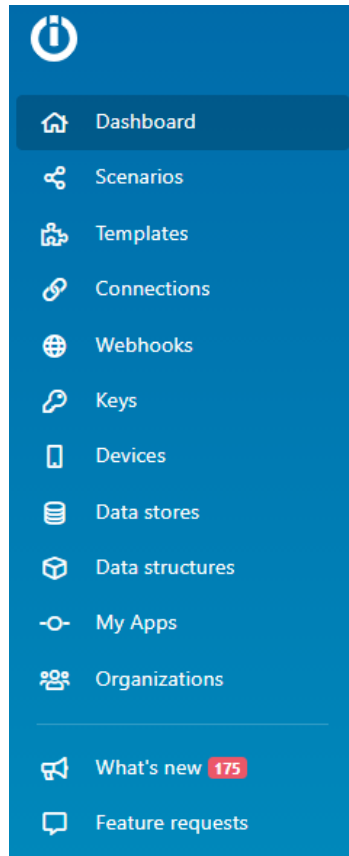


Figura 99. Panel de herramientas de Integromat. Fuente: adaptado de [29].

2. Presionar con el ratón sobre el escenario de interés. Ver Figura 86.
3. Presionar con el ratón sobre la pestaña **INCOMPLETE EXECUTIONS**. Ver Figura 100.

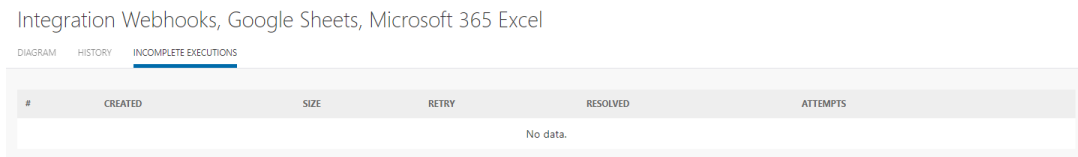


Figura 100. Pestaña INCOMPLETE EXECUTIONS. Fuente: adaptado de [29].

4. Limpiar la cola de ejecuciones incompletas.
5. Activar el escenario. Ver Figura 86.



## Problemas con los Puertos en Serie

Si accidentalmente se desconectan los adaptadores USB-TTL, es posible que la Raspberry Pi 4 confunda el orden en que han sido asignados. Si esto sucede, los pasos a seguir son los siguientes:

1. Salir de la interfaz gráfica.
2. Detener la ejecución desde Thonny. Para la realización de este paso es necesario presionar con el ratón sobre el botón cuadrado rojo o de **stop** en el panel de herramientas de la aplicación. Ver Figura 97.
3. Ejecutar el código **listausb.py** [9] que se ubica dentro de la carpeta **interfaz\_E16**.
4. Desconectar uno a uno los adaptadores USB-TTL conectados en los puertos USB de la extensión USB conectada a la Raspberry Pi 4.
5. Conectar uno a uno los adaptadores USB-TTL a la extensión USB conectada a la Raspberry Pi 4.
6. Anotar el valor **/dev/ttyUSB** asociado con cada uno de los sistemas de adquisición de datos. Ver Figura 101.

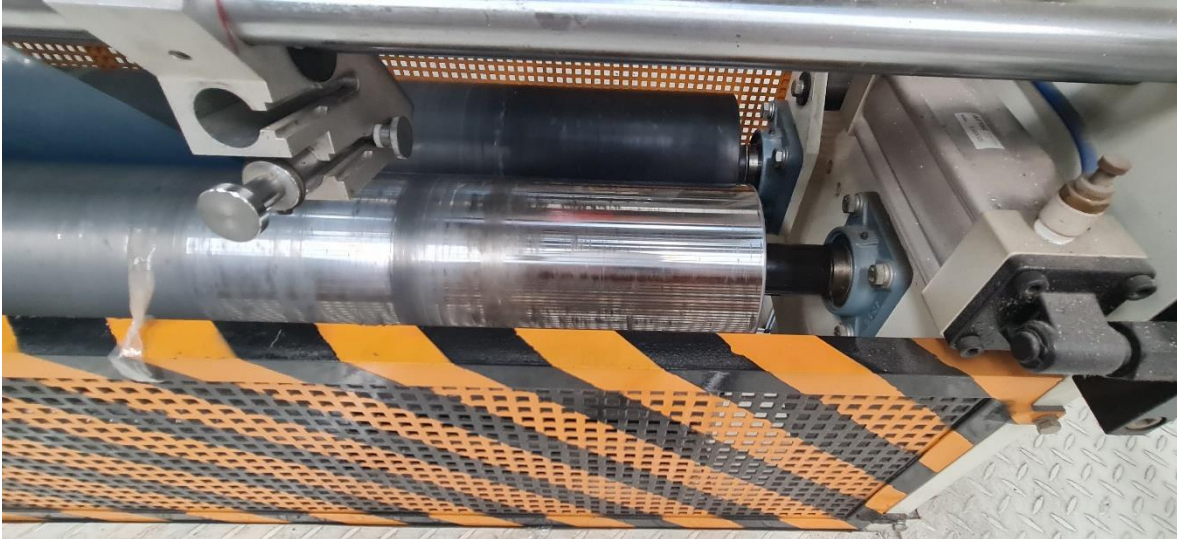
```
Monitoring for USB changes and changes in /dev directory
-----
2021-05-04 18:43:27 - Removed : Bus 001 Device 009: ID 067b:2303 Prolific Technology, Inc. PL2303 Serial Port
2021-05-04 18:43:27 - Removed : Bus 001 Device 008: ID 067b:2303 Prolific Technology, Inc. PL2303 Serial Port
2021-05-04 18:43:27 - Removed : /dev/ttyUSB3
-----
2021-05-04 18:43:28 - Added   : Bus 001 Device 010: ID 067b:2303 Prolific Technology, Inc. PL2303 Serial Port
-----
2021-05-04 18:43:31 - Added   : Bus 001 Device 011: ID 067b:2303 Prolific Technology, Inc. PL2303 Serial Port
2021-05-04 18:43:31 - Added   : /dev/ttyUSB3
```

*Figura 101. Información correspondiente a cada adaptador USB-TTL (pasos 5 y 6). Fuente: Elaboración propia.*

7. Abrir el archivo **interfaz\_E16.py**.
8. Actualizar los valores **/dev/ttyUSB** existentes con los nuevos valores correspondientes. Ver Figura 101.
9. Inicializar la interfaz normalmente.

## Sistema de Adquisición de la Velocidad de Extrusión

El sistema de medición de velocidad estará ubicado en la rejilla de seguridad junto al rodillo de presión y se apoyará suavemente sobre él. Ver Figura 102.



*Figura 102. Rodillo de presión y rejilla de seguridad. Fuente: Elaboración propia.*

Como se describió en la sección Resumen, los elementos básicos del sistema de adquisición de velocidad son un sensor de efecto Hall A3144 [40] capaz de detectar campos magnéticos cercanos a una frecuencia de detección de hasta 200 KHz e imanes permanentes. La salida de dicho sensor es un pulso inverso de igual magnitud que el voltaje con el que está siendo alimentado, esto significa que su salida se mantiene en 5V y cuando es sometido a un campo magnético de suficiente intensidad, su salida baja hasta 0V.

Los imanes utilizados para su activación son imanes permanentes de neodimio de diámetro igual a 5mm por un grosor de 3mm disponibles como kit en la página web de CrCibernética [50].

Estos imanes están incrustados en una rueda de Nylon de 60 mm de diámetro de 15 mm de grosor con 5 agujeros de 5 mm de diámetro por 3 mm de profundo a cada lado de la rueda, separados entre ellos por un ángulo de 72° respecto al centro de la rueda. Ver Figura 103.

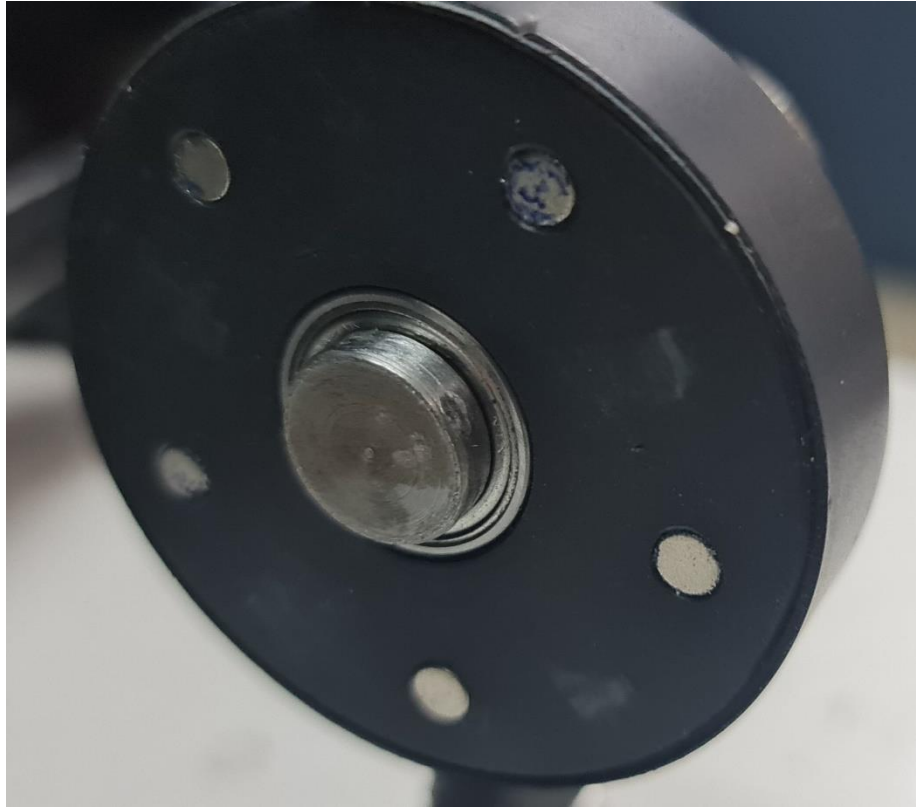


Figura 103. Rueda con imanes incrustados. Fuente: Elaboración propia.

### Sección de Diseño Mecánico del Sistema de Adquisición de Datos de Velocidad

Para la selección del o de los cojinetes se utilizará el catálogo NSK [26]. Si se tiene que el radio de la rueda es 0,03 m y la velocidad máxima 150 m/min entonces la cantidad de revoluciones por minuto a las que girará la rueda será de 796 rpm, esta cantidad puede ser redondeada a 800 rpm como referencia para la vida útil del cojinete.

Según la Figura 95.3., en [3; A26] el factor de velocidad del cojinete para una velocidad de 800 rpm es aproximadamente:  $f_n \approx 0,34$ .

Si se selecciona de manera arbitraria una vida útil de 25000 horas, que corresponde a 2,85 años, el valor del factor de vida frente a la fatiga según la Figura 95.4., en [3; A26] es aproximadamente:  $f_h \approx 3,7$ .

Según la fórmula de catálogo [3; A26]:

$$C = \frac{f_h \times P}{f_n} \quad (3.6)$$

Donde  $P$  corresponde a la carga radial a la que se somete el cojinete y  $C$  es el índice básico de carga. Una vez se sustituyen los valores obtenidos de las Figuras 5.3. y 5.4. [26] se obtiene la siguiente igualdad:

$$C \approx 11xP \quad (3.7)$$

Como nota aclaratoria, redondear estas cifras hacia arriba provoca un factor de seguridad mayor a 1 por lo que no existe inconveniente excepto si la diferencia de precios es considerable entre los cojinetes.

En [5; 555] también se tiene la siguiente fórmula:

$$C_{10} = F_D \left( \frac{L_D n_D 60}{L_R n_R 60} \right)^{1/a} \quad (3.8)$$

Donde  $C_{10}$  corresponde a la clasificación de carga de catálogo y es equivalente al índice básico de carga de la ecuación (1.1),  $F_D$  es la carga radial a la que se somete el rodamiento  $L_R n_R 60$  es el índice básico de vida que según el catálogo de NSK es  $10^6$  revoluciones,  $L_D$  es la cantidad de horas deseada,  $n_D$  es la velocidad deseada en revoluciones por minuto y  $a$  corresponde a 3 para cojinetes de bolas.

Si se sustituyen los valores utilizados anteriormente se obtiene la siguiente igualdad:

$$C_{10} = 10,6x F_D \quad (3.9)$$

Para mantener un margen por exceso se utilizará la igualdad (1.2) obtenida por la fórmula del catálogo junto a un factor de seguridad de 1,2 para así tener:

$$C = 12,75x F_D \quad (3.10)$$

El siguiente paso por seguir es definir la fuerza radial a la que está sometido el rodamiento. En el peor de los casos, todo el peso de la rueda de Nylon estará sobre el rodamiento, y hasta el momento se tiene que la rueda tiene un diámetro de 60 mm por 15 mm de grosor. Las anteriores dimensiones permiten calcular un volumen de  $4,24x10^{-5}m^3$ , de acuerdo con la densidad del Nylon 101 que es  $1150 \text{ kg}/m^3$  [41] la masa total es  $4,877x10^{-2} \text{ kg}$ ; finalmente, la fuerza de gravedad ejercida sobre su centro de masa es de 0,48 N por lo que:

$$C = 12,75x0,48 \text{ N} = 6,1N \quad (3.11)$$

Al tener este dato, se cuenta con la libertad de elegir el rodamiento que cumpla con esta condición y se ajuste al tamaño de la rueda.

A partir del catálogo se ha escogido el rodamiento de bolas abierto 6801 que tiene un diámetro menor de 12 mm y un diámetro mayor de 21 mm; su índice de carga es igual a 1040 N por lo que se tiene un factor de seguridad de 170,5; a su vez se tiene que la carga máxima que puede soportar para cumplir con el objetivo de las 25000 es de  $\frac{1040}{12,75} = 81,57 \text{ N}$  o lo que es equivalente al peso de 8,32kg.

También se puede hacer la estimación de cuanto realmente es la vida esperada de este rodamiento bajo estas condiciones al sustituir el índice de carga del cojinete 6801 y la carga obtenida en la igualdad (1.6) en la ecuación (1.3), y esto resulta en:

$$1040 = 6,1 \left( \frac{L_D * 800 * 60}{10^6} \right)^{1/3} \quad (3.12)$$

Al despejar  $L_D$  de la igualdad (1,7) se obtiene un valor de  $1,03 \times 10^8$  horas o 103 millones de horas que es equivalente a 11758 años aproximadamente.

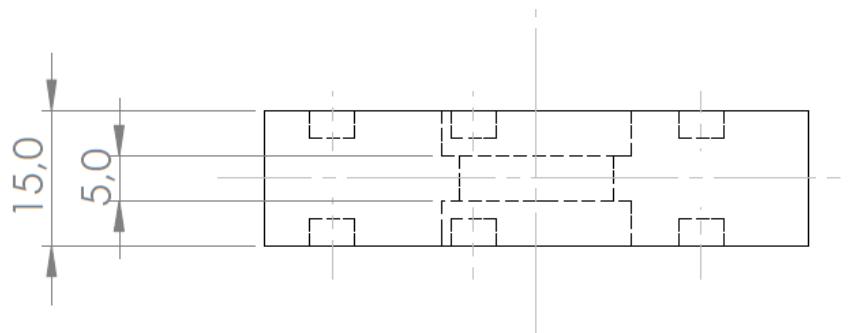
Los cálculos anteriores se realizan en el caso de contar con un solo cojinete sin embargo el NSK 6801 tiene un grosor de apenas 5 mm, por lo tanto, para evitar cualquier tipo de pandeo, se utilizarán 2 cojinetes y de esta manera se mejorará la resistencia ante posibles cargas axiales que no hayan sido consideradas. Dicho lo anterior, el factor de seguridad se duplicaría y el tiempo en horas aumentaría a 523 millones de horas o 59703 años.

Se puede entonces asegurar que, para el diseño actual bajo las condiciones de carga consideradas, la vida útil de los cojinetes seleccionados es virtualmente infinita.

Los rodamientos están claramente sobredimensionados y aunque en el catálogo de NSK existen rodamientos miniatura con índices de carga considerablemente más bajos [3, B40] como el 681 Abierto que tiene un índice de carga de 23N y un diámetro menor de 1 mm, o el 689 Abierto con un índice de carga de 655 y un diámetro menor de 9mm, las dimensiones del eje que permiten podrían provocar pandeo innecesario en el sistema y riesgo mayor de fractura.

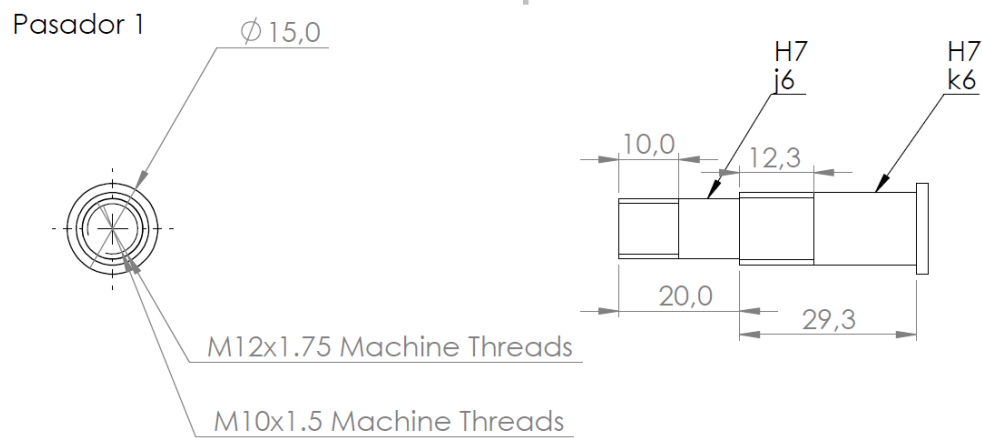
El rodamiento NSK 6801 Abierto [3, B8] tiene un diámetro menor de 12 mm y un diámetro mayor de 21 mm con un grosor de 5 mm. Estas dimensiones son las que dictaran el diámetro y profundidad de la cuenca, y el diámetro del eje para la rueda.

Las cuencas para los rodamientos tienen un diámetro de 21 mm y una profundidad 5mm a cada lado del eje transversal de la rueda. Ver Figura 93.



*Figura 104. Vista con transparencia de la rueda para la adquisición de la velocidad, sin escala. Fuente: Elaboración propia.*

El eje para la rueda tendrá un diámetro de 12 mm y su ajuste será k6 para eliminar el deslizamiento axial de los cojinetes, además tendrá un tope de carrera en uno de sus extremos para eliminar la necesidad de un elemento de sujeción extra, ver Figura 94. El ajuste de las cuencas de la rueda será H7.



*Figura 105. Dimensiones del eje para la rueda, sin escala. Fuente: Elaboración propia.*

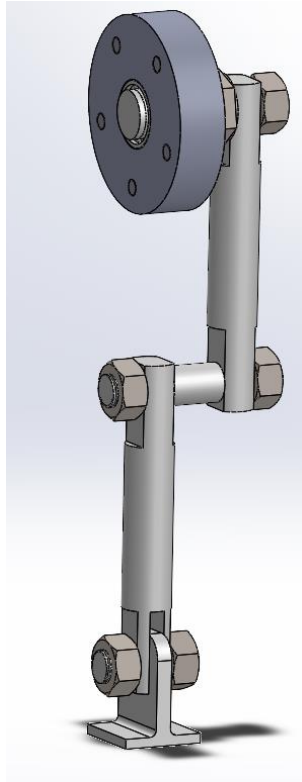
La sujeción de la rueda y los cojinetes se logra con una arandela y una tuerca M12, posteriormente se reduce el diámetro a 10mm para adjuntar el siguiente elemento del montaje.

El cuerpo del montaje está conformado por dos brazos de 10 mm de diámetro cada uno, por dos pasadores que los unen entre ellos, por una base de sujeción que será atornillada a la rejilla y por los elementos estandarizados de sujeción no permanente.

Todas las piezas a excepción de los elementos estandarizados, la rueda y los imanes estarán fabricados en acero 4140 [42]

Utilizando el software SolidWorks [43] se puede simular cada uno de los elementos y obtener la cantidad de masa exacta y el peso exacto de los brazos, de la base, del eje y de los pasadores. Estos valores permiten hacer los cálculos pertinentes para los valores de esfuerzo cortante y normal máximos en los puntos cruciales del ensamblaje.

La simulación (ver Figura 95.) se realizará tomando en cuenta las propiedades mecánicas y físicas del Neodimio [44] para los imanes y del acero SUJ2 [3; A116] [45].



*Figura 106. Ensamblaje simulado en SolidWorks. Fuente: Elaboración propia.*

Según SolidWorks la masa total son 420g por lo que el peso total del conjunto será igual a 4,116 N

La sección del diseño que en el peor de los casos estará sometida a mayor esfuerzo normal debido a la geometría será la unión entre la base y el brazo inferior y la sección sometida a un mayor esfuerzo cortante será el pasador que mantiene en posición a la base y al brazo inferior.

Según el software SolidWorks la menor área transversal del brazo es igual a:  $16,1 \text{ mm}^2$ . Ver Figura 96.

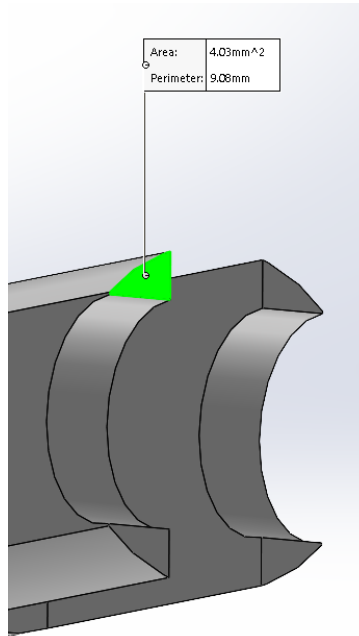


Figura 107. Sección transversal de brazo inferior. Fuente: propia.

Esto representaría un esfuerzo normal de tensión de:

$$\frac{4,116 \text{ N}}{16,1 \text{ mm}^2} = 255652,174 \text{ Pa} \approx 255,7 \text{ KPa} \quad (3.13)$$

Si se toma en cuenta que el límite a la fluencia del acero SAE4140 [42] es 415 MPa, el factor de seguridad para el brazo inferior es igual a 1623.

El área transversal menor en la base es de  $71,44 \text{ mm}^2$  por lo que no es necesario el cálculo del esfuerzo pues el factor de seguridad será aún mayor. Ver Figura 97.

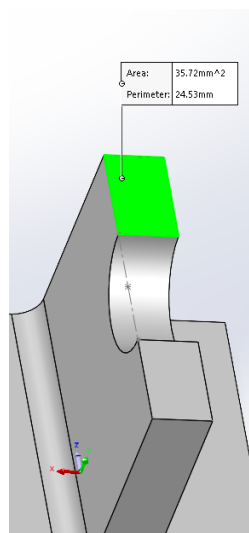


Figura 108. Área transversal menor en la base. Fuente: Elaboración propia.



Ahora, para el esfuerzo cortante máximo, si se utiliza la fórmula descrita en [4: 367], con el radio del pasador inferior en la Figura 98.

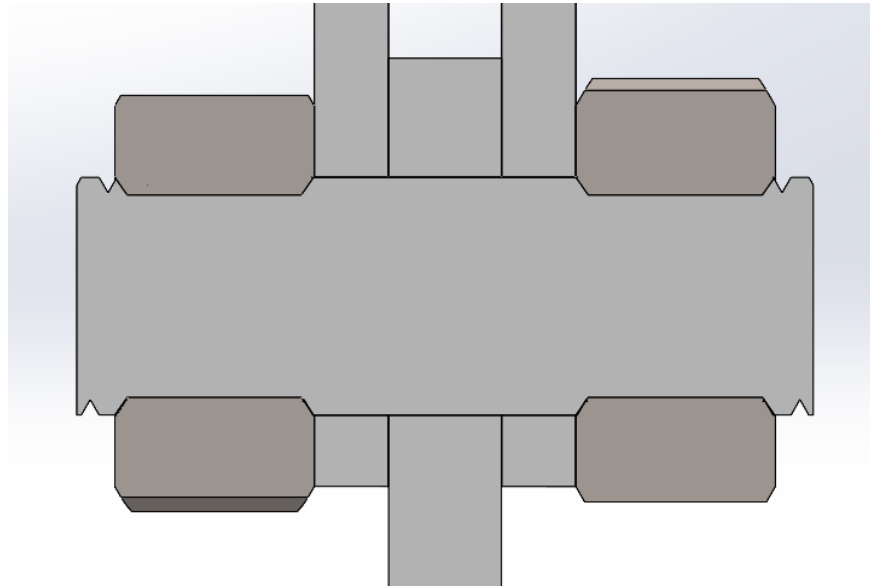


Figura 109. Vista de corte del pasador que sujeta el brazo a la base. Fuente: Elaboración propia.

El radio del pasador es 5mm y la fuerza cortante máxima es 4,116 N, sin embargo, se utiliza únicamente la mitad de dicha fuerza por lo que la ecuación para el esfuerzo cortante es:

$$\tau_{m\acute{a}x} = 3493,7 \text{ Pa} = 3,5 \text{ KPa} \quad (3.14)$$

Donde  $\tau_{m\acute{a}x}$  es el cortante máximo.

Según la teoría del Esfuerzo Cortante Máximo [5], la resistencia máxima a la cortante del acero SAE 4140 está dada por:

$$S_{\tau} = \frac{415 \text{ MPa}}{2} = 207,5 \text{ MPa} \quad (3.15)$$

Por lo tanto, el factor de seguridad en el peor de los casos para el pasador de interés es igual a 59228.

## Sección de Diseño del Circuito Electrónico para el Sistema de Adquisición de Datos de Velocidad

Para la adquisición de la señal se utiliza el ATmega 328p [46] asistido por un cristal de reloj a 16 MHz que comunica la información procesada a una frecuencia de 9600 baudios con el protocolo USART.

Para su programación se utiliza el programa Atmel Studio [47] y la plataforma de desarrollo Arduino R3.

### Código Fuente para el Sistema de Adquisición de la Velocidad

El lenguaje utilizado para la programación del Atmega 328 es el GCC [48] o la Colección de Compiladores GNU por sus siglas en inglés.

La lógica de programación se basa en la utilización de interrupciones; cada vez que se percibe la interrupción provocada por el cambio de estado del sensor de efecto Hall, se captura el valor del **Timer 1** y se envía a través de USART.

En las primeras líneas se incluyen las bibliotecas **io.h** [49], **interrupt.h** [49] y **stdio.h** [50]

- **#include <avr/io.h>**: permite la utilización de los pines de entrada y salida del microprocesador Atmega 328.
- **#include <avr/interrupt.h>**: permite la utilización de las interrupciones internas e internas del microprocesador Atmega 328.
- **#include <stdio.h>**: permite las operaciones estándar de entrada y salida.

Posteriormente se definen las variables **F\_CPU**, **BAUD** y **BRC** que están relacionadas con la transmisión de datos a través de USART.

- **#define F\_CPU 16000000**: define la frecuencia de operación del microprocesador a 16 MHz.
- **#define BAUD 9600**: define los baudios de comunicación a 9600.
- **#define BRC ((F\_CPU/16/BAUD) – 1)**: calcula el contenido del registro **UBRRn** según la fórmula de la hoja de datos [46: 182].

Además, se importa la biblioteca **USART\_irq.h** [51], ver Anexo A y B para las bibliotecas y funciones utilizadas.

Seguidamente se definen las variables que se utilizarán:

- **timer1**: de tipo entero largo, para poder almacenar el valor de los registros del temporizador **Timer 1**.
- **mystdout**: de tipo archivo estático; se utiliza como puntero en memoria para el envío de los datos por USART. Se inicializa con la función

***USART0\_Transmit\_IO*** [51] la cual se encuentra en la biblioteca ***USART\_irq.h***.

## Interrupciones

Además de lo descrito anteriormente respecto al cambio de estado del sensor de efecto Hall, también se requiere la detección del momento en que la rueda se detiene para poder activar una alarma. Este proceso de detección se lleva a cabo a partir del microprocesador con la ayuda de la interrupción por bandera de desbordamiento del ***Timer 1*** el cual se ha programado con un factor pre escalado de 1024 [46: 74]

Las siguientes líneas declaran las rutinas de interrupción:

- ***ISR(INT0\_vect){}***: Se utiliza el vector ***INT0\_vect*** para detectar la interrupción por cambio de estado del sensor de efecto Hall en el pin ***PD2*** en la patilla 4 [46].

Cada vez que se ejecuta esta rutina, se cambian los estados de los ***PB4*** y ***PB5*** que están conectados físicamente a un LED RGB que sirve a manera de identificador visual; se asigna el contenido del registro ***TCNT1*** [46: 120] a la variable ***timer1***, la información contenida en esta variable se envía en formato hexadecimal, utilizando la línea ***printf("%1X\n", timer1)*** y finalmente se restablece el contenido de los registros ***TCNT1H*** y ***TCNT1L*** a cero para reiniciar el conteo del tiempo.

- ***ISR(TIMER1\_OVF\_vect){}***: Se utiliza el vector ***TIMER1\_OVF\_vect*** para detectar la bandera por desbordamiento del ***Timer 1***. De acuerdo con la frecuencia de 16 MHz y teniendo en cuenta que los registros del ***Timer 1*** suman 16 bits entre ambos, el desbordamiento del temporizador sucede a los 4,19 segundos. Este valor dicta una velocidad mínima medible de 0,5 m/min; bajo consenso con la gerencia, una velocidad menor a esta indica que la rueda está funcionalmente detenida.

Cuando se entra a esta rutina, se hace parpadear un LED conectado al pin ***PB5*** y se envía la letra "g" a través de USART para indicar a la interfaz que la rueda está por debajo de la velocidad mínima o funcionalmente detenida.

## Inicialización

En la secuencia principal se definen las entradas y salidas y se configuran los registros que controlan al temporizador, a las interrupciones y la comunicación USART.

Las salidas se definen con las líneas:

- **`DDRB |= (1<<DDB4);`**
- **`DDRB |= (1<<DDB5);`**

Estas dos líneas configuran los pines **PB4** y **PB5** como salidas y las siguientes líneas configuran el estado inicial de dichas salidas en cero:

- **`PORTB &= ~(1<<PORTB5);`**
- **`PORTB &= ~(1<<PORTB4);`**

La entrada para el pulso del sensor de efecto Hall se define con la siguiente línea:

- **`DDRD &= ~(1 << DDD2);`**

Y su estado inicial en cero se define con:

- **`PORTD |= (1 << PORTD2);`**

Seguidamente, los registros **EICRA** y **EIMSK** se encargan de configurar la interrupción en **PD2** disparada en flanco decreciente [46: 80] y de asignar la interrupción a **INT0** [46: 81] respectivamente.

A continuación, la asignación del registro **TIMSK1** [46: 144] habilita la interrupción por desbordamiento; la asignación de **TCCR1A** [46: 140] configura el temporizador bajo operación normal y la asignación del registro **TCCR1B** [46: 143] configura el factor de pre-escala a 1024.

La línea:

- **`stdin=stdout=&mystdout;`**

Asigna el envío de datos seriales a través de USART con ayuda de las funciones en la biblioteca **USART\_irq.h**.

A los registros **UBRR0H** y **UBRR0L** [46: 182] se les asigna el valor definido anteriormente de **BRC** para definir la velocidad de transmisión a los 9600 baudios.

La asignación al registro **UCSR0B** habilita la recepción y transmisión de datos, y **UCSR0C** configura los paquetes enviados como de 8 bits.

Finalmente, se habilitan las interrupciones con **sei()** y se inicializa un bucle infinito.

## **Sistema de Adquisición del Ancho del Plástico Extruido por la Extrusora 16**

El sistema de medición del ancho del plástico extruido funciona a partir de la medición precisa de la cantidad de pasos ejecutados por un motor a pasos de 2 bobinas y del inicio y del final de la sección transversal de la película plástica detectada por un sensor foto receptivo de infrarrojos.

Ambos elementos se mueven en conjunto para mantener una medida precisa del ancho, la cual se puede extraer a partir de la fórmula matemática de la circunferencia del círculo, en este caso, la polea conductora de la cinta fija.

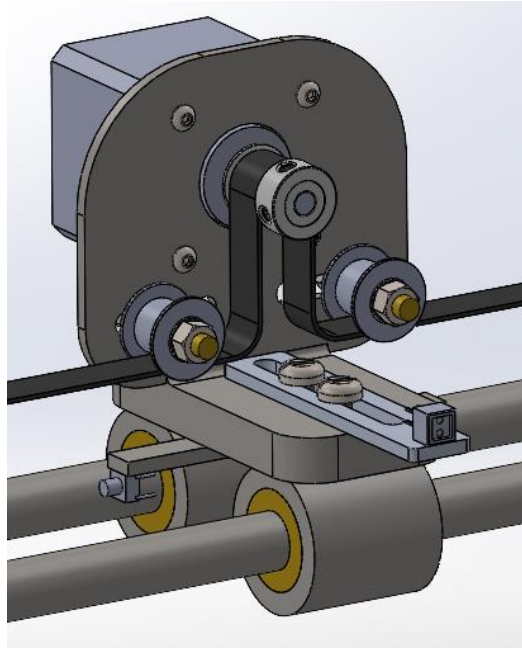
La medida es interpretada por un microprocesador Atmega 328p y posteriormente es comunicada a través de protocolo USART hacia la Raspberry Pi 4.

### **Sección de Diseño Mecánico del Sistema de Adquisición de Ancho**

El motor a pasos utilizado es el JK42HS40-1304F disponible en [50] y según la misma página, su resolución es de 200 pasos por revolución. A su eje se le adjunta en su extremo una polea conductora para banda de tipo G2T de 10mm con una circunferencia experimental de 4,1 cm (este dato fue obtenido sobre el prototipo pues no existe documentación oficial sobre las poleas adquiridas). Para la tensión necesaria de la cinta se utilizan poleas conducidas G2T de 10mm.

En la base del carrito se apoyan el motor, la polea conductora, las poleas conducidas, los finales de carrera, la pletina para el sensor y el sensor CNY70. En la parte inferior, dos cilindros de acero permiten la inserción de dos bujes de bronce que facilitan el deslizamiento libre del carrito sobre las barras.

La masa total del carrito es de 741 g según el software SolidWorks [43], si a esto se le suma la masa del motor (280 g), el total es de 1021 g y corresponde a un peso de 10,02 N.



*Figura 110. Modelo 3D del carrito completo. Simulado en SolidWorks [43]. Fuente: Elaboración propia.*

Las barras utilizadas tienen una extensión de 1650 mm (se pierden 40 mm en los soportes) y un diámetro de ½ pulgada o 12,7 mm y su material es acero AISI 4140.

En cada extremo de las barras, se posiciona una base de aluminio encargada de mantenerlas a una distancia fija entre ellas, de sujetar la cinta G2T y de unir el ensamblaje con cada una de las barras de soporte.

El análisis estático en las barras se realiza según la técnica de integración para vigas en [4]. Tomando en cuenta la densidad del acero AISI 4140 [42], se puede posicionar una carga linealmente distribuida de aproximadamente 9,76 N/m.

El valor de la fuerza cortante en función de  $x$  según la posición,  $a$ , del carrito es predicha por la función:

- Previo al carrito:

$$V(x) = 10,36 - 9,76x \quad (3.16)$$

- Posterior al carrito:

$$V(x) = 5,35 - 9,76a - 9,76(x - a) \quad (3.17)$$

A partir de ellas se puede obtener la función del momento flector:

Previo al carrito:

$$M(x) = 10,36x - 4,88 x^2 \quad (3.18)$$

Posterior al carrito:

$$M(x) = 10,36a - 4,88a^2 - 5,35x - 9,76a(x - a) - 4,88(x - a)^2 \quad (3.19)$$

Donde 'a' representa la posición del carrito, en metros, sobre las barras y V(x) es el valor en Newtons de la fuerza cortante respecto de la distancia 'x', en metros, y M(x) es el valor del momento flector en Newtons-metro.

Con ayuda de las bibliotecas matplotlib [52] y NumPy [53] disponibles para Python [54], las funciones anteriores se pueden simular en un espacio de tres ejes para todos los valores entre 0 y 1,610 m. El cambio de variable (x-a) es necesario a la hora de graficar con un origen en común.

La simulación genera los siguientes gráficos:

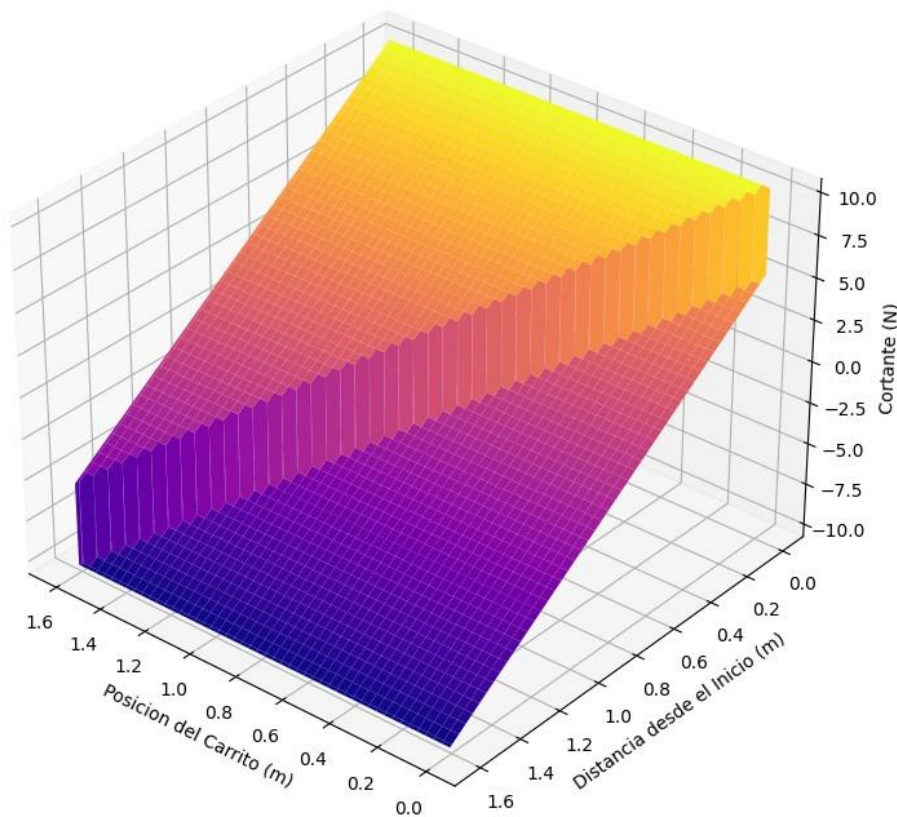


Figura 111. Representación de la Fuerza Cortante a lo largo de una barra de soporte según la posición del carrito. Generado con Python [54]

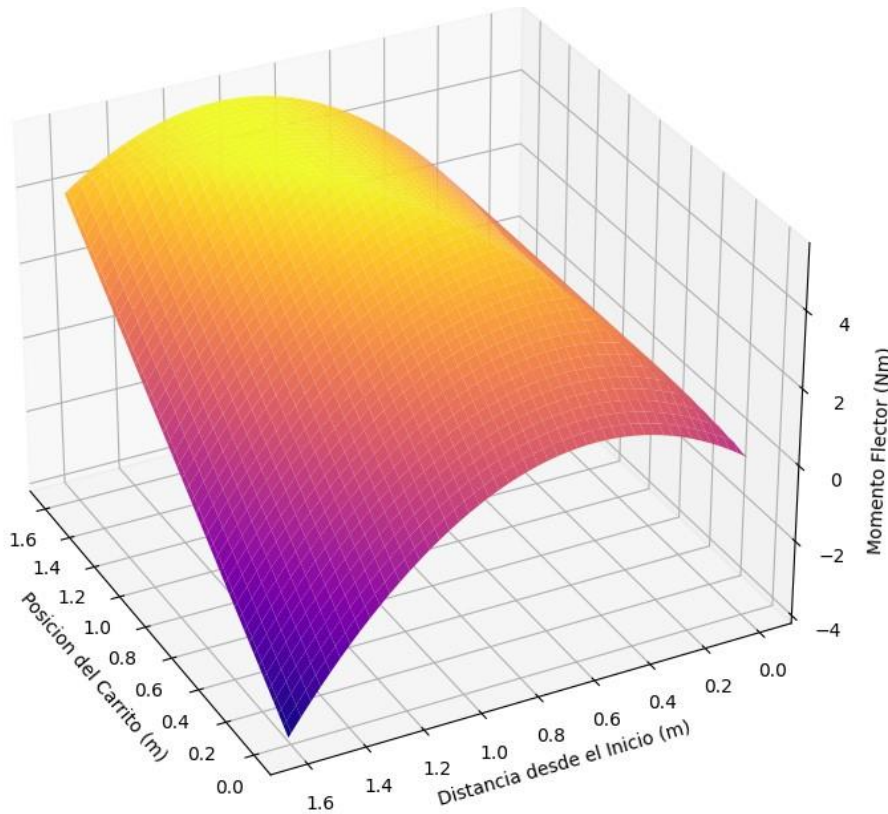


Figura 112. Representación del Momento Flector a lo largo de una barra de soporte según la posición del carrito. Generado con Python [54]

Adicionalmente, a partir del análisis computacional anterior, se pueden obtener las siguientes coordenadas para la Cortante y Momento Flector mínimos y máximos con una resolución de  $\pm 2\text{mm}$ :

- La fuerza cortante mínima es de  $-10,34\text{ N}$  en la posición de  $1,608\text{ m}$  cuando el carrito está posicionado a los  $0,352\text{ m}$ .
- La fuerza cortante máxima es de  $10,36\text{ N}$  en la posición de  $0\text{ m}$  cuando el carrito está posicionado al inicio de su recorrido.
- El valor de momento flector mínimo es de  $-4,02\text{ Nm}$  y sucede en la posición de  $1,608\text{ m}$  cuando el carrito está posicionado al inicio m.
- El valor de momento flector máximo es de  $5,50\text{ Nm}$  y sucede en la posición de  $1,602\text{ m}$  cuando el carrito está a  $1,602\text{ m}$  del inicio.

Utilizando la fórmula para el esfuerzo normal producido en una viga por el momento flector [4: 291] se puede obtener el valor de los esfuerzos en tensión y compresión experimentados por cada barra y utilizando la fórmula del esfuerzo cortante [4: 32] se puede obtener el valor del esfuerzo cortante. Estos datos serán útiles para conocer el factor de seguridad al que operan las barras de deslizamiento.



Según [4: 291] el valor de la magnitud del esfuerzo tanto de compresión como de tensión en el punto de mayor esfuerzo es de 27,3 MPa. El factor de seguridad para esfuerzo normal de las barras es de 15,17.

Según [4: 32] el valor del esfuerzo cortante en el punto de mayor esfuerzo es de 81,78 KPa. El factor de seguridad para esfuerzo cortante de las barras es 5074,4.

Debido a la alta relación del factor de seguridad y a que en el caso del momento flector las condiciones para que el mayor esfuerzo se dé, rara vez se cumplen debido a la función del sistema de adquisición de datos, se omitirá el análisis respectivo a la fatiga del sistema.

Las barras laterales que soportan el ensamblaje completo tienen un diámetro de 25,4 mm y una extensión de 500 mm. En el peor de los casos, si se utiliza la extensión completa, la zona expuesta a fuerza cortante y a momento flector tiene una extensión de 460 mm en cada una.

De acuerdo con SolidWorks, el peso del ensamblaje es aproximadamente 81,7 N, sin tomar en cuenta los soportes de pared y sus tornillos, las poleas para la cinta G2T, la cinta G2T y los sensores. Si además de eso, se toma en cuenta su distribución lineal de peso (39 N/m), las ecuaciones que describen la fuerza cortante interna y el momento al que las barras laterales de soporte están sometidas, son:

- Fuerza cortante:

$$40,85 - 39x \quad (3.20)$$

- Momento Flector:

$$40,85x - 19,5 x^2 \quad (3.21)$$

Donde  $x$  es la distancia en metros desde el inicio de la barra.

Debido a que no existe ninguna variación de la distribución del peso en toda su extensión, la posición del carrito es irrelevante. Los valores máximos para las barras son los siguientes:

- La intensidad de la fuerza cortante máxima es de 40,85 N.
- El momento flector máximo es de 15,55 Nm.

El valor del esfuerzo cortante en el punto de mayor fuerza cortante es de 80,62 KPa y el esfuerzo normal máximo, tanto de compresión como tensión es de 9,67 MPa.

La operación de estos elementos tiene un factor de seguridad ante esfuerzo normal de 42,9 y un factor de seguridad ante esfuerzo cortante de 2574.

## Sección de Diseño del Circuito para el Sistema de Adquisición de Ancho

La sección electrónica del sistema de adquisición de datos de ancho del producto extruido por la máquina 16 está conformada por tres etapas:

- Adquisición, comunicación y toma de decisiones.
- Control del motor.
- Transporte de potencia y señales.

### Adquisición, Comunicación y Toma de Decisiones

Al igual que para el circuito del sistema de adquisición de velocidad, esta etapa consta de un microprocesador ATmega 328p, un kit con cristal de reloj a 16 MHz.

El listado de entradas y salidas es el siguiente:

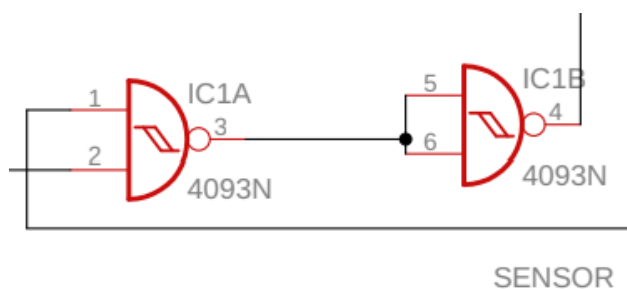
- **PD0 y PD1 (RX/TX)**: se utilizan para la comunicación con la Raspberry Pi 4 a través de un adaptador TTL-USB tipo PL2303HXA [55].
- **PD2**: recibe la señal proveniente del final de carrera derecho. También está conectado a un LED para visualizar el evento y a un interruptor para controlar la interrupción de manera manual.
- **PD3**: recibe la señal proveniente del final de carrera izquierdo. También está conectado a un LED para visualizar el evento y a un interruptor para controlar la interrupción de manera manual.
- **PB1**: recibe la señal proveniente del disparador Schmitt encargado de filtrar la señal del sensor CNY70
- **PB4**: controla la frecuencia de pasos del motor con un pulso modulado hacia el controlador del motor. También conectado a la terminal azul de un LED RGB para la visualización de la señal.
- **PB5**: controla la dirección del motor con una señal binaria hacia el controlador del motor. También conectado a la terminal roja de un LED RGB para la visualización de la señal.
- **PB6 y PB7**: cada uno está conectado a un terminal distinto del cristal de reloj y a su vez cada uno está separado de la referencia por un capacitor no electrolítico de 22 pF.
- **PC0**: conectado a la terminal verde de un LED RGB para la visualización del estado del sensor.

Parte importante de esta misma etapa es el filtrado de la señal proveniente del sensor CNY70. Este filtrado se logra a partir de una compuerta NAND 4093B con función de disparador Schmitt; la compuerta recibe una entrada **VERDADERA** cuando el valor del voltaje en uno de sus pines de entrada sobrepasa los 3,3 V y retorna a una entrada **FALSA** cuando el valor del voltaje en ese mismo pin alcanza un voltaje inferior a los 1,1V.

La descripción de la utilización de sus pines es la siguiente:

- **Pin 1:** Está conectado directamente al sensor.
- **Pin 2:** Está conectado a la alimentación de 5 V.
- **Pin 3:** Corresponde a la salida de la primera compuerta NAND. Su valor es **FALSO** cuando la entrada del sensor es **VERDADERO**.
- **Pines 5 y 6:** Ambos pines están en corto circuito entre ellos y reciben la salida del **Pin 3**.
- **Pin 4:** Es la salida de la segunda compuerta, se encarga de invertir el valor de la primera compuerta, está conectada al pin **PB1**. Su salida es **VERDADERO** cuando la entrada desde el sensor es **VERDADERO**.

Esta conexión se puede observar gráficamente de la siguiente manera:



*Figura 113. Representación gráfica de la conexión del disparador Schmitt. Fuente: propia, confeccionado en EAGLE [56].*

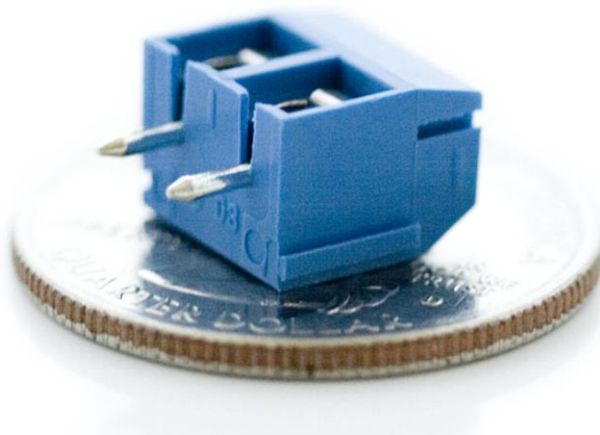
El sensor CNY70 requiere de ciertos parámetros eléctricos para funcionar correctamente:

- El ánodo del LED se conecta directamente a la alimentación de 5 V, el cátodo del LED se conecta a una resistencia variable de 1Kohm que está referenciada en su otro extremo, esta resistencia se encarga de limitar la corriente de entrada al LED y regula directamente la intensidad luminosa de dicho elemento. En pruebas se encontró que, para el funcionamiento nocturno del sistema, el valor adecuado de esta resistencia es aproximadamente 70 Ohmios y para el funcionamiento diurno se pueden utilizar 470 Ohmios, sin embargo, la condición determinante para la intensidad del LED es el funcionamiento nocturno.
- El colector del fototransistor se conecta directamente a la alimentación y su emisor se conecta a una resistencia variable de 1Mohm. Este parámetro regula directamente la sensibilidad al ruido del sensor y la amplitud de salida en Voltios. En pruebas se determinó que el valor más adecuado para operación diurna son 4 Kohm, en pruebas nocturnas se utilizó hasta 1Mohm sin embargo la condición determinante para su valor es el funcionamiento diurno.

Tanto el emisor como el ánodo cuentan con una resistencia de 47 Ohms en serie para la protección del LED y el transistor.

### Control del Motor

Esta etapa debe ser alimentada por separado debido a la carga de potencia, su alimentación proviene directamente de un convertidor de corriente directa LM2596 [56] conectado a una terminal atornillada como la de la Figura 103.



*Figura 114. Terminal atornillada. Fuente: tomado de [50].*

La terminal negativa se conecta a la referencia de la fuente directa y a la tierra común de la tarjeta impresa del circuito, la terminal positiva alimenta el controlador del motor, el sensor CNY70 y los interruptores de final de carrera.

El control del motor se realiza con la ayuda de un controlador para motores a pasos EasyDriver [57]. Dicho controlador recibe dos entradas desde el microprocesador, estas entradas controlan el paso, la velocidad y la dirección del motor. Adicionalmente, el controlador permite modificar la cantidad de pasos que se requieren para una vuelta completa.

La descripción de la conexión de los pines utilizados es la siguiente:

- **M+**: Conectado directamente a la alimentación.
- **DIR**: Conectado al pin **PB5**. En este pin se controla la dirección.
- **STEP**: Conectado al pin **PB4**. En este pin se controlan el paso y la frecuencia del motor.
- **GND1**: Conectado al terminal negativo de la terminal atornillada y a la referencia común de la placa impresa.
- **A1**: Salida a la bobina A del motor a través del cable rojo del motor a pasos.
- **A2**: Salida a la bobina A del motor a través del cable azul del motor a pasos.

- **B1:** Salida a la bobina B del motor a través del cable negro del motor a pasos.
- **B2:** Salida a la bobina B del motor a través del cable verde del motor a pasos.
- **MS1:** conectado a un interruptor de dos posiciones que permite modificar la cantidad de pasos requeridos para una vuelta. La posición predeterminada es conectada al neutro (hacia arriba).
- **MS2:** conectado a un interruptor de dos posiciones que permite modificar la cantidad de pasos requeridos para una vuelta. La posición predeterminada es conectada al neutro (hacia arriba).

## Transporte de Potencia y Señales

El transporte de las señales del sensor, de los interruptores de final de carrera y la potencia del motor se realiza a través de dos cables apantallado de 8 hilos / 4 pares cada uno.

La conexión de la placa hacia el cable es posible gracias a un adaptador macho de 2 x 5 pines, conectado a un cable listón hembra 2 x 5.

Debido al ruido provocado por las señales de alta frecuencia y potencia dirigidas hacia el motor, los 4 hilos conectados a las bobinas deben estar en un cable separado del que contiene los 6 hilos que transportan las señales eléctricas del sensor CNY70 y los interruptores de final de carrera del carrito.

Por razones estéticas, el cable está protegido por una malla plástica negra retráctil y sujetado en sus extremos por gazas plásticas.

## Código Fuente para el Sistema de Adquisición de Ancho

La configuración inicial es exactamente igual a la realizada para el sistema de adquisición de velocidad sin embargo las variables utilizadas son distintas:

- **limite:** esta variable dicta cuantos pasos recorrerá el carrito antes de cambiar de dirección en caso de que no se detecte la filmina o se llegue al final de la carrera.
- **pasos:** esta variable se utiliza para contar los pasos ejecutados a partir de la interrupción entregada por la compuerta NAND 4093B.
- **cont\_global:** esta variable almacena la cantidad total de pasos mientras no se detecte la presencia de la filmina.
- **medicion:** mantiene el estado en **VERDADERO** si se detecta el inicio de la medición.

## Interrupciones

Para el correcto funcionamiento del sistema de adquisición de ancho, se requieren cuatro rutinas de interrupción:

- **ISR(TIMER1\_COMPA\_vect)**: esta rutina de interrupción es activada por la bandera de comparación A del **Timer 1**. Este valor se configura en la función **PWM\_Init**.

Dentro de esta rutina se realiza el conteo de los pasos y se define el estado del pin **PB4** a **VERDADERO**.

- **ISR(TIMER1\_COMPB\_vect)**: esta rutina es activada por la bandera de comparación B del **Timer 1**. Este valor se configura en la función **PWM\_Init**.

En ella se cambió el estado de **PB4** a **FALSO**.

- **ISR(PCINT0\_vect)**: esta rutina es activada por el cambio de estado del pin **PB1**.

Dentro de ella se llevan a cabo tres validaciones:

- Si el cambio de estado se da cuando el pin **PB1** cambia a positivo: en este caso se reinicia el contador **pasos**, se cambia el estado del pin **PC0** a encendido y se cambia el estado de **medicion** a **VERDADERO**.
- Si el cambio de estado se da después de que el motor haya completado un cuarto de vuelta y aún no haya llegado al límite de pasos: en este caso el estado de **PC0** se cambia a apagado, se cambia el estado de **PB5** para cambiar la dirección de movimiento del carrito, se envía el número de pasos en formato hexadecimal, se reinicia el número de pasos y finalmente se cambia el estado de **medicion**.
- Si el cambio de estado se da cuando el número de pasos excede el límite preestablecido: se cambia el estado de **PC0**, se envía el número de pasos por USART en formato hexadecimal, a la variable **cont\_global** se le asigna el valor de **pasos** antes de reiniciar el conteo y finalmente se cambia el estado de medición. La razón por la que se asigna el valor de **pasos** a **cont\_global** es debido a que el bucle encargado del cambio de dirección automático es el encargado de comparar el valor de **cont\_global** con **limite** por lo que el cambio de dirección no se realiza dentro de esta rutina.
- **ISR(INT1\_vect)**: Se dispara cuando se recibe la interrupción externa **INT1**. Esta interrupción corresponde al accionamiento del límite de banda izquierdo, cuando se detecta esta señal el estado de **PB5** cambia a positivo, lo que hace mover el carrito hacia la derecha.

- **ISR(INT0\_vect):** Se dispara cuando se recibe la interrupción externa **INT0**. Esta interrupción corresponde al accionamiento del límite de banda derecho, cuando se detecta esta señal el estado de **PB5** cambia a positivo, lo que hace mover el carrito hacia la izquierda.

## Inicialización

Para la producción del pulso de amplitud modulada, es necesaria la creación de una función aparte y la configuración del temporizador en modo comparación de registro. Lo descrito se realiza dentro de la subrutina:

- **PWM\_init():**

El contador **Timer 1** se configura en modo pulso rápido de amplitud modulada con los bits **WGM11** y **WGM10** [46: 141] y **WGM12** [46: 142]. Al temporizador se le asigna un factor de pre-escala de 64 [46: 143].

Las siguientes líneas de código se encargan de que el pulso tenga una amplitud de 600 bits en 'alto' y 200 bits en 'bajo'; esto representa un periodo de 800 bits para el pulso completo a un factor de pre-escala de 64 a 16 MHz, lo que significa que la frecuencia de pulsos es de 312,5 Hz y según las propiedades del motor [50] esto es igual a 93,75 rpm lo cual se encuentra dentro de la operación aceptable del motor. Si se toma en cuenta la circunferencia de la polea directora de la cinta, la conversión sugiere que el carrito se mueve a una velocidad de 3,84 m/min.

Dentro de la rutina principal se configuran los pines de entrada y salida según lo descrito en la sección Adquisición, Comunicación y Toma de Decisiones.

Además, se configuran las interrupciones por cambio de pin con los registros **PCICR** [46: 82] para habilitar la interrupción por cambio de pin 0, y **PCMSK0** [46: 83] para asociar dicha interrupción al pin correspondiente.

Las interrupciones externas se configuran con el registro **EICRA** [46: 80] para ser disparadas por un flanco positivo en los interruptores de final de carrera, y con el registro **EIMSK** [46: 81] para habilitar ambas interrupciones: **INT0** e **INT1**.

Lo correspondiente a la comunicación serial, se configura exactamente igual que para el Sistema de Adquisición de Datos de Velocidad.

La rutina siguiente corresponde a un bucle infinito en el que se compara constantemente la cantidad almacenada por **cont\_global** respecto a **limite**, si **cont\_global** alcanza o sobrepasa la misma cantidad, la dirección del motor se invierte y a través de comunicación serial se envía el carácter "I", esto notifica a la Raspberry Pi 4 que el sistema no ha encontrado la presencia del plástico.

## Capítulo 5: Resultados y Análisis

### Sistema de Adquisición de Datos de Velocidad

La propuesta de diseño se sometió a repetidas pruebas para asegurar que su funcionamiento es satisfactorio y representa una verdadera solución al problema. En el caso del sistema de medición de velocidad, las variables más importantes para su funcionamiento son:

- La detección de cada pulso del sensor de efecto Hall.
- El funcionamiento en la posición deseada.
- Que el contacto entre el rodillo y el sistema permita el giro sin deslizamiento.
- La comunicación entre el Atmega 328 y la Raspberry Pi 4.

Es importante señalar que la realización de pruebas para el sensor de velocidad en la máquina está limitada por la velocidad de producción a la que esté funcionando, por lo que se dará por aprobado el resultado a la velocidad disponible y no necesariamente a la velocidad máxima de producción.

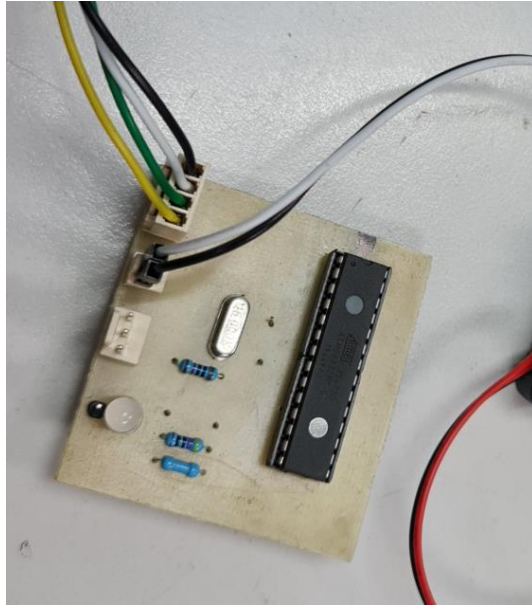
### Detección de Cada Pulso del Sensor de Efecto Hall

La prueba para determinar la detección de los pulsos generados por el movimiento cercano de los imanes de neodimio se puede realizar por inspección tanto visual como con la ayuda de un osciloscopio.

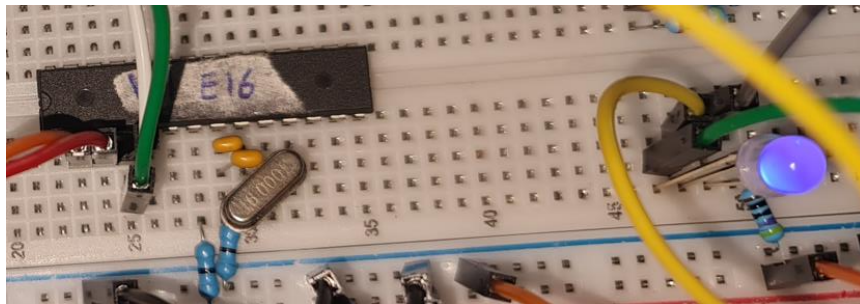
En pruebas preliminares, se encontró que los imanes de neodimio deben estar orientados de cierta manera respecto al sensor de efecto Hall y que la distancia de detección es de alrededor de 3 mm. Estas dos propiedades se tomaron en cuenta para el diseño de la rueda y del brazo.

Para determinar visualmente si el sensor A3144 de efecto Hall detecta los pulsos, se conectó un LED a su patilla número 3 con una resistencia de 470 Ohms referenciada a tierra y una resistencia de 10 Komhs entre las patillas 1 y 2. En estado inactivo, el LED debería estar encendido y cuando el sensor detecta el pulso, debería de apagarse.





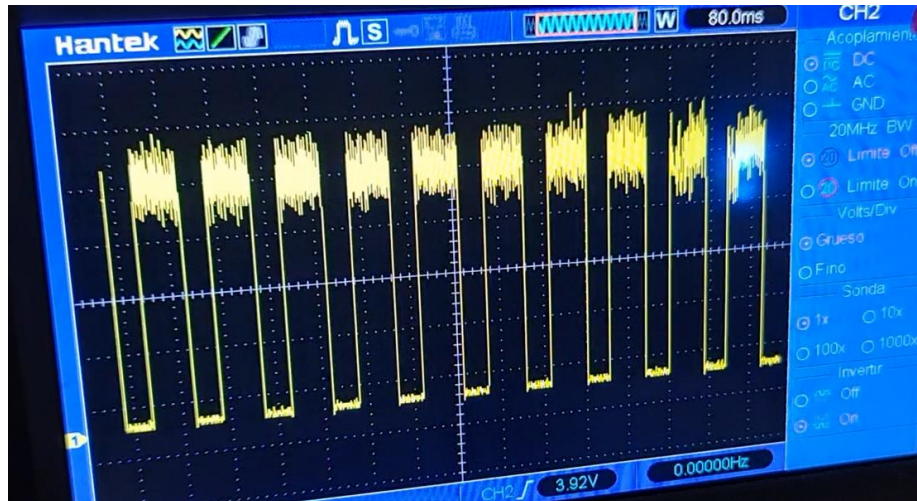
*Figura 115. Circuito del sistema de adquisición de velocidad. Fuente: elaboración propia.*



*Figura 116. Prototipo del circuito para la visualización del pulso, el cable verde transporta la señal del sensor A3144. Fuente: elaboración propia.*

En la Figura 115., se puede observar el LED utilizado para la inspección visual, los pines libres corresponden a los que permiten la alimentación y el transporte de señales hasta y desde el sensor A3144.

En todas las pruebas realizadas sobre el circuito, se pudo observar el parpadeo del LED correspondiente a cada pulso, además, se utilizó un osciloscopio Hantek DSO4202C para observar la señal eléctrica.



*Figura 117. Observación de los pulsos registrados por el sensor de efecto Hall.  
Fuente: elaboración propia.*

En la Figura 116., se puede observar la presencia de ruido en el pulso dado por el sensor A3144, sin embargo, ese no representó ningún tipo de problema debido a que el voltaje necesario para registrar un 1 lógico en el Atmega 328p está alrededor de los 2,5V [25: 272]. Además, se puede notar que la pendiente del flanco es virtualmente vertical por lo que, para fines prácticos, la frecuencia de operación es virtualmente infinita. Sin embargo, según [9] el periodo de operación del sensor es de 4  $\mu$ s, lo que corresponde a una frecuencia de 250 kHz, para fines del proyecto, esto corresponde a una velocidad máxima de producción de 9739 m/s o 584336 m/min; este valor está muy por encima del límite requerido.

En una ocasión, se notó que la amplitud del pulso era muy baja y no era detectado por el microprocesador; luego de una cuidadosa revisión de los elementos electrónicos, se llegó a la conclusión de que el problema era generado por un corto circuito en el LED RGB de ánodo común, por lo que la resistencia que transporta la corriente desde el sensor hasta el LED se removió, al hacer este cambio, la generación de los pulsos volvió a la normalidad.

La operación del sensor de efecto Hall queda validada por las pruebas.

### **Funcionamiento en la Posición Deseada**

En esta sección también se analizará el contacto entre el rodillo de presión y la rueda del sistema de adquisición de datos de velocidad.

La rueda se construyó de Nylon, en pruebas preliminares donde se probó el contacto entre la superficie de acero del rodillo de presión y la circunferencia de la rueda, se observó que la fricción entre ellas podría no ser suficiente para garantizar

medidas precisas de velocidad, ver Figura 118., por lo que se decidió agregarle una cinta de caucho alrededor de toda su circunferencia, ver Figura 119.



*Figura 118. Prueba preliminar de fricción entre la rueda y el rodillo de presión.  
Fuente: elaboración propia.*



*Figura 119. Adición de la cinta de caucho alrededor de la rueda. Fuente:  
elaboración propia.*

La adición de esta cinta de caucho a la rueda asegura que el contacto sea el mejor posible en todo momento para que no haya deslizamiento entre las superficies y que la medida sea lo más exacta posible. En las pruebas realizadas posteriormente, no se observó ningún tipo de deslizamiento ni golpeteo provocado por la discontinuidad de la cinta. Sin embargo, fue necesario cambiar la dimensión del radio de 3 cm a 3,1 cm en las funciones programadas.



*Figura 120. Rueda de imanes en la posición final. Fuente: elaboración propia.*

### **Comunicación entre el Atmega328P y la Raspberry Pi 4**

La comunicación entre el Atmega328 P es de vital importancia pues esta etapa transporta los datos desde el dominio físico hasta el virtual; una vez que los datos hayan sido obtenidos por la Raspberry Pi 4, se pueden disponer y almacenar.

Las pruebas continuas se llevaron a cabo con la utilización de un adaptador TTL-Serie genérico de tipo PL2303, este adaptador se conecta al puerto USB de la Raspberry Pi 4 B y su terminal RX se conecta a la patilla TX del Atmega 328p; la frecuencia de comunicación se configuró a los 9600 Baudios en ambos elementos.

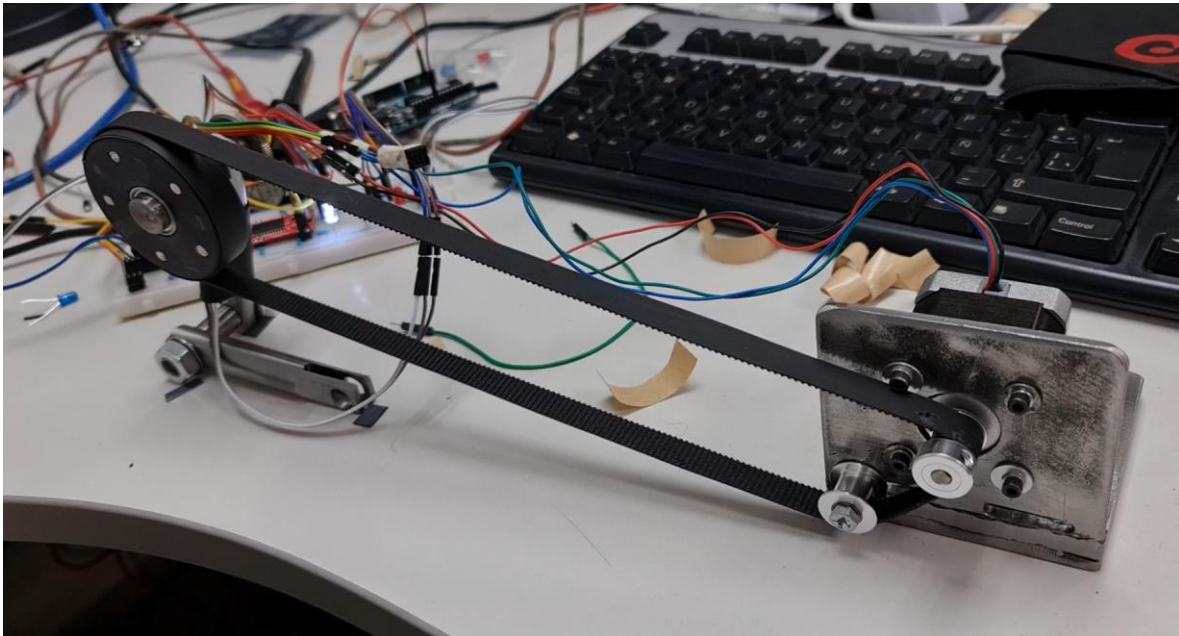
Además, se confeccionó una interfaz de usuario preliminar para poder observar e interpretar la interacción entre las partes.

Se determinó que el formato más adecuado para el envío de datos es hexadecimal pues se evitan confusiones con los signos. Las pruebas iniciales evidenciaron problemas de interpretación por parte de Python pues a partir de cierta magnitud, las cadenas de bits se interpretaban como números negativos cuando son enviados como decimales, sin embargo, al utilizar números hexadecimales el problema se elimina.

Tanto la componente eléctrica como computacional funcionaron bien a lo largo del diseño como de la construcción del prototipo como se observa en la Figura 122., en la Figura 123., y en la Figura 124.

En la Figura 121., se puede observar la elaboración de un prototipo para pruebas que permitía mantener una velocidad constante a partir de un motor a pasos.

En la Figura 122., se puede observar que la totalidad del tren de pulsos se envía en un máximo de 5,6 microsegundos; si se interpreta este valor como la mayor cantidad de tiempo que puede haber entre los pulsos del sensor para que el microprocesador comunique los datos fielmente a la Raspberry Pi 4 se obtiene que la frecuencia de pulsos máxima 178,57 Hz; esta cifra corresponde a una velocidad máxima permitida por la comunicación serial de hasta aproximadamente 417,38 m/min la cual está casi 3 veces por encima de la velocidad máxima de producción de la máquina.



*Figura 121. Prototipo de prueba para mantener velocidad constante. Fuente: elaboración propia.*



Figura 122. Visualización del tren de pulsos enviados por el ATmega 328p hasta la Raspberry Pi 4. Fuente: elaboración propia.

```

57     try:
58         line = int(line,16)
59         ancho = (line/pasos)*circunferencia
60         ancho = round(ancho,2)
61         ancho_ini_text.value = ancho
62         a_ini_error.value = ancho - int(a_ini_set.value)
63         print('El ancho medido es ', ancho, ' cm')
64     except:
65         **

```

---

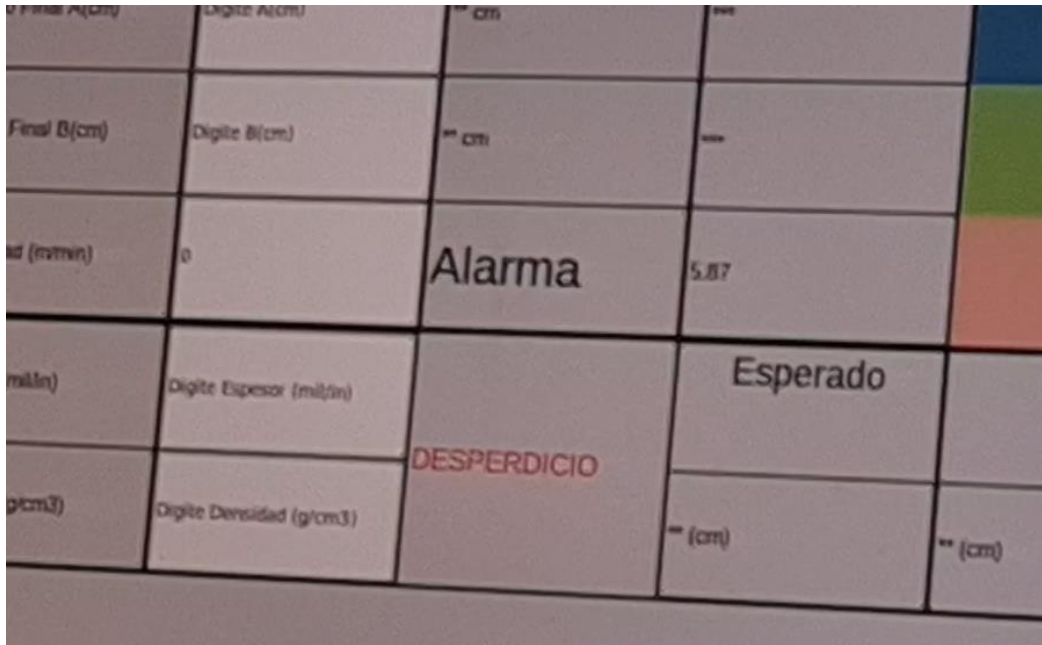
```

Shell X
El ancho medido es 21.68 cm
La velocidad es de 1.674825257670897 m/min.
El promedio de los últimos 4 conteos es de 3.78 m/min.

El ancho medido es 18.76 cm
La velocidad es de 1.8365247213489584 m/min.
El promedio de los últimos 3 conteos es de 3.29 m/min.

```

Figura 123. Interpretación de los datos recibidos por comunicación Serial. Fuente: elaboración propia.

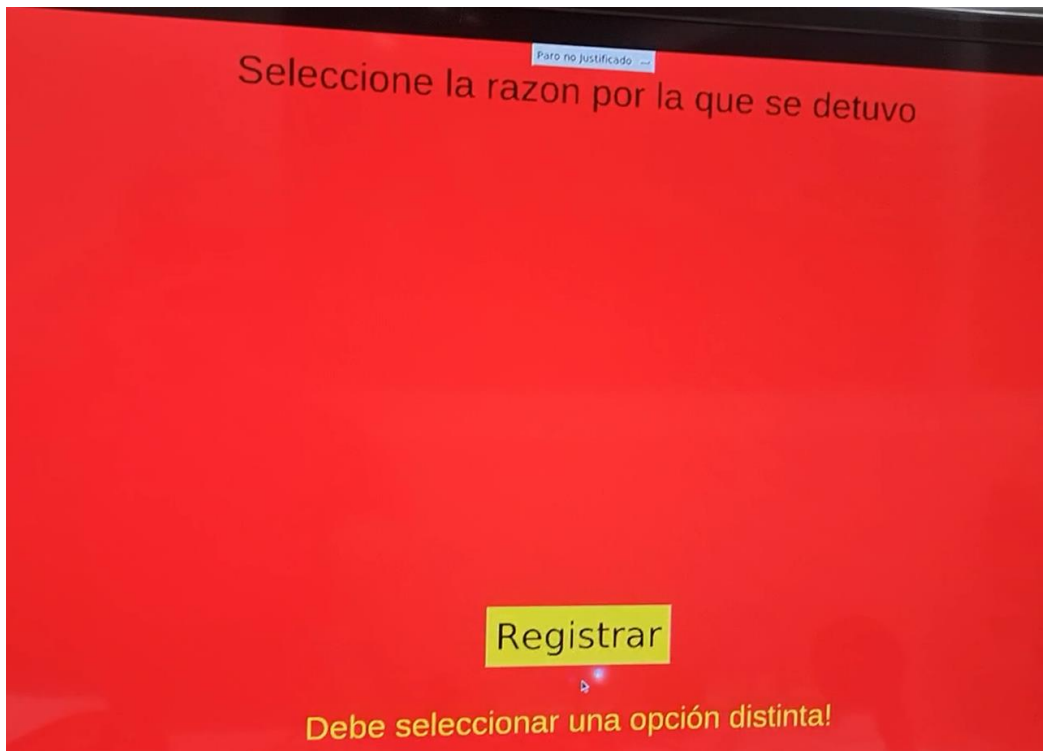


*Figura 124. El momento en el que aparece la alerta en el prototipo de la interfaz.  
Fuente: elaboración propia.*

La palabra *Alarma* que se observa en la Figura 124., corresponde al momento exacto en el que la velocidad de registró como 1,87 m/min.

Este evento se relaciona directamente con el umbral de detección aceptado por producción: se había determinado que, si el tiempo entre dos pulsos del sensor es de alrededor de 4 segundos, se puede interpretar como una detención práctica de la máquina; el microprocesador envía la letra *g* que es interpretada por el algoritmo como una alerta. Ver Sección de Diseño del Circuito Electrónico para el Sistema de Adquisición de Datos de Velocidad.

En la Figura 125., se puede observar la pantalla que se muestra al operador cuando la rueda está prácticamente detenida. Esta pantalla se muestra fielmente cada vez que la rueda se detiene.



*Figura 125. Pantalla de alerta de parada. Fuente: elaboración propia.*

Con la aparición de la pantalla de alerta en la Figura 125., queda validado el sistema de alerta visual.



## Análisis Estadístico del Muestreo de Datos del Sistema Adquisición de Datos de Velocidad

El muestreo realizado se hizo con ayuda de un tacómetro digital de mano como el de la Figura 126.



Figura 126. Tacómetro digital de mano utilizado. Fuente: elaboración propia.

Durante las pruebas, se notó que la velocidad era considerablemente variable además que, al ser un dispositivo de mano, el ángulo de contacto puede influir en la medición. Debido a lo anteriormente descrito y a que el valor del diámetro de la rueda se puede ajustar de manera muy sencilla para acercarse más a la medida real, un análisis de la exactitud de la medida sería engañoso. Lo más conveniente es realizar un análisis sobre la desviación estándar de las mediciones.

Tabla 8. Muestreo para el sistema de medición de velocidad. Fuente: elaboración propia.

Tacómetro	Atípico	Interfaz	Atípico		Q1	Q3
19,82	FALSO	19,83	FALSO	Tacómetro	19,78	19,94
19,88	FALSO	19,65	FALSO	Interfaz	19,62	19,81
19,95	FALSO	19,7	FALSO	RQ	Límite Superior	Límite Inferior
19,91	FALSO	19,84	FALSO	0,17	20,19	19,53
19,77	FALSO	19,53	FALSO	0,19	20,1025	19,3225
20,1	FALSO	19,76	FALSO			
19,52	VERDADERO	19,78	FALSO			
20,09	FALSO	19,67	FALSO			
19,78	FALSO	19,66	FALSO			
19,85	FALSO	19,83	FALSO			
19,76	FALSO	19,49	FALSO			
20,04	FALSO	19,59	FALSO			
19,87	FALSO	19,79	FALSO			

19,76	FALSO	19,54	FALSO
19,94	FALSO	19,61	FALSO
19,74	FALSO	19,78	FALSO
19,99	FALSO	19,54	FALSO
19,76	FALSO	19,64	FALSO
19,81	FALSO	19,88	FALSO
19,98	FALSO	19,64	FALSO
19,68	FALSO	19,65	FALSO
19,88	FALSO	19,85	FALSO
19,78	FALSO	19,62	FALSO
19,96	FALSO	19,63	FALSO
19,82	FALSO	19,94	FALSO
19,91	FALSO	19,63	FALSO
19,79	FALSO	19,49	FALSO
19,86	FALSO	19,92	FALSO
19,96	FALSO	19,69	FALSO
19,91	FALSO	19,43	FALSO
19,85	FALSO	19,96	FALSO
19,5	VERDADERO		

Los datos medidos en metros por minuto se pueden observar en la Tabla 8.

Las medidas desde la interfaz se tomaron directamente del monitor conectado a la Raspberry Pi 4, las sesiones de muestreo se realizaron en momentos distintos dentro de la misma rutina de producción unos minutos aparte la una de la otra. 32 datos se obtuvieron en un espacio de 31 segundos con el tacómetro y 31 datos se obtuvieron en 19 segundos con el tacómetro.

Al realizar el análisis estadístico sobre los datos típicos en dichas listas, se obtienen los siguientes resultados:

- Las desviaciones estándar son 0,105 m/min para el tacómetro y 0,141 m/min para la interfaz.
- El promedio de las muestras es de 19,873 m/min para el tacómetro y 19,695 m/min para la interfaz.
- El coeficiente de variación es de 0,0052 para los datos obtenidos con el tacómetro y 0,0072 para los datos obtenidos desde la interfaz.

Es importante señalar que, entre las mediciones realizadas por la interfaz, no se encontraron datos atípicos, en cambio, con el tacómetro se recibieron 2 mediciones atípicas.

En estas velocidades, las frecuencias de muestreo son 1,03 datos por segundo para el tacómetro y 1,63 datos por segundo para la interfaz. Es importante notar que cada uno de los datos presentados en la interfaz es un promedio de las 5 últimas

muestras por lo que la frecuencia real de muestreo a una velocidad promedio de 19,695 m/min es aproximadamente 8,16 datos por segundo.

### **Análisis Mecánico del Sistema de Adquisición de Datos de Velocidad**

El análisis del ensamblaje del sistema de adquisición de velocidad se realizó en el apartado de la propuesta del diseño, los resultados obtenidos son los siguientes:

El factor de seguridad de operación de los cojinetes a un máximo de 800 rpm es igual a 341 y la vida útil esperada es de 59703 años.

El factor de seguridad para el brazo que lleva el mayor esfuerzo es igual a 1623, ver Figura 107.

El factor de seguridad ante el esfuerzo cortante del pasador 3 es igual a 59228, ver Figura 109.

### **Sistema de Adquisición de Datos del Ancho**

En el caso del sensor de ancho, las variables más importantes por validar son:

- La detección de los límites del producto.
- El conteo correcto de los pasos del motor.
- La comunicación entre el Atmega 328 y la Raspberry Pi 4: las funciones programadas para la comunicación y la configuración electrónica de ambos circuitos es exactamente la misma, así como el algoritmo que interpreta la información recibida por la Raspberry Pi por lo que esta etapa queda validada por las mismas pruebas realizadas en el sistema de adquisición de velocidad.
- Un margen de error menor a los 2 mm: para que pueda detectar fielmente hasta 5 mm de déficit o superávit del producto.

### **La Detección de los Límites del Producto**

Esta etapa de validación no corresponde únicamente al inicio y al final del ancho de la filmina plástica, sino que también comprende la reacción del sistema ante posibles arrugas u ondulaciones dentro del producto.

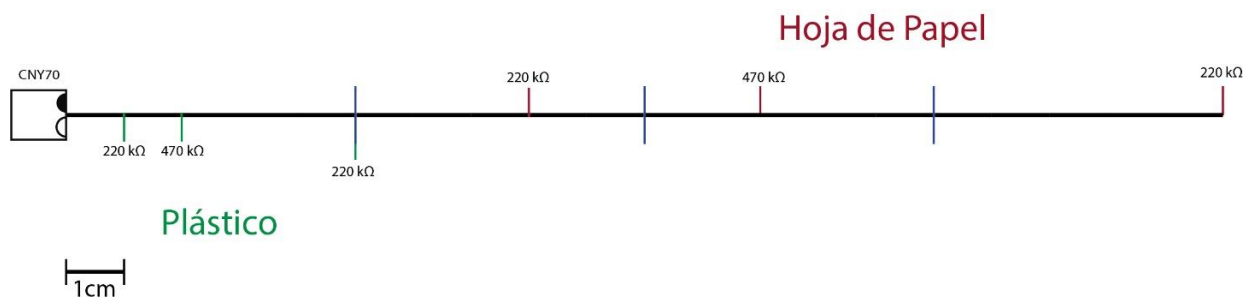
Se aclara que cualquier tipo de discontinuidad en el producto (agujeros, cortes, etc.) es un defecto por lo que las pruebas no contemplan este escenario. Si la discontinuidad es suficientemente pequeña, el sistema podría no mostrarse reactivo ante ella y si es suficientemente grande, podría interpretarla como un límite.

Para la detección de la presencia del producto y la modulación de los valores de resistencia conectadas a las patillas del ánodo y del emisor del CNY70, se realizaron 4 tipos de pruebas:

1. La primera corresponde a la detección de objetos aleatorios frente al sensor CNY70 a diferentes distancias, los objetos utilizados fueron: hojas blancas y plástico translucido.
2. La segunda prueba corresponde a la detección de los límites de una caja de cartón forrada con uno de los productos plásticos extruidos por la máquina E16.
3. La tercera prueba consistió en colocar distintas filminas de productos tensadas perpendicularmente frente al ensamblaje para determinar cuáles eran los valores más adecuados de resistencias por utilizar de acuerdo con el material.
4. La cuarta prueba se realizó sobre la máquina en funcionamiento, tomando en cuenta las horas de mayor intensidad solar y las de menor intensidad solar (noche).

### Primera Prueba

La primera prueba consistió simplemente en el acercamiento de diversos objetos al sensor CNY70 a una distancia variable, una vez el sensor detectara la presencia del objeto y el microprocesador enviara un dato por USART, se realizaba una marca en una hoja fija en el suelo, se medía la distancia y se anotaba el valor de las variables eléctricas utilizadas; para cada objeto se variaban los valores eléctricos del circuito y se observaba la reacción del sistema. La prueba se realizó con una resistencia de ánodo constante a 220 Ohms y con valores variables de resistencia de emisor.



*Figura 127. Representación de la hoja utilizada para la primera prueba. Fuente: elaboración propia.*

En la Figura 127., se pueden observar una serie de líneas que marcan el lugar donde el sensor detecta el ruido y debajo de ese punto se anota el valor de la resistencia de emisor.

Los resultados de la primera prueba son los siguientes, ver Figura 127.:

- Para plástico translúcido:
  - Cuando la resistencia de emisor es igual a 220 kOhms, la distancia máxima de detección es de 2 cm.
  - Cuando la resistencia de emisor es igual a 470 kOhms, la distancia máxima de detección es de 3 cm.
  - Cuando la resistencia de emisor es igual a 1Mohm, la distancia máxima de detección es 5 cm.
  
- Para hoja blanca de papel Bond:
  - Cuando la resistencia de emisor es igual a 220 kOhms, la distancia máxima de detección es de 8 cm.
  - Cuando la resistencia de emisor es igual a 470 kOhms, la distancia máxima de detección es de 12 cm.
  - Cuando la resistencia de emisor es igual a 1 Mohm, la distancia máxima de detección es de 20 cm o más.

La prueba dejó en evidencia que la variabilidad de la resistencia de emisor en el sensor permite aumentar y disminuir la distancia a la que se detecta el material y que la sensibilidad ante el ruido es inversa a la magnitud de la resistencia de emisor, y que la mayor distancia posible para el plástico utilizado es de 5 cm.

## **Segunda Prueba**

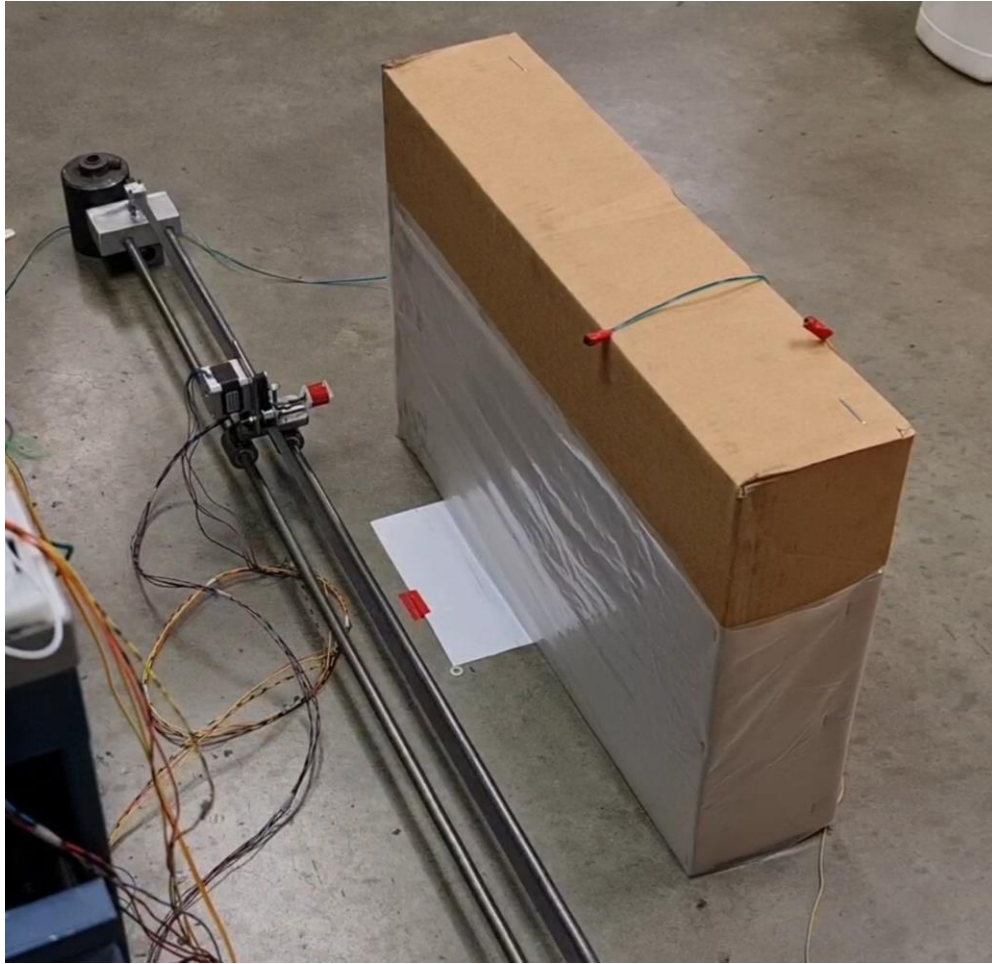
En este momento, ya se contaba con un prototipo avanzado del ensamblaje, sin embargo, el material utilizado para las barras era aluminio 6061 y el rodamiento utilizado era uno lineal Thomson A81420, ver Figura 127.; este cojinete se escogió de manera arbitraria pues era el único disponible en inventario para las dimensiones de las barras en existencia, sin embargo, en pruebas preliminares del movimiento del carrito, se encontró que el aluminio era rayado fácilmente por las bolas de acero del cojinete, debido a este problema, se decidió cambiar el material de las barras por acero 4140. Los rodamientos resultaron también ser problemáticos pues las bolas de acero se desprendían y cualquier escombros que se introdujera en el cojinete provocaba mucha fricción innecesaria; a partir de estas pruebas se decidió reemplazar el cojinete lineal por un cojinete liso de bronce.



*Figura 128. Rodamiento utilizado previo a la segunda prueba. Fuente: adaptado de [70].*

Se realizaron repetidas pruebas de movimiento del carrito que fueron satisfactorias, sin embargo, pusieron en evidencia la necesidad de utilizar una buena lubricación de las barras.

Al haberse familiarizado con el comportamiento del sensor CNY70 y al haber realizado la modulación de los valores de resistencia, se procede a realizar una prueba con el plástico translucido sujeto por una caja de cartón para poder mantener una dimensión constante del material frente al sensor y, además, probar que el algoritmo detecte los bordes y cambie la dirección del movimiento del motor para realizar el recorrido nuevamente. Ver Figura 129.



*Figura 129. Configuración para la segunda prueba. Fuente: elaboración propia.*

La caja tenía un ancho de 63 cm.

En las pruebas se realizó cambiando la distancia con un valor de resistencia igual a 220 Kohm en el emisor y 220 Ohms en el ánodo. Estos muestreos se realizaron al menos 10 veces cada uno:

- A 1cm del sensor: 70 cm en promedio.
- A 3 cm del sensor: 66 cm en promedio.
- A 5 cm del sensor: 63,2 cm en promedio.

Los datos se podían observar en la interfaz de usuario. Ver Figura 130.

EXTRUSORA E16			
	VARIABLES	SETEADO	REAL
CONTROLADAS	ANCHO INICIAL (cm)	63	63.27
	ANCHO FINAL A (cm)	94	**
	ANCHO FINAL B (cm)	94	**
	VELOCIDAD (m/min)	79.99	**

Figura 130. Muestras visualizados en la interfaz. Fuente: elaboración propia.

Los resultados preliminares fueron alarmantes, sin embargo, al analizar detenidamente la configuración de la prueba, se pudo concluir que el superávit de la medida se daba del lado donde el plástico rodeaba la zona lateral de la caja y quedaba frente a la ventana, por lo que la distancia adicional medida era provocada por el ruido introducido por la geometría de la muestra y por la luz solar.

Al aumentar la distancia se disminuía la proyección del área lateral de la caja sobre el sensor por lo que se conseguía una medida bastante exacta. Las dos décimas adicionales se podrían deber a que el plástico queda un tanto levantado justo al rodear el área lateral de la caja o por una falta de paralelismo entre la superficie del plástico y el recorrido del sensor CNY70.

### Tercera Prueba

La tercera prueba se realizó tensando filminas de distintos productos extruidos por la máquina E16 frente al sensor CNY70 y permitiendo que el dispositivo de medición completara su recorrido completo de lado a lado.

Ocasionalmente, el soporte donde se encontraba la filmina tensada se acercaba y alejaba para observar el comportamiento de la señal del CNY70. Sin embargo, se decidió mantener la distancia constante a aproximadamente 2 mm de distancia del sensor para todas las pruebas pues la posición en la máquina será fija.

Con ese breve experimento se encontró que disminuyendo la resistencia del ánodo se aumenta la intensidad del LED del CNY70 y esto influye directamente en la distancia a la que el sensor puede detectar el plástico de acuerdo con su color: el plástico translúcido y el plástico negro requieren mayor intensidad lumínica para la detección proporcional a la distancia respecto a sus contrapartes.





*Figura 131. Tercera prueba con producto negro. Fuente: elaboración propia.*

Esta prueba permitía modular la resistencia del ánodo del CNY70 para los distintos productos, pues en las pruebas iniciales solo se había utilizado el plástico translúcido.

Durante esta prueba se implementó también la utilización de un capacitor en paralelo a la resistencia del emisor para observar su comportamiento ante discontinuidades del plástico.

Según este procedimiento, los valores más adecuados para la resistencia del ánodo y un capacitor en paralelo a la resistencia del emisor con resistencia de emisor igual a 220 kOhms son:

- 200 Ohms para plástico translúcido con 0,1  $\mu$ F de capacitor en paralelo con la resistencia del emisor.
- 200 Ohms para plástico translúcido blanco con 0,1  $\mu$ F de capacitor en paralelo con la resistencia del emisor.
- 200 Ohms para plástico blanco sin capacitor en paralelo con la resistencia del emisor.
- 200 Ohms para plástico negro sin capacitor en paralelo con la resistencia del emisor.

A pesar de que el ruido de la señal del sensor CNY70 era disminuido por el capacitor, el tiempo de caída de la señal se aumentaba y esto tenía un impacto

directo sobre la medida, por lo que la hacía menos exacta. Ver Figura 132., y Figura 133.

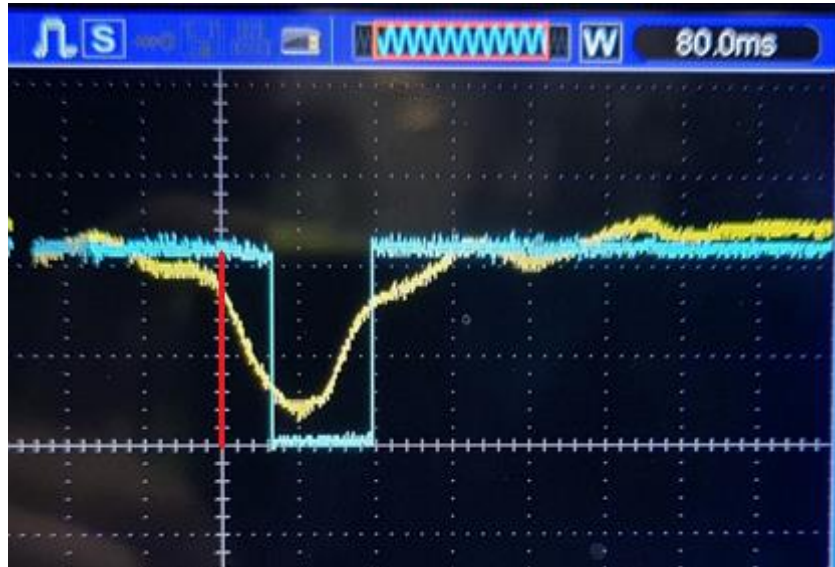


Figura 132. Tiempo de caída de la señal sin capacitor. Fuente: elaboración propia.

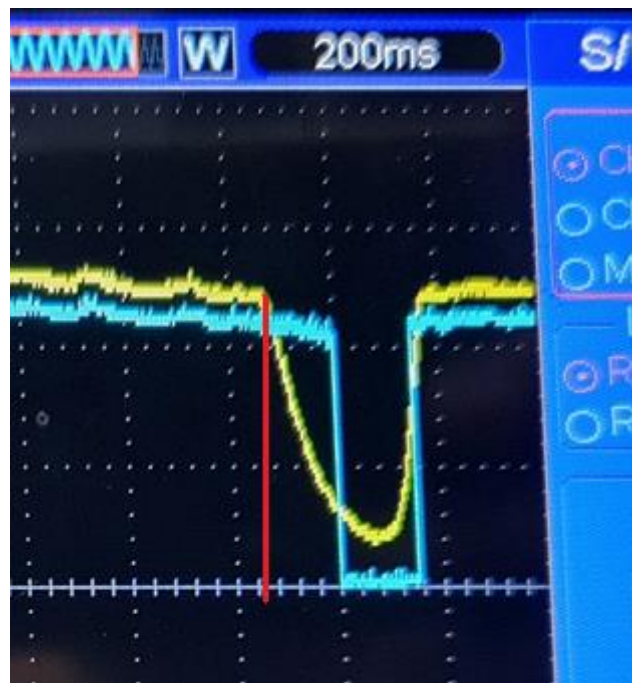


Figura 133. Tiempo de caída de la señal con capacitor. Fuente: elaboración propia.

La señal representada por la curva amarilla corresponde a la que sale directamente del sensor CNY70, esta señal es medida en paralelo a la resistencia del emisor, la señal celeste corresponde a la salida de la compuerta NAND 4093 con disparador Schmitt; el tiempo desde la caída inicial de la señal del CNY70 y el cruce con el

flanco decreciente de la compuerta 4093 corresponde a un error propio del sistema debido al umbral de la compuerta. La línea roja ha sido introducida manualmente para visualizar el momento en que inicia la caída, este punto de la curva representa el límite real de la filmina.

Se puede entonces observar, que, sin la utilización de un capacitor, el tiempo de caída es aproximadamente 48 ms según la escala del osciloscopio y con capacitor es 160 ms según la escala del osciloscopio.

Como consecuencia de este resultado, se decidió no utilizar capacitor en paralelo a la resistencia del emisor a pesar de que disminuía la cantidad de ruido pues introduce un error no deseado y, además, el umbral de la compuerta 4093 ya representa una especie de filtro.

Simultáneamente, en repetidas ocasiones a lo largo de esta etapa, se puso a prueba la tolerancia ante las ondulaciones presentes en el plástico durante la operación de la máquina; este ensayo se realizó golpeando de manera frecuente el plástico para provocar el acercamiento y el alejamiento de la filmina, la magnitud de estas ondulaciones fue de entre 1 y 2 cm; a pesar de la ocurrencia de dichas perturbaciones, el dispositivo no se mostró reactivo ante ellas y continuó su curso hasta llegar al límite de la filmina.

Durante esta prueba, también se decidió realizar un análisis estadístico sobre un grupo de 39 valores donde:

- Su medida promedio fue de 632,266 mm.
- Su desviación estándar fue de 0,64 mm.

De esta manera queda validada la precisión del sistema de adquisición de ancho.

*Tabla 9. Muestreo realizado para la tercera prueba. Fuente: elaboración propia.*

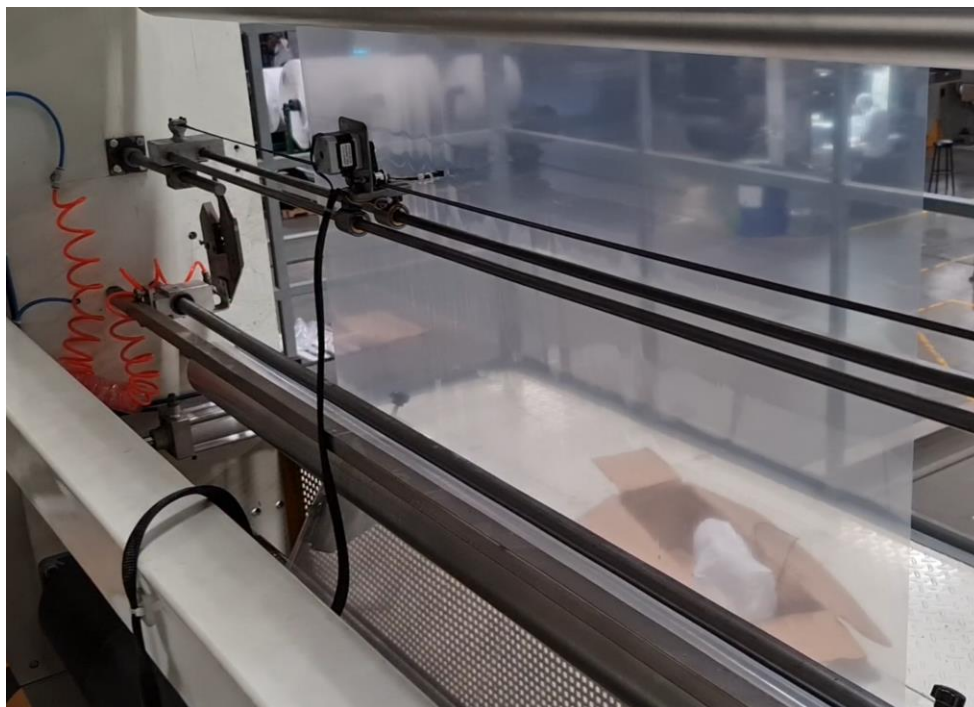
Muestra	Unidades	Decimales	Medida				cm	mm
1	63	324	63,324			Varianza	0,00411611	0,41161134
2	63	243	63,243			Desviacion Estandar	0,06415694	0,64156943
3	63	140	63,14			Promedio	63,2266923	632,266923
4	63	160	63,16			Coef. Variacion	0,00101471	0,00101471
5	63	263	63,263					
6	63	243	63,243					
7	63	201	63,201					
8	63	201	63,201					
9	63	201	63,201					
10	63	263	63,263					
11	63	284	63,284					
12	63	243	63,243					

13	63	345	63,345					
14	63	325	63,325					
15	63	181	63,181					
16	63	222	63,222					
17	63	284	63,284					
18	63	222	63,222					
19	63	284	63,284					
20	63	263	63,263					
21	63	201	63,201					
23	63	160	63,16					
24	63	284	63,284					
25	63	181	63,181					
26	63	284	63,284					
27	63	181	63,181					
28	63	324	63,324					
29	63	201	63,201					
30	63	99	63,099					
31	63	140	63,14					
32	63	222	63,222					
33	63	140	63,14					
34	63	304	63,304					
35	63	222	63,222					
36	63	222	63,222					
37	63	201	63,201					
38	63	140	63,14					
39	63	324	63,324					
40	63	119	63,119					

### **Cuarta Prueba**

La cuarta prueba se realizó con el dispositivo implementado en la máquina, en ella se ensayaron las distintas condiciones de luminosidad en las que se lleva a cabo la operación de la extrusora.

El objetivo es tanto modular nuevamente los valores de las resistencias conectadas al ánodo y al emisor del CNY70, y verificar que el dispositivo sea capaz de detectar fielmente los límites del producto plástico y su comportamiento ante las ondulaciones presentes en la filmina.



*Figura 134. Prototipo instalado en la máquina. Fuente: elaboración propia.*

Se hicieron pruebas continuas de funcionamiento durante 2 horas cada una; una de ellas desde las 11 am hasta la 1 pm en día soleado para así comprobar el funcionamiento cuando la intensidad solar es máxima, y la otra durante 2 horas desde las 5:40 pm hasta las 7:40 pm para comprobar el funcionamiento del dispositivo cuando la intensidad solar es nula.

Mientras se realizaba la primera prueba, los operadores de la máquina E16 hicieron cambio de resina por lo que el color del plástico cambio de blanco translúcido a translucido. Esta transición permitió observar el comportamiento de la señal con ambos productos.

Los resultados de la primera ronda son:

- 16 kOhms al emisor a las 10:30 am aseguran un éxito de detección cercano al 100% en todos los recorridos sobre resina blanca.
- 4 kOhms al emisor a las 12:01 pm aseguran un éxito de detección cercano al 100% en todos los recorridos sobre resina blanca.
- 4 kOhms al emisor a las 12:30 pm y 135 ohms al cátodo aseguran un éxito de detección cercano al 100% en todos los recorridos sobre resina transparente.

La prueba de día limita la magnitud de la resistencia del emisor a un máximo de 4 kOhms, un valor superior a ese provoca que la magnitud base del voltaje de la señal sea mayor al límite inferior del disparador Schmitt y que no se detecte el final de la filmina.

Los resultados de la segunda ronda son:

- Desde las 5:45 pm hasta las 7:01 pm, la resistencia del ánodo del CNY70 se mantuvo a un valor de 62 Ohms con una tasa de éxito de 150 carreras sobre 152 carreras, equivalente a un 98,7% de éxito.

Las carreras fallidas se debieron a problemas de deslizamiento por falta de lubricación, si no hubiera sido por eso, la tasa de éxito habría sido del 100 %

Los resultados obtenidos refuerzan lo observado en las pruebas anteriores referente al comportamiento del dispositivo según los valores de las resistencias utilizadas: la resistencia del emisor influye directamente en la sensibilidad al ruido y la distancia de detección; y la resistencia del ánodo influye inversamente en la intensidad de luz entregada por el LED del CNY70.

Durante esta prueba también se cronometró la cantidad de tiempo que toma el dispositivo en recorrer una filmina de 123 cm de lado a lado y se encontró que el promedio era de 25,16s; esto significa que el carrito se mueve a una rapidez promedio de 4,88 cm/s.

Con las cuatro pruebas realizadas, se puede validar completamente el funcionamiento del sistema de medición de ancho diseñado.

### **Análisis Mecánico del Sistema de Adquisición de Ancho.**

En la sección de la propuesta de diseño se realizó un análisis de las condiciones de operación del sistema de adquisición de datos del ancho; los resultados fueron los siguientes:

El factor de seguridad ante esfuerzo normal de las barras de acero 4140 de  $\frac{1}{2}$  pulgada es de 15,17 y de 5074,4 ante esfuerzo cortante. Ver Figura 111., y Figura 112.

El factor de seguridad ante esfuerzo normal de las barras de acero 4140 de 1 pulgada es igual a 42,9 y de 2574 ante el esfuerzo cortante.

## Interfaz de Usuario y Comunicación de Datos

Gracias a la flexibilidad que ofrece Python y las posibilidades disponibles para manejar los datos en línea, las pruebas serán discretas en tanto a los requerimientos.

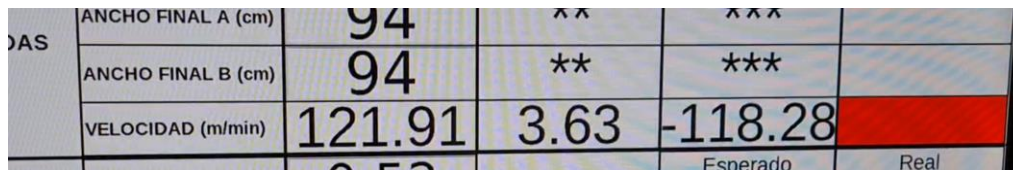
Se validarán los siguientes eventos:

- Cálculo correcto de los valores medidos por los sistemas de medición.
- Activación de la alerta visual de manera adecuada.
- Comunicación de los datos en línea en la frecuencia deseada.

### Cálculo Correcto de los Valores Medidos por los Sistemas de Medición

En el caso del sistema de adquisición de velocidad fue necesario recalculer el valor del radio de la rueda debido a la adición de la cinta de caucho alrededor de esta. Eso agregó un pequeño error en la medición que fue fácilmente corregido al ajustar el valor en el algoritmo.

La prueba de visualización se llevó a cabo con el ensamblaje observado en la Figura 121., al realizar esta prueba, se puede observar el resultado en pantalla: ver Figura 135.



DAS	ANCHO FINAL A (cm)	94	**	***	
	ANCHO FINAL B (cm)	94	**	***	
	VELOCIDAD (m/min)	121.91	3.63	-118.28	
				Esperado	Real

Figura 135. Prueba de visualización de datos. Fuente: elaboración propia.

Esta prueba se llevó a cabo durante varios minutos y la comunicación se mantuvo constante sin ningún fallo.

Es necesario aclarar que la prueba no tenía el objetivo de registrar la exactitud o la precisión del dispositivo, únicamente la visualización de los datos y la alarma alerta visual como se puede observar.

Posteriormente, con el dispositivo implementado sobre la máquina, se hizo un ensayo breve de exactitud con ayuda de un tacómetro calibrado en manos del supervisor de planta.

Inicialmente el error observado fue de aproximadamente +12,35%, posterior a una cuidadosa revisión, se notó que el valor del radio utilizado en la función era incorrecto (4,1 cm) y procedió a corregirse por el dato correcto (3,1 cm), a partir de ese momento el error cambió a 0,7% y se mantuvo así por el resto de la prueba.

Los datos observados en dicha prueba fueron los siguientes:

- Previo al ajuste:
  - La rapidez medida por el dispositivo fue de 30 m/min en promedio.
  - La rapidez medida por el tacómetro fue de 26,7 m/min en promedio.
- Posterior al ajuste:
  - La rapidez medida por el dispositivo fue de 26,9 m/min en promedio.
  - La rapidez medida por el tacómetro se mantuvo en 26,7 m/min en promedio.

La visualización y el muestreo de datos de la velocidad queda validada a partir de las pruebas realizadas.

En el caso del sistema de adquisición de datos del ancho, la segunda prueba realizada dejó en evidencia la medición correcta de los valores, ver Figura 130., el comportamiento sobre la máquina fue consistente con dicho resultado sin embargo la comparación concreta entre la medida real y la adquirida en la cuarta prueba se dificulta grandemente pues el único instrumento disponible para la comparación es la cinta métrica y la medición manual con dicho instrumento resulta inexacta e imprecisa.

Es importante notar que debido a la duración de la caída del pulso (48 ms), si se toma en cuenta la velocidad del carrito (4,88 cm/s), se puede esperar un error constante de 2,3 mm de superávit por lo que el error en la Figura 130., es apenas de 0,4 mm. Este error calculado permite el ajuste de los cálculos realizados dentro de las funciones encargadas.

También se ensayó el funcionamiento simultaneo de los 3 hilos de programación para el cálculo del ancho en los 3 puntos determinados, los resultados evidenciaron la correcta comunicación entre el microprocesador y la Raspberry Pi 4 para los 3 dispositivos. Ver Figura 136.

EXTRUSORA E16					
	VARIABLES	SETEADO	REAL	DIFERENCIA	ALERTA
CONTROLADAS	ANCHO INICIAL (cm)	100	40.69	-59.31	
	ANCHO FINAL A (cm)	94	40.69	-53.31	
	ANCHO FINAL B (cm)	94	40.69	-53.31	

Figura 136. Prueba con los 3 hilos para los 3 dispositivos de medición de ancho.  
Fuente: elaboración propia.

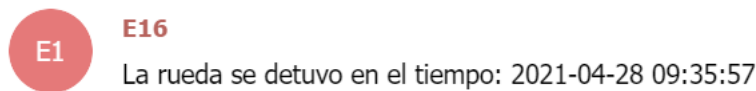


## Activación de la Alerta Visual

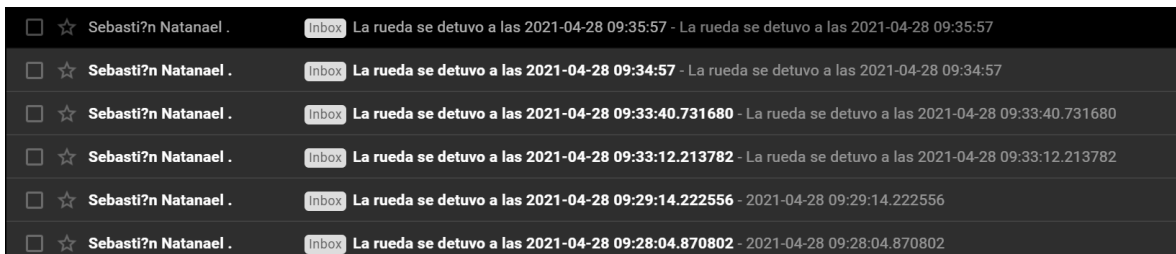
En el transcurso de las pruebas sobre el sistema de adquisición de velocidad, se puso a prueba la activación de la alerta visual en la interfaz, esta alerta visual funcionó de manera correcta, cada vez que el microprocesador comunicó el reporte de detención a la Raspberry Pi 4; además, la interfaz de usuario permite visualizar cuando la rueda se mueve a una velocidad menor que la permitida por el personal de producción. Ver Figura 125., Figura 135., y Figura 136.

Como función adicional, se implementó la notificación de la detención de la rueda a través de Telegram, ver Figura 137., y Outlook, ver Figura 138.

miércoles, 28 de abril de 2021



*Figura 137. Mensaje enviado por el escenario de Integromat a través de la aplicación Telegram. Fuente: elaboración propia.*



*Figura 138. Mensajes enviados a través de correo electrónico por el escenario de Integromat. Fuente: elaboración propia.*

A partir de las pruebas realizadas, el funcionamiento de la alerta visual queda validado.

## Comunicación y Almacenamiento de los Datos en Línea

La comunicación y el envío de los datos en línea se probó en numerosas ocasiones tanto con ejecuciones aleatorias como continuas. La comunicación en línea a través de la implementación de la biblioteca *Requests* [32] y la ejecución de los escenarios de Integromat encargados del almacenamiento de los datos en línea fueron exitosas.

Los datos se enviaron de manera exitosa hasta con algunos pocos segundos de diferencia entre los pedidos, por lo que la frecuencia de envío cada 5 minutos funcionó de manera satisfactoria.

	Fecha y Hora	Velocidad	Velocidad	Diferencia de	Ancho Inid	Ancho Inid	Diferencia	Tolerancia	Ancho
404	26/04/2021 13:35	81,73	0	81,73	100	66,05	-33,95	1	
405	26/04/2021 13:30	81,73	0	81,73	100	66,06	-33,94	1	
406	26/04/2021 13:25	81,73	0	81,73	100	66,12	-33,88	1	
407	26/04/2021 13:20	81,73	0	81,73	100	66,06	-33,94	1	
408	26/04/2021 13:15	81,73	0	81,73	100	66,05	-33,95	1	
409	26/04/2021 13:10	81,73	0	81,73	100	66,03	-33,97	1	
410	26/04/2021 13:05	81,73	0	81,73	100	66,06	-33,94	1	
411	26/04/2021 13:00	81,73	0	81,73	100	66,12	-33,88	1	
412	26/04/2021 12:55	81,73	0	81,73	100	66,07	-33,93	1	
413	26/04/2021 12:50	81,73	0	81,73	100	66,15	-33,85	1	
414	26/04/2021 12:45	81,73	0	81,73	100	66,06	-33,94	1	
415	26/04/2021 12:40	81,73	0	81,73	100	66,1	-33,9	1	
416	26/04/2021 12:35	81,73	0	81,73	100	84,84	-15,16	1	
417	26/04/2021 12:30	81,73	0	81,73	100	94,42	-5,58	1	
418	26/04/2021 12:25	81,73	0	81,73	100	66,15	-33,85	1	
419	26/04/2021 12:20	81,73	0	81,73	100	66,38	-33,62	1	
420	26/04/2021 12:15	81,73	0	81,73	100	66,27	-33,73	1	
421	26/04/2021 12:10	81,73	0	81,73	100	66,2	-33,8	1	
422	26/04/2021 12:05	81,73	0	81,73	100	66,29	-33,71	1	

Figura 139. Datos recibidos por Excel 365, enviados por la Raspberry Pi 4. Fuente: elaboración propia.

En la Figura 139., se muestra la forma en la que se observan los datos enviados de manera automática a un libro de Excel 365 almacenado en la nube.

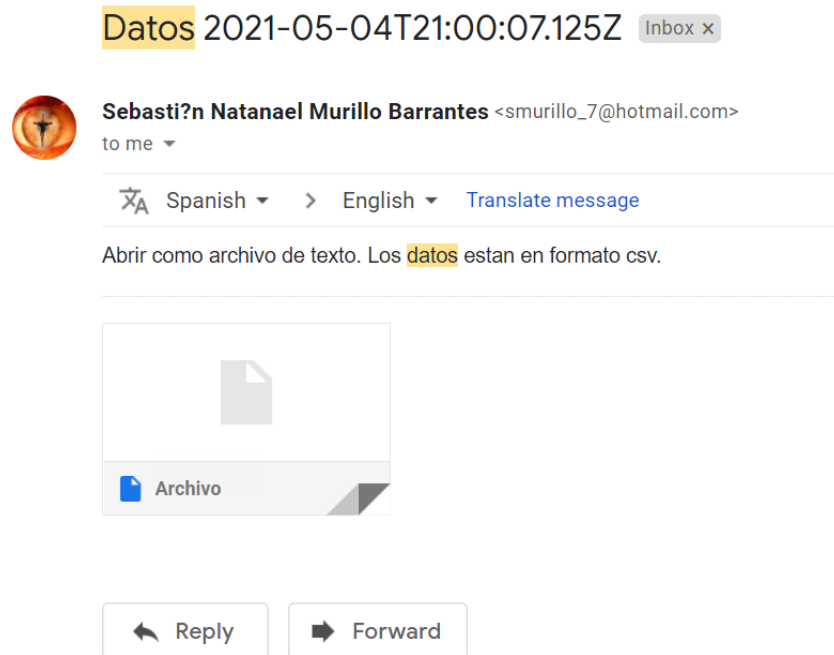
Como prestación adicional, la interfaz crea un registro de datos en formato .CSV el cuál almacena en la memoria de la Raspberry Pi 4 y envía por correo al final del día para así tener un respaldo en bruto de la información. Ver Figura 140.

2021-04-30\_08.txt - Mousepad

File Edit Search View Document Help

2021-04-30 08:41:49,835.47,0,835.47,10.0,9.12,-0.88,1.0,0.0,9.12,9.12,1.0,0.0,9.12,0.5,0.923,0.52,20.0,0.0,110.0,0.0,10.0,\*,\*,\*, DIGITE DAT  
2021-04-30 08:41:54,835.47,0,835.47,10.0,9.14,-0.86,1.0,0.0,9.14,9.14,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.02,110.0,0.0,10.0,\*,\*,\*, DIGITE D/  
2021-04-30 08:41:59,835.47,0,835.47,10.0,9.12,-0.88,1.0,0.0,9.12,9.12,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.04,110.0,0.0,10.0,\*,\*,\*, DIGITE D/  
2021-04-30 08:42:04,835.47,0,835.47,10.0,9.14,-0.86,1.0,0.0,9.14,9.14,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.02,110.0,0.0,10.0,\*,\*,\*, Bolsa ne,  
2021-04-30 08:42:09,835.47,0,835.47,10.0,9.18,-0.82,1.0,0.0,9.18,9.18,1.0,0.0,9.16,0.5,0.923,0.52,20.0,0.02,110.0,0.0,10.0,\*,\*,\*, Bolsa Negr  
2021-04-30 08:42:14,835.47,0,835.47,10.0,9.14,-0.86,1.0,0.0,9.14,9.14,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.02,110.0,0.0,10.0,\*,\*,\*, Bolsa Neg  
2021-04-30 08:42:19,835.47,0,835.47,10.0,9.14,-0.86,1.0,0.0,9.14,9.14,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.02,110.0,0.0,10.0,\*,\*,\*, Sebastian,\*, Br  
2021-04-30 08:42:24,835.47,0,835.47,10.0,9.13,-0.87,1.0,0.0,9.13,9.13,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.03,110.0,0.0,10.0,\*,\*,\*, Sebastian, 2011f  
2021-04-30 08:42:30,835.47,0,835.47,10.0,9.15,-0.85,1.0,0.0,9.15,9.15,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.01,110.0,0.0,10.0,\*,\*,\*, Sebastian, 2011f  
2021-04-30 08:42:35,835.47,0,835.47,10.0,9.16,-0.84,1.0,0.0,9.16,9.16,1.0,0.0,9.16,0.5,0.923,0.52,20.0,0.0,110.0,0.0,10.0,\*,\*,\*, Sebastian, 201156f  
2021-04-30 08:42:40,835.47,0,835.47,10.0,9.08,-0.92,1.0,0.0,9.08,9.08,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.08,110.0,0.0,10.0,\*,\*,\*, Sebastian, 2011f  
2021-04-30 08:42:45,835.47,0,835.47,10.0,9.1,-0.9,1.0,0.0,9.1,9.1,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.06,110.0,0.0,10.0,\*,\*,\*, Sebastian, 201156897  
2021-04-30 08:42:50,835.47,0,835.47,10.0,9.11,-0.89,1.0,0.0,9.11,9.11,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.05,110.0,0.0,10.0,\*,\*,\*, Sebastian, 2011f  
2021-04-30 08:42:55,835.47,0,835.47,10.0,9.11,-0.89,1.0,0.0,9.11,9.11,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.05,1.0,0.0,10.0,\*,\*,\*, Sebastian, 201156f  
2021-04-30 08:43:00,835.47,0,835.47,10.0,9.08,-0.92,1.0,0.0,9.08,9.08,1.0,0.0,9.16,0.5,0.923,0.52,20.0,-0.08,1.0,0.0,10.0,\*,\*,\*, Sebastian, 201156f  
2021-04-30 08:43:05,68.36,0,68.36,10.0,9.02,-0.98,1.0,5.0,9.02,4.02,1.0,0.0,9.16,0.5,0.923,0.52,15.0,-0.06,1.0,0.0,10.0,\*,\*,\*, Sebastian, 201156897  
2021-04-30 08:43:10,9.77,0,9.77,10.0,9.07,-0.93,1.0,7.0,9.07,2.07,1.0,0.0,9.16,0.5,0.923,0.52,12.0,-0.09,1.0,0.0,10.0,\*,\*,\*, Sebastian, 201156897, f  
2021-04-30 08:43:15,7.6,0,7.6,10.0,9.1,-0.9,1.0,9.0,9.1,0.1,1.0,0.0,9.16,0.5,0.923,0.52,11.0,-0.06,1.0,0.0,10.0,\*,\*,\*, Sebastian, 201156897, Bolsa M  
2021-04-30 08:43:20,7.6,0,7.6,10.0,9.09,-0.91,1.0,9.0,9.09,0.09,1.0,0.0,9.16,0.5,0.923,0.52,11.0,-0.07,1.0,0.0,10.0,\*,\*,\*, Sebastian, 201156897, Bol  
2021-04-30 08:43:25,7.6,0,7.6,10.0,9.1,-0.9,1.0,9.0,9.1,0.1,1.0,0.0,9.16,0.5,0.923,0.52,11.0,-0.06,1.0,0.0,10.0,\*,\*,\*, Sebastian, 201156897, Bolsa M

Figura 140. Registro en formato .CSV de los datos adquiridos. Fuente: elaboración propia.



*Figura 141. Registro en formato .CSV enviado por correo electrónico. Fuente: elaboración propia.*

De esta manera quedan validados el almacenamiento de datos en la nube, la comunicación en línea y la alerta visual.

## **Análisis Económico**

Como último objetivo por cumplir, se requiere comprobar que el diseño de dicho sistema sea viable económicamente en cuestión de ahorro y retorno de la inversión, poco útil resultaría un sistema de adquisición de datos que represente más costo del que puede ser recuperado durante la vida útil del sistema.

Para justificar este diseño en el ámbito económico se contrastará el ahorro generado, los costos totales y el retorno de inversión.

### **Del Costo Material**

- Total, en colones, de componentes electrónicos por cada prototipo del sistema de adquisición de ancho: ₡55 529,82.
- Total, en colones, por los tres prototipos relativo a componentes electrónicos para los prototipos del sistema de adquisición de datos de ancho: ₡166 586,46.
- Total, en colones, de componentes estandarizados por cada prototipo del sistema de adquisición del ancho: ₡1 315,96.
- Total, en colones, por los tres prototipos relativo a componentes estandarizados para los prototipos del sistema de adquisición de datos de ancho de ancho: ₡3 947,88.
- Total, en colones, de componentes electrónicos por el prototipo del sistema de adquisición de velocidad: ₡10 094,38.
- Total, en colones, de componentes estandarizados para el prototipo del sistema de adquisición de velocidad: ₡392,25.
- Total, en colones, relativo a la etapa de computación de datos: ₡142 468,92.
- Total, en colones, por los materiales de construcción para todos los prototipos: ₡73 320,22.

### **Del Costo Humano**

- Dos meses en modalidad a medio tiempo del practicante: ₡300 000 x 2 = ₡600 000.
- Cuatro meses en modalidad a tiempo completo del practicante: ₡450 000 x 4 = ₡1 800 000.
- Cinco días de tres técnicos en metalmecánica según el salario mínimo (incluyendo cargas sociales) por día: 3 x 5 x ₡14 601,01 = ₡219 015,15

El costo total del proyecto en colones es de ₡3 072 671,04.

## Cálculo del Beneficio

Para la máquina E16 se estima que se realizan rondas de monitorización cada hora, esta ronda la realiza el operador asignado a la máquina y se estima que la duración al tomar las 4 medidas es de 5 minutos.

Según [71] el salario mínimo para un operario de maquinaria pesada es igual a ₡11 761,76 por día más el 5,25% [74] que corresponde a un total de ₡12 379,25. Esto corresponde a ₡1 547,41 por hora y, por consiguiente, el costo neto de la ronda es:

$$\frac{₡1\ 547,41 \times 5}{60} = ₡128,95$$

Al ser 24 rondas al día, el costo total de estas rondas es ₡3 094,8 al día. La producción de plástico se realiza 6 días a la semana por lo que el costo anual de estas rondas es igual a:

$$6 \times 52 \times ₡3\ 094,8 = ₡965\ 577,6$$

Los costos de operación del sistema completo de medición se pueden estimar de la siguiente manera:

- 15,3 Watts de potencia para la Raspberry Pi 4 [59].
- 9,9 Watts de potencia para los tres motores [19].
- ₡18 154 por mes para la plataforma Integromat [29] [73].

La potencia eléctrica total por día consumida es igual a 604,8 Wh a una tarifa de ₡63,29 por cada Kilowatt hora [72], equivale a ₡38,27 al día y ₡995,02 al mes.

Si se consume un recipiente de lubricante al mes que tiene un costo de ₡5 392,82 según el inventario de artículos de Polymer S.A.; el costo mensual de la operación del sistema de adquisición de datos es igual a ₡24 541,84 o el equivalente a

Tomando en cuenta que el sistema de medición realiza “rondas” completas cada 5 minutos, se puede calcular el valor aportado por su funcionamiento en términos monetarios.

Una “ronda” cada 5 minutos equivale a 12 rondas por hora o igual a 288 rondas al día. Estas 288 rondas al día tendrían un valor de ₡37 137,6 al día, equivalentes a ₡965 577,6 al mes.

El costo del proyecto sería completamente cubierto por el beneficio del sistema en un total de 3,18 meses (3 meses y 6 días aproximadamente).

El retorno de inversión mensual a partir de ese momento, tomando en cuenta los costos de operación sería igual a:

$$\frac{\text{€}965\,577,6 - \text{€}24\,541,84}{\text{€}24\,541,84} \times 100 = 3834,41\%$$

En la Figura 142., se puede observar el flujo de caja para los primeros 12 meses.

	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5	Mes 6	Mes 7	Mes 8	Mes 9	Mes 10	Mes 11	Mes 12
<b>Ingresos de Caja</b>	965577,6	965577,6	965577,6	965577,6	965577,6	965577,6	965577,6	965577,6	965577,6	965577,6	965577,6	965577,6
Cantidad de Rondas	7488	7488	7488	7488	7488	7488	7488	7488	7488	7488	7488	7488
Precio	128,95	128,95	128,95	128,95	128,95	128,95	128,95	128,95	128,95	128,95	128,95	128,95
<b>Egresos de Caja</b>	3096217,86	23546,82	23546,82	23546,82	23546,82	23546,82	23546,82	23546,82	23546,82	23546,82	23546,82	23546,82
Potencia Eléctrica	995,02	995,02	995,02	995,02	995,02	995,02	995,02	995,02	995,02	995,02	995,02	995,02
Costo Mensual de Plataforma Integromat	18154	18154	18154	18154	18154	18154	18154	18154	18154	18154	18154	18154
Lubricante	5392,82	5392,82	5392,82	5392,82	5392,82	5392,82	5392,82	5392,82	5392,82	5392,82	5392,82	5392,82
Costo Inicial (Mes 1)	3072671,04											
<b>Flujo del Mes</b>	-2130640,26	942030,8	942030,8	942030,8	942030,8	942030,8	942030,8	942030,8	942030,8	942030,8	942030,8	942030,8
<b>Flujo Acumulado</b>	-2130640,26	-1188609	-246579	695452,1	1637483	2579514	3521544	4463575	5405606	6347637	7289668	8231698

Figura 142. Flujo de Caja para Análisis Económico. Fuente: elaboración propia.

El flujo acumulado al final del segundo año es €19 536 068.

Esta información nos permite calcular el Valor Actual Neto, utilizando la tasa de interés igual a 10,31% [74] de la siguiente forma:

$$-\text{€}3\,072\,671,04 + \frac{\text{€}8\,231\,698}{1 + 0,1031} + \frac{\text{€}19\,536\,068}{(1 + 0,1031)^2}$$

$$-\text{€}3\,072\,671,04 + \text{€}7\,462\,331,61 + \text{€}16\,054\,892,02 = \text{€}20\,444\,552,59$$

Por lo tanto, el Valor Actual Neto del proyecto, a 2 años, es €20 444 552,59.

Ahora, la Tasa Interna de Retorno se calcula de la siguiente manera:

$$-\text{€}3\,072\,671,04 + \frac{\text{€}8\,231\,698}{1 + TIR} + \frac{\text{€}19\,536\,068}{(1 + TIR)^2}$$

$$0 = -3\,072\,671,04(1 + TIR)^2 + 8\,231\,698(1 + TIR) + 19\,536\,068$$

$$TIR = -15,16\% \mid TIR = 41,94\%$$

Donde TIR corresponde a la Tasa Interna de Retorno. De las anteriores soluciones, la de interés es 41,94%

Para contrastar con el costo de adquirir una nueva máquina extrusora por filmina soplada, en [77] su precio ronda desde los 15 000 USD hasta los 100 000 USD.

Adicionalmente, no se puede cuantificar el beneficio que el sistema completo traería a la producción en forma de prevención de pérdidas, cálculos automáticos y control de desperdicio pues no se lleva un control estadístico de la ocurrencia de estos eventos, sin embargo, las funciones del sistema de medición de ancho y velocidad diseñado tendrán un impacto positivo en el costo que estas actividades y eventos.

Los cálculos anteriores permiten así mismo validar la longevidad de los elementos pues cada uno de los sistemas está diseñado para funcionar mucho más que los 3 meses y 6 días en los que se recupera la inversión inicial y el tiempo adicional que tomaría cubrir el costo de sus repuestos que en el peor de los casos sería el 100% del proyecto.

De esta manera, queda completamente validado el diseño de un sistema de adquisición de datos sobre la medida de ancho del producto y velocidad de extrusión de la máquina E16 de la empresa Polymer en El Coyol de Alajuela.

## Capítulo 6: Conclusiones y Recomendaciones

### Conclusiones

- El sistema de adquisición de datos de velocidad puede medir y comunicar valores de velocidad desde los 0,5 m/min hasta los 417,38 m/min.
- La operación mecánica del sistema de adquisición de la velocidad de extrusión tiene un factor de seguridad de 341 y una vida útil teórica de 59703 años aproximadamente.
  
- El sistema de adquisición de datos del ancho puede detectar el plástico extruido con un error de precisión de 0,64 mm (desviación estándar 3,125 veces menor a la requerida) y una exactitud de hasta 99,3%.
- El factor de seguridad del ensamblaje mecánico del sistema de adquisición de ancho es de 15,17.
  
- El sistema de visualización muestra los datos de manera correcta y los registra cada 5 minutos.
- El sistema de comunicación puede comunicar fielmente los datos hacia un archivo Excel disponible en línea y en cualquier dispositivo móvil.
- El sistema de alerta visual alerta de manera correcta cuando la velocidad de extrusión es menor a 0,5 m/min y cuando los valores de ancho se salen de los límites permitidos en todas las ubicaciones. Además, se recibe una alerta por mensajería instantánea cuando la máquina tiene una velocidad de extrusión menor a 0,5 m/min.
  
- El costo del proyecto es compensado por su funcionamiento en aproximadamente 3 meses y 6 días, su Retorno de Inversión mensual a partir de ese momento es 3834,41%, su Valor Actual Neto es de ₡20 444 552,59 colones y su Tasa Interna de Retorno es del 41,94%.



## Recomendaciones

- Para evitar problemas generados por los LED RGB, es recomendable utilizar, en cambio, LEDs monocromáticos.
- Según [75] uno de los mejores coeficientes de fricción estática registrados para el acero es cuando tiene contacto con plomo, sin embargo, para el uso como recubrimiento para el rodillo del sistema de adquisición de datos de la velocidad, esta selección podría dar problemas de desgaste de las superficies. En [76: 442] se asegura que el coeficiente de fricción entre el acero y el caucho cuando existe deslizamiento puede llegar a ser el doble que el visto en [75] por lo que el caucho es una opción recomendable para el recubrimiento del rodillo.
- Las pruebas realizadas deben ser lo más parecido posible al entorno en el que se implementará el dispositivo de medición para evitar resultados engañosos.

## Capítulo 7: Referencias Bibliográficas

- [1] J. Fraden, *Handbook of Modern Sensors Physics, Designs, and Applications*. 4ta. Ed. New York: Springer. 2010
- [2] K. Kalantar-zadeh, *Sensors An Introductory Course*. New York: Springer. 2013
- [3] Y. Hugh D y R. A. Fredman, *Física universitaria, con física moderna*. México: Pearson. 2009
- [4] R. Hibbeler. *Mechanics of Materials*. Ed. 9. Estados Unidos de América: Pearson Prentice Hall. 2014.
- [5] R. G. Budynas y J. K. Nisbett. *Diseño en Ingeniería Mecánica de Shygley*. 8va. Ed. México: McGRAW-HILL/INTERAMERICANA EDITORES. 2008
- [6] P. Grech. *Introducción a la Ingeniería. Un enfoque a través del diseño*. 2da. Ed. Colombia: PEARSON. 2013
- [7] Mouser Electronics. *Encoders Incremental Codificadores* [En línea]. Disponible en: <https://www.mouser.co.cr/Electromechanical/Encoders/ /N-39xfc?P=1yys9mqZ1yzh66u>
- [8] Würth Elektronik. *WS-ENTV Mechanical Incremental Encoder Switch 11.7 x 13.75 x 26.6 mm Shaft with bushing*. Alemania. 2018
- [9] Allegro MicroSystems, Inc. *Sensitive Hall Effect Switches for High-Temperature Operation*. Hoja de datos 27621.6B, octubre 2005.
- [10] ST Microelectronics. *Proximity and ambient light sensing (ALS) module*, hoja de datos VL6180X, mayo 2016.
- [11] *USER'S MANUAL FOR GigE VISION CAMERAS*, 8va ed. Basler. 2019.
- [12] Catálogo No. P37E-23, Piezoelectric Sound Components, muRata, Nagaokakyo-shi, OP.
- [13] ElecFreaks. *Ultrasonic Ranging Module HC – SR04*. 1era ed.
- [14] Basler AG. *raL2048-48gm - Basler racer*. 2021. Disponible en: <https://www.baslerweb.com/en/products/cameras/line-scan-cameras/racer/ral2048-48gm/> [accesado: Abr 08, 2021]
- [15] CrCibernética. *CrCibernetica: Open Source Hardware*. 2021. Disponible en: <https://www.crcibernetica.com/> [accesado: Abr 08, 2021]
- [16] Micro JPM. *Inicio*. 2020. Disponible en: <https://www.microjpm.com/> [accesado: Abr 08, 2021]

- [17] Mouser Electronics, Inc. *Products*. 2021. Disponible en: <https://www.mouser.co.cr/> [accesado: Abr 08, 2021]
- [18] Digi-Key ELECTRONICS. *11484*. 2021. Disponible en: <https://www.digikey.com/en/products/detail/allied-vision-inc./11484/11200664> [accesado: Abr 17, 2021]
- [19] CrCibernética. *Stepper Motor Nema17*. 2021. Disponible en: <https://www.crcibernetica.com/stepper-motor-nema17/> [accesado: Abr 17, 2021]
- [20] Yueqing Hengwei Electronics Co., LTD. *Conductive Proximity Switch Series*. 1era ed.
- [21] Interlink Electronics. *FSR 402*. Hoja de datos P/N: 30-81794. Rev: A.
- [22] Vishay Semiconductors. *Reflective Optical Sensor with Transistor Output*. Hoja de datos 83751, julio 2012.
- [23] B. Schmalz. *EasyDriver Stepper Motor Driver*. 1era Ed. 2012.
- [24] Microchip Technology Inc. *PIC18F2455/2550/4455/4550*. Hoja de datos DS39617A, julio 2003.
- [25] Atmel Corporation. *8-bit AVR Microcontroller with 32K Bytes In-System Programmable Flash*. Hoja de datos 7810D-AVR-01/15, enero 2015.
- [26] Atmel Corporation. *8-bit AVR Microcontroller with 2/4/8K Bytes In-System Programmable Flash*. Hoja de datos 8006K-AVR-10/10, octubre 2010.
- [26] Catálogo No. E1102K, Rodamientos, NSK, Cuerpo Bajo, BCN.
- [27] *Guizero*. (1.1.1), Lawsie. Accesado: 04 de mayo de 2021 [en línea]. Disponible en: <https://lawsie.github.io/guizero/>
- [28] *Tkinter* (8.5). Disponible en: [https://github.com/python/cpython/blob/main/Lib/tkinter/\\_init\\_.py](https://github.com/python/cpython/blob/main/Lib/tkinter/_init_.py)
- [29] O. Gazda, P. Simek. *Integromat* 2021. Accesado: 04 de mayo de 2021 [en línea]. Disponible en: <https://www.integromat.com>
- [30] C. Lietchi. *pySerial* (2017). Accesado: 04 de mayo de 2021 [en línea]. Disponible en: <https://pyserial.readthedocs.io/en/latest/shortintro.html>
- [31] *threading* (3.7) Disponible en: <https://github.com/python/cpython/blob/3.9/Lib/threading.py>
- [32] K. Reitz. *Requests* (2.25.1) Accesado: 05 de mayo de 2021 [en línea]. Disponible en: <https://pypi.org/project/requests/>
- [33] Telegram. *Bots: An introduction for developers* (2021). Accesado: 05 de mayo de 2021 [en línea]. Disponible en: <https://core.telegram.org/bots>

- [34] `datetime` (3.8) Disponible en: <https://github.com/python/cpython/tree/3.9/Lib/datetime.py>
- [35] hippy. *P3B+, Get the USB port name* (Jul 11, 2021). Accesado: 05 de mayo de 2021 [en línea]. Disponible en: <https://www.raspberrypi.org/forums/viewtopic.php?t=217209>
- [36] Fairchild Semiconductor Corporation. *CD4071BC-CD4081BC*. Hoja de datos DS005977. Octubre 1987.
- [37] B. Nutall, D. Jones. *gpiozero* (2.x). Accesado: 05 de mayo de 2021 [en línea]. Disponible en: <https://gpiozero.readthedocs.io/en/stable/>
- [38] Microsoft. *Microsoft 365* (2021). Accesado: 05 de mayo de 2021 [en línea]. Disponible en: <https://www.microsoft.com>
- [39] *shelve* (abr. 9, 2018). Accesado: 11 de mayo de 2021 [en línea]. Disponible en: <https://github.com/python/cpython/blob/3.9/Lib/shelve.py>
- [40] Allegro MicroSystems, Inc. *Sensitive Hall Effect Switches for High-Temperature Operation*. Hoja de datos 27621.6B, octubre 2005.
- [41] United States Plastic Corp. *Physical Properties for Quadrant Nylon 101 Sheet*. 2013. Accesado: 06 de mayo de 2021 [en línea]. Disponible en: <https://www.usplastic.com/knowledgebase/article.aspx?contentkey=530>
- [42] AZO materials. *AISI 4140 Alloy Steel (UNS G41400)*. Accesado: 06 de mayo de 2021 [en línea]. Disponible en: <https://www.azom.com/article.aspx?ArticleID=6769>
- [43] Dassault Systèmes, Vélizy-Villacoublay, Francia, *Solidworks*, ver SP2.0, enero 11, 2021. Disponible en: <https://my.solidworks.com/try-solidworks> [accesado: May 07, 2021]
- [44] AZO materials. *Neodymium - History, Properties and Applications*. Accesado: 06 de mayo de 2021 [en línea]. Disponible en: <https://www.azom.com/properties.aspx?ArticleID=1595>
- [45] Otai Steel. *ASTM 52100 Bearing Steel | 1.3505 | 100Cr6 | SUJ2 | EN3*. Accesado: 06 de mayo de 2021 [en línea]. Disponible en: <http://www.astmsteel.com/product/52100-bearing-steel-aisi/>
- [46] Microchip Technology Inc. *ATmega48A/PA/88A/PA/168A/PA/328/P*. Hoja de datos DS40002061A, 2018.
- [47] Microchip Technology Inc. *Microchip Studio for AVR and SAM Devices*, ver 7.0.2542. Disponible en: <https://www.microchip.com/en-us/development-tools-tools-and-software/microchip-studio-for-avr-and-sam-devices>
- [48] Free Software Foundation, Inc. *GCC, the GNU Compiler Collection*, (19 de mayo de 2019). Disponible en: <https://gcc.gnu.org/>

- [49] *avr-libc* (09 de febrero de 2016). Disponible en: <https://www.nongnu.org/avr-libc/>
- [50] The IEEE and The Open Group. *stdio.h – standard buffered input/output*. (2004). Disponible en: <https://pubs.opengroup.org/onlinepubs/009695399/basedefs/stdio.h.html>
- [51] Microchip Makes. “Getting Started with AVR: Splitting Functions into USART.h and.c Files (#19)” *YouTube*, 03 de abril, 2015. Disponible en: [https://www.youtube.com/watch?v=almRz0\\_m9g&list=PLtQdQmNK\\_0DRhBWYZ32BEILOykXLpJ8tP&index=19](https://www.youtube.com/watch?v=almRz0_m9g&list=PLtQdQmNK_0DRhBWYZ32BEILOykXLpJ8tP&index=19) [Accesado: 26 de mayo, 2021]
- [52] J.D. Hunter. *matplotlib* (3.4.2). Disponible en: <https://matplotlib.org/stable/users/installing.html>
- [53] The SciPy Community. *NumPy* (31 de enero de 2021). Disponible en: <https://numpy.org/install/>
- [54] Python Software Foundation. *Python* (3.9.5). Disponible en: <https://www.python.org/downloads/>
- [55] Prolific Technology Inc. *PL-2303HX Edition (Chip Rev D) USB to Serial Bridge Controller Product Datasheet*. Hoja de datos ds\_pl2303HXD\_v1.4.4, 2013.
- [56] Autodesk Inc. *EAGLE* (ver. 9.6.2) Disponible en: <https://www.autodesk.com/products/eagle/overview?term=1-YEAR>
- [57] Texas Instruments. *LM2596 SIMPLE SWITCHER Power Converter 150- kHz 3-A Step- Down Voltage Regulator*. Hoja de datos SNVS124F, 2021.
- [58] B. Schmalz. *EasyDriver Stepper Motor Driver*. 1era Ed. 2012.
- [59] The Raspberry Pi Foundation. *Raspberry Pi 4*. Disponible en: <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/> [accesado: 06 de julio de 2021]
- [60] The Raspberry Pi Foundation. *Raspberry Pi OS with Desktop* (07 de mayo de 2021). Disponible en: <https://www.raspberrypi.org/software/operating-systems/#raspberrypi-os-32-bit>
- [61] Free Software Foundation. *GNU/Linux* (ver. 5.12.13)
- [62] Extreme Tech Costa Rica. *extremetech*. Disponible en: <https://extremetechcr.com/tienda/> [Accesado: 08 de julio de 2021]
- [63] Almacenes Siman Costa Rica. *siman.com*. Disponible en: <https://cr.siman.com/> [Accesado: 08 de julio de 2021]
- [64] Amazon Web Services, Inc. *¿Qué es AWS?* (2021) Disponible en: <https://aws.amazon.com/es/what-is-aws/> [Accesado: 15 de julio de 2021]

- [65] Amazon Web Services. "AWS Quick Start - How AWS Pricing Works *YouTube*, 23 de noviembre, 2018. Disponible en: <https://www.youtube.com/watch?v=WrBmmcwh-BA> [Accesado: 15 de julio de 2021]
- [66] Google, LLC. *Servicios de cloud computing* (2021) Disponible en: <https://cloud.google.com/> [Accesado: 15 de julio de 21]
- [67] Google, LLC. *Google Cloud Pricing Calculator* (2021) Disponible en: <https://cloud.google.com/products/calculator?id=7888ca99-beca-4e08-8f3e-631721fe1112> [Accesado: 15 de julio de 21]
- [68] Teknika4. *Aluminio 6061* (2021) Disponible en: <https://www.teknika4.com/es/aluminio-6061> [Accesado: 20 de julio de 2021]
- [69] Ferguson Perforating. *6061 Aluminium Alloy* (2021) Disponible: <https://www.fergusonperf.com/the-perforating-process/material-information/specialized-aluminum/6061-aluminium-alloy/> [Accesado: 20 de julio de 2021]
- [70] Thomson Industries, Inc. *A81420* (2021) Disponible en: <https://www.thomsonlinear.com/en/product/A81420> [Accesado: 23 de julio de 2021]
- [71] Ministerio de Trabajo y Seguridad Social. *Lista de salarios por ocupación año 2020* (2020) Disponible en: [https://www.mtss.go.cr/temas-laborales/salarios/Documentos-Salarios/lista\\_salarios\\_2020.pdf](https://www.mtss.go.cr/temas-laborales/salarios/Documentos-Salarios/lista_salarios_2020.pdf) [Accesado: 26 de julio de 2021]
- [72] CNFL. *Tarifa Industrial T-IN* (2021) Disponible en: <https://www.cnfl.go.cr/servicios-electricos-para-inmuebles/tarifas-vigentes/tarifa-industrial-t-in> [Accesado: 27 de julio de 2021]
- [73] Banco Central de Costa Rica. *Tipos de cambio anunciados en ventanilla por los intermediarios cambiarios* (2021) Disponible en: <https://gee.bccr.fi.cr/IndicadoresEconomicos/Cuadros/frmConsultaTCVentanilla.aspx> [Accesado: 20 de julio de 2021]
- [74] Ministerio de Hacienda. *Listado de tasas de interés (vigente e histórico)* (agosto de 2021) Disponible en: <https://www.hacienda.go.cr/contenido/12998-listado-de-tasas-de-interes-vigente-e-historico> [Accesado: 05 de agosto de 2021]
- [75] D. D. Fuller, *Theory and Practice of Lubrication of Lubrication for Engineers*. Universidad de Michigan. 1956.
- [76] F. L. Roth, R. L. Driscoll y W. L. Holt, "Frictional Properties of Rubber," *Journal of Research of the National Bureau of Standards.*, vol. 28, pp. 442, abril 1942.

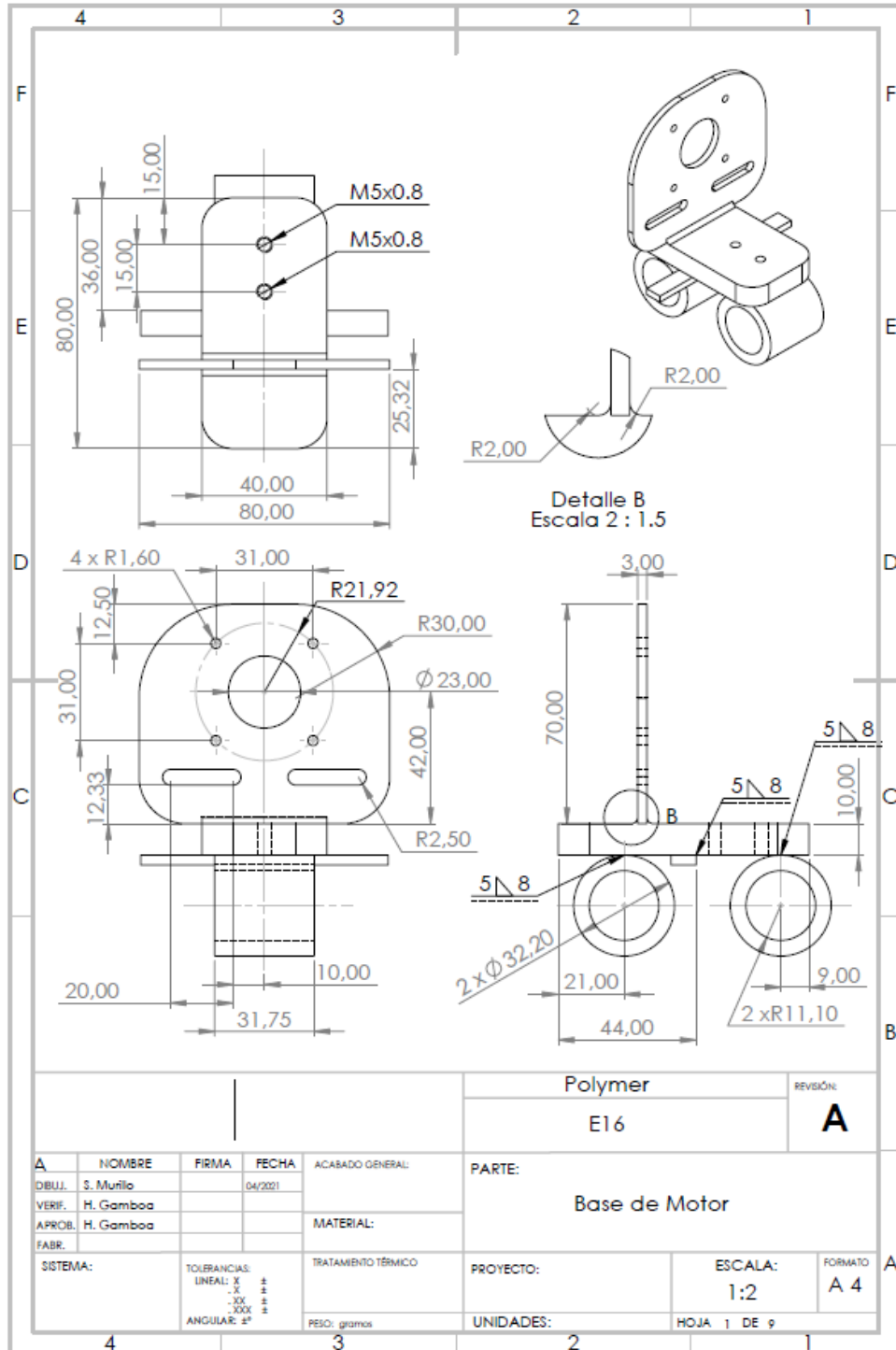
[77] Focus Technology Co., *Made-in-China* Disponible en: <https://zxpanyuying.en.made-in-china.com/product/keywordSearch?searchKeyword=film+blown+machine&viewType=1> [Accesado: 9 de agosto de 2021]

[78] Caja Costarricense de Seguro Social, *Patronos trabajadores y Estado aportarán más al régimen IVM a partir de enero 2020* (diciembre de 2019) Disponible en: [https://www.ccss.sa.cr/noticias/servicios\\_noticia?patronos-trabajadores-y-estado-aportaran-mas-al-regimen-ivm-a-partir-de-enero-2020](https://www.ccss.sa.cr/noticias/servicios_noticia?patronos-trabajadores-y-estado-aportaran-mas-al-regimen-ivm-a-partir-de-enero-2020) [Accesado: 16 de agosto de 2021]

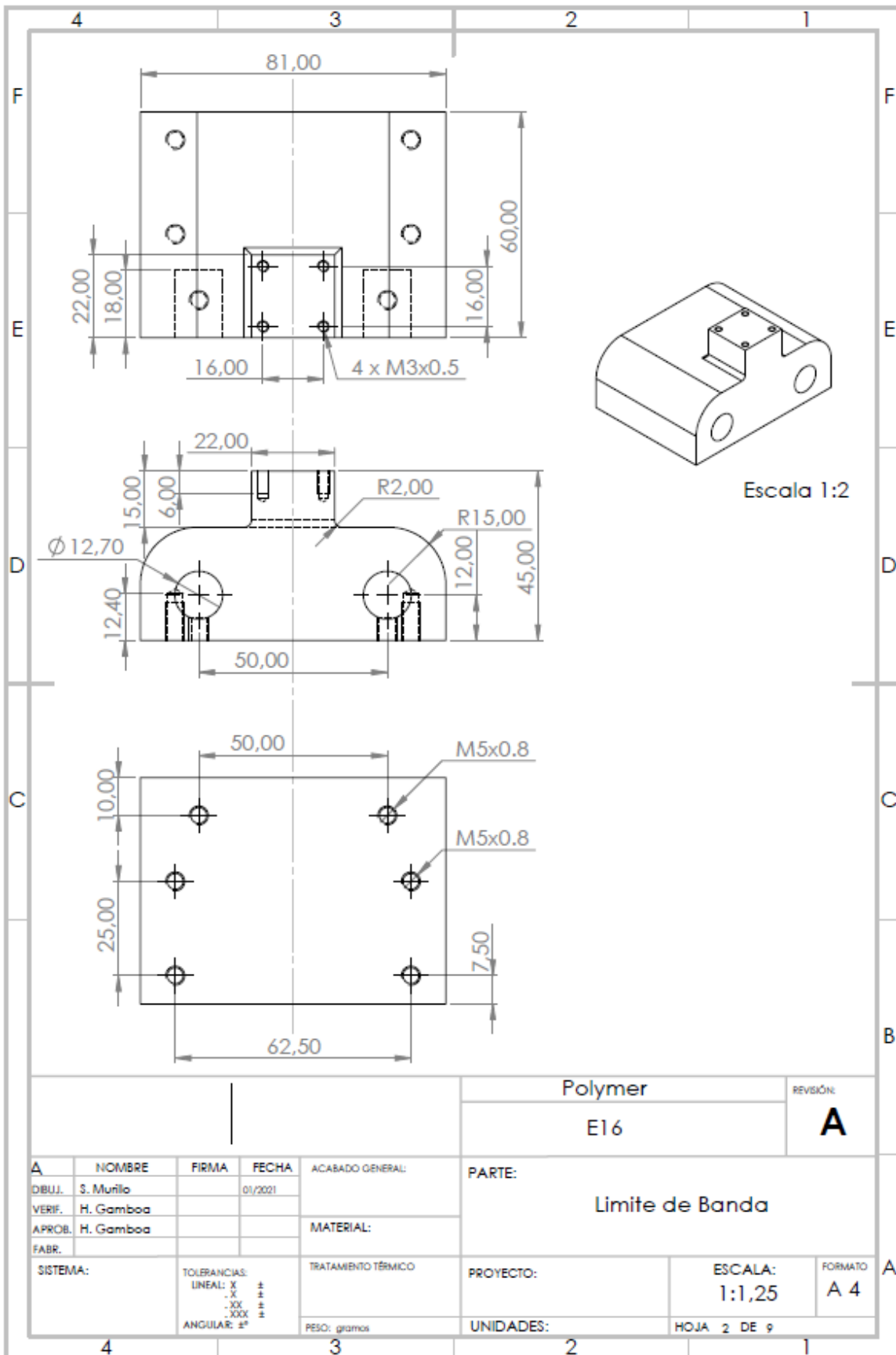
[79] Universidad de Tartu, *Thonny* (ver 3.3) Disponible en: <https://thonny.org/>

# Capítulo 8: Apéndices

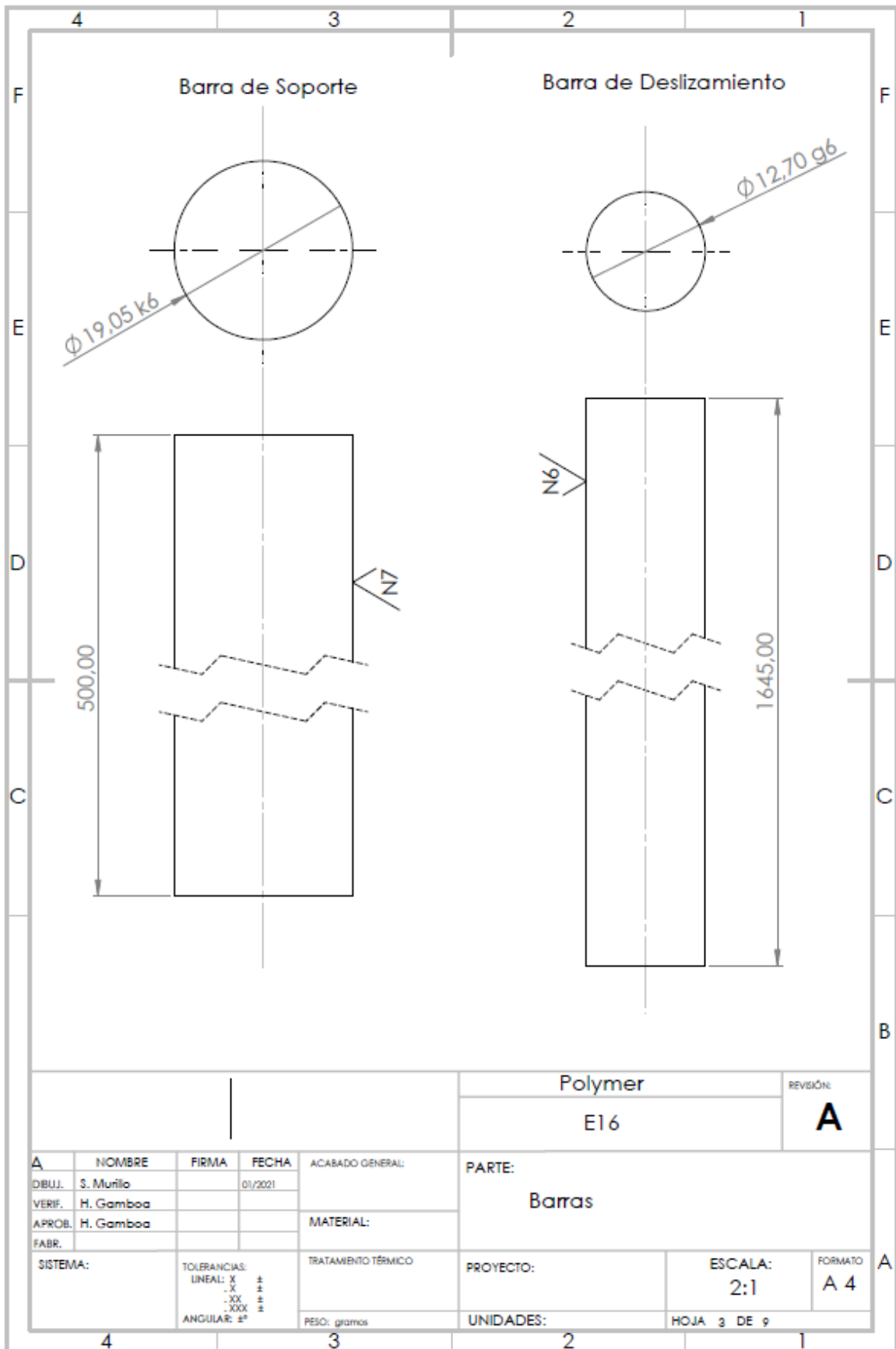
## Apéndice A - Planos del Sistema de Medición de Ancho



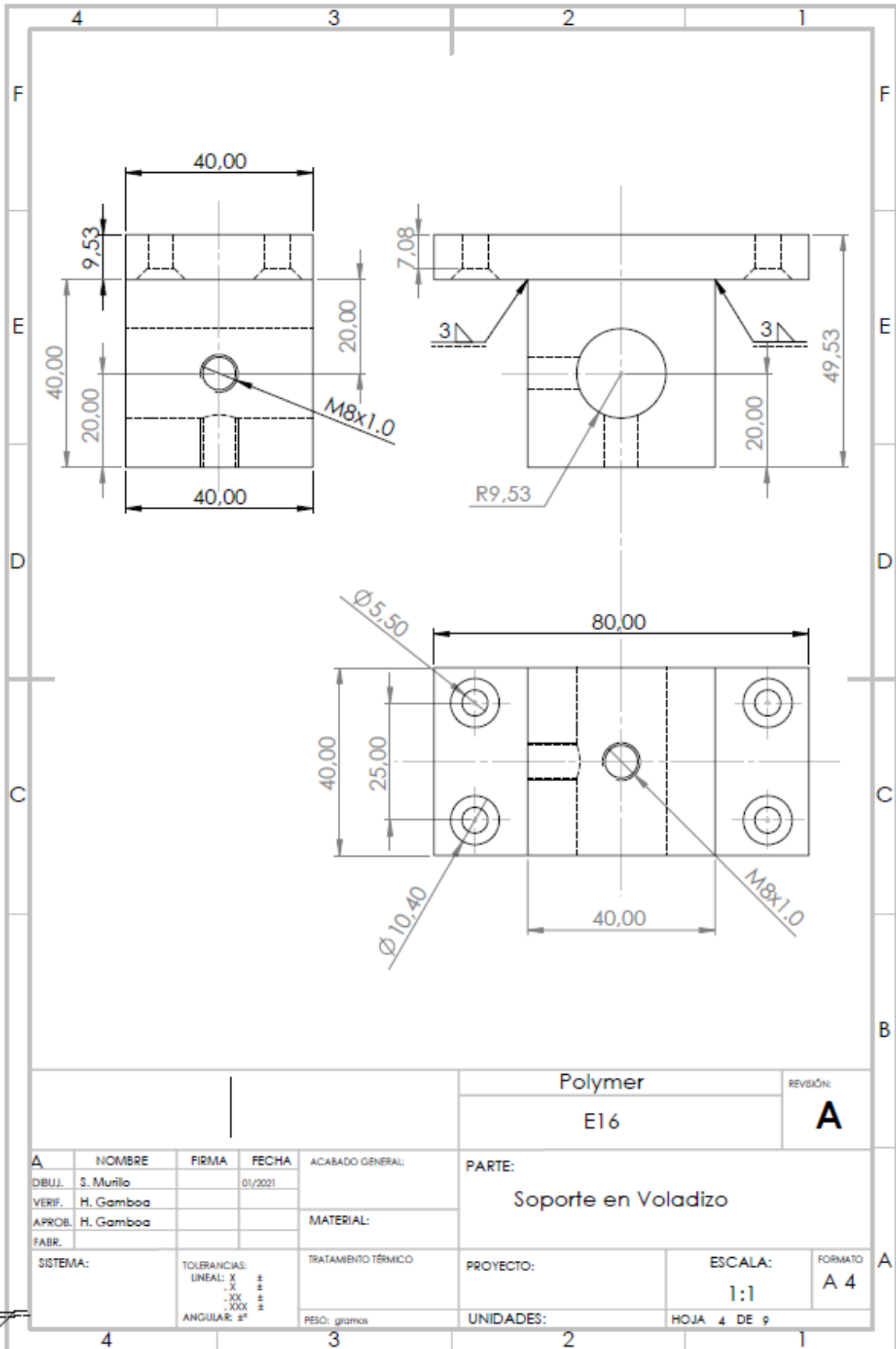




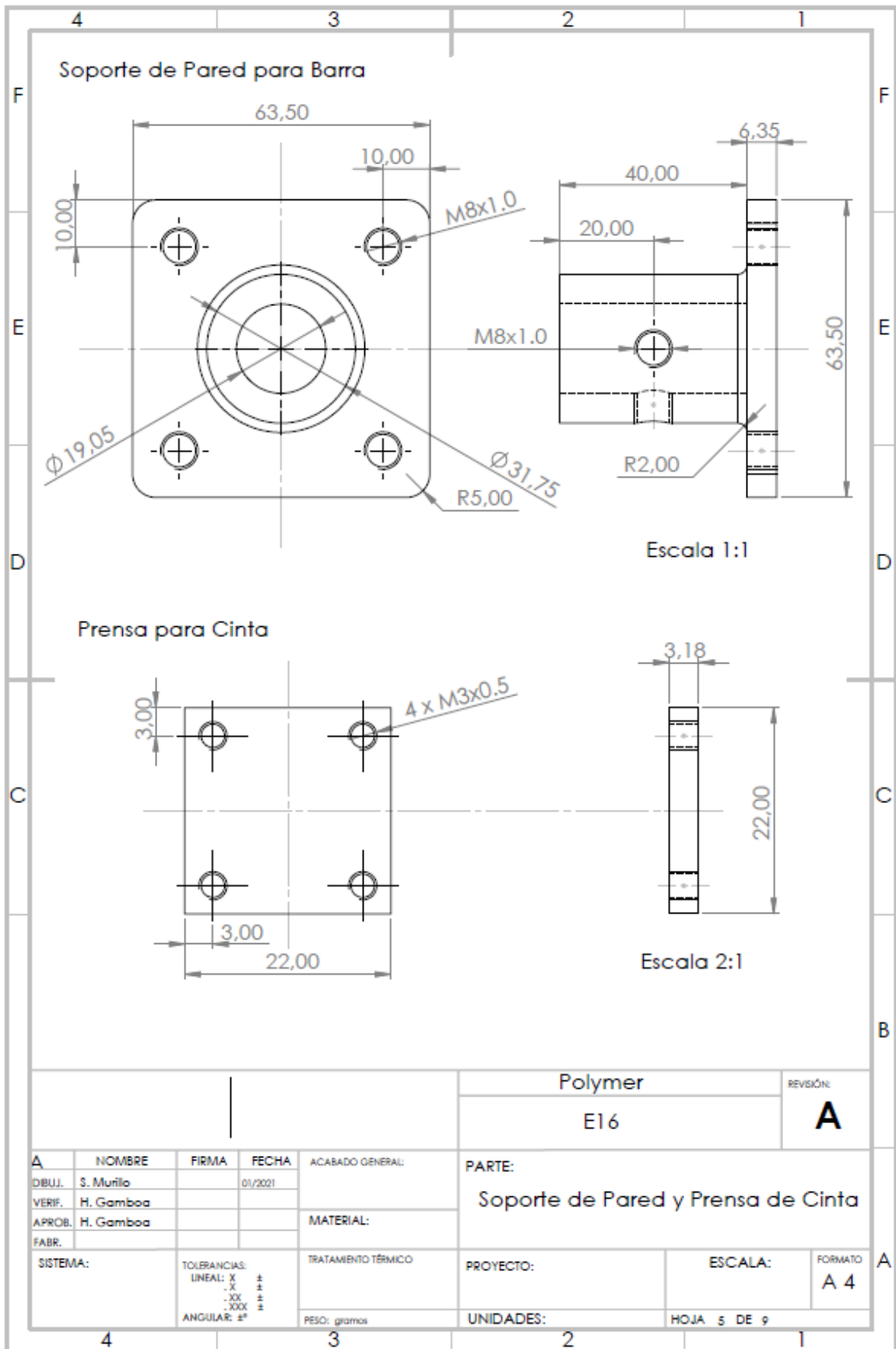
				Polymer		REVISIÓN:
				E16		<b>A</b>
<b>A</b>	NOMBRE	FIRMA	FECHA	ACABADO GENERAL:	PARTE:	
	DBUJ. S. Murilo		01/2021		Limite de Banda	
	VERIF. H. Gamboa			MATERIAL:		
	APROB. H. Gamboa			TRATAMIENTO TÉRMICO	PROYECTO:	ESCALA:
	FABR.					1:1,25
	SISTEMA:	TOLERANCIAS:		PESO: gramos	UNIDADES:	FORMATO
		LINEAL: X ±				A 4
		.X ±				
		.XX ±				
		.XXX ±				
		ANGULAR: ±°				
					HOJA 2 DE 9	



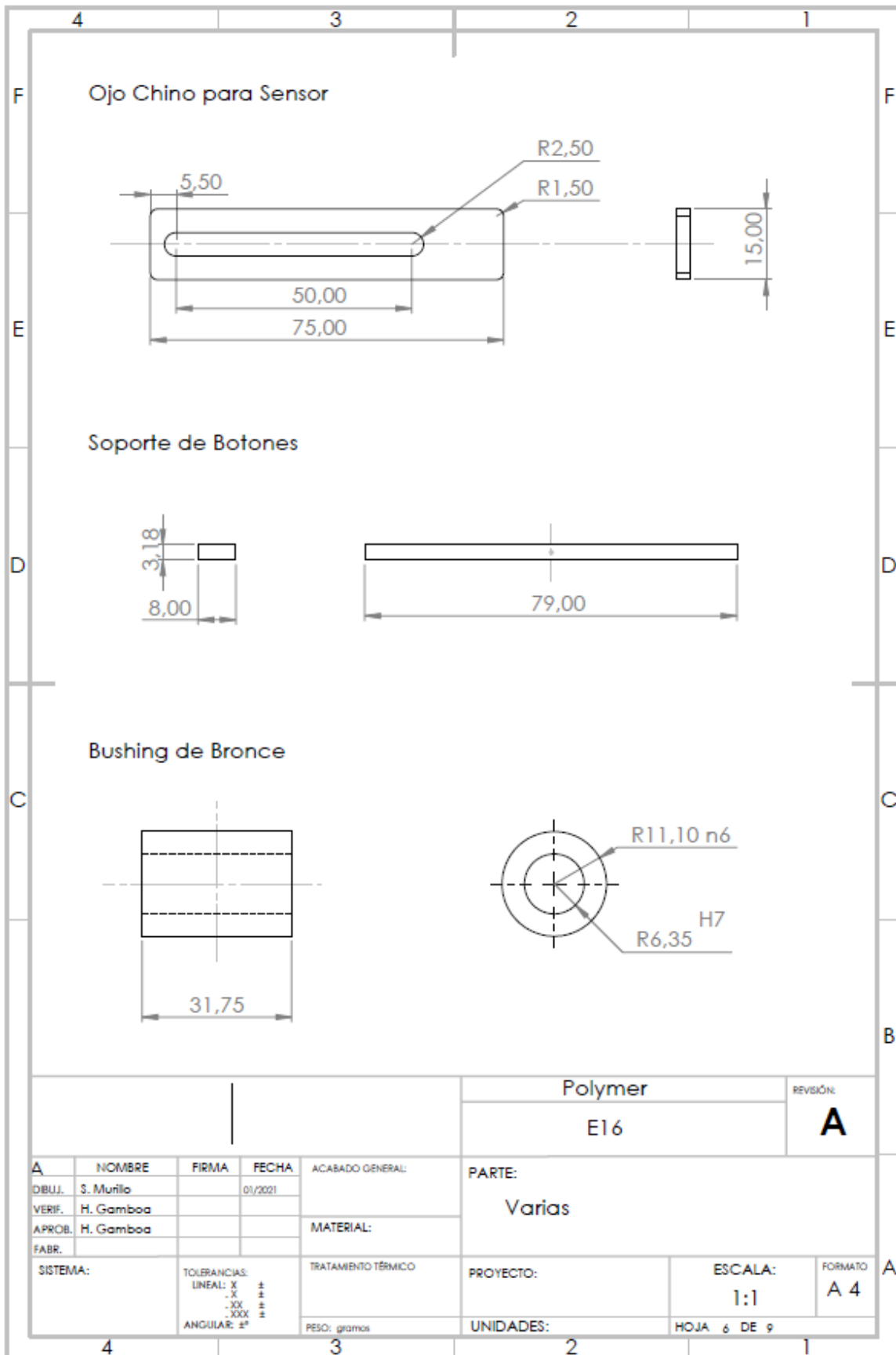
				Polymer		REVISIÓN:
				E16		<b>A</b>
<b>A</b>	NOMBRE	FIRMA	FECHA	ACABADO GENERAL:	PARTE:	
DIBUJ.	S. Murillo		01/2021		Barras	
VERIF.	H. Gamboa			MATERIAL:		
APROB.	H. Gamboa					
FABR.						
SISTEMA:		TOLERANCIAS:		TRATAMIENTO TÉRMICO		PROYECTO:
		LINEAL: X ±				ESCALA:
		- X ±				2:1
		.XX ±				FORMATO
		.XXX ±				A 4
		ANGULAR: ±°		PESO: gramos		UNIDADES:
						HOJA 3 DE 9



				Polymer		REVISIÓN:
				E16		<b>A</b>
<b>A</b>	NOMBRE	FIRMA	FECHA	ACABADO GENERAL:	PARTE:	
	DIBUJ. S. Murillo		01/2021		Soporte en Voladizo	
	VERIF. H. Gamboa			MATERIAL:		
	APROB. H. Gamboa			TRATAMIENTO TÉRMICO	PROYECTO:	ESCALA:
	FABR.					1:1
SISTEMA:		TOLERANCIAS: LINEAL: X ± .XX ± .XXX ± ANGULAR: ±°		PESO: gramos	UNIDADES:	FORMATO A 4
					HOJA 4 DE 9	

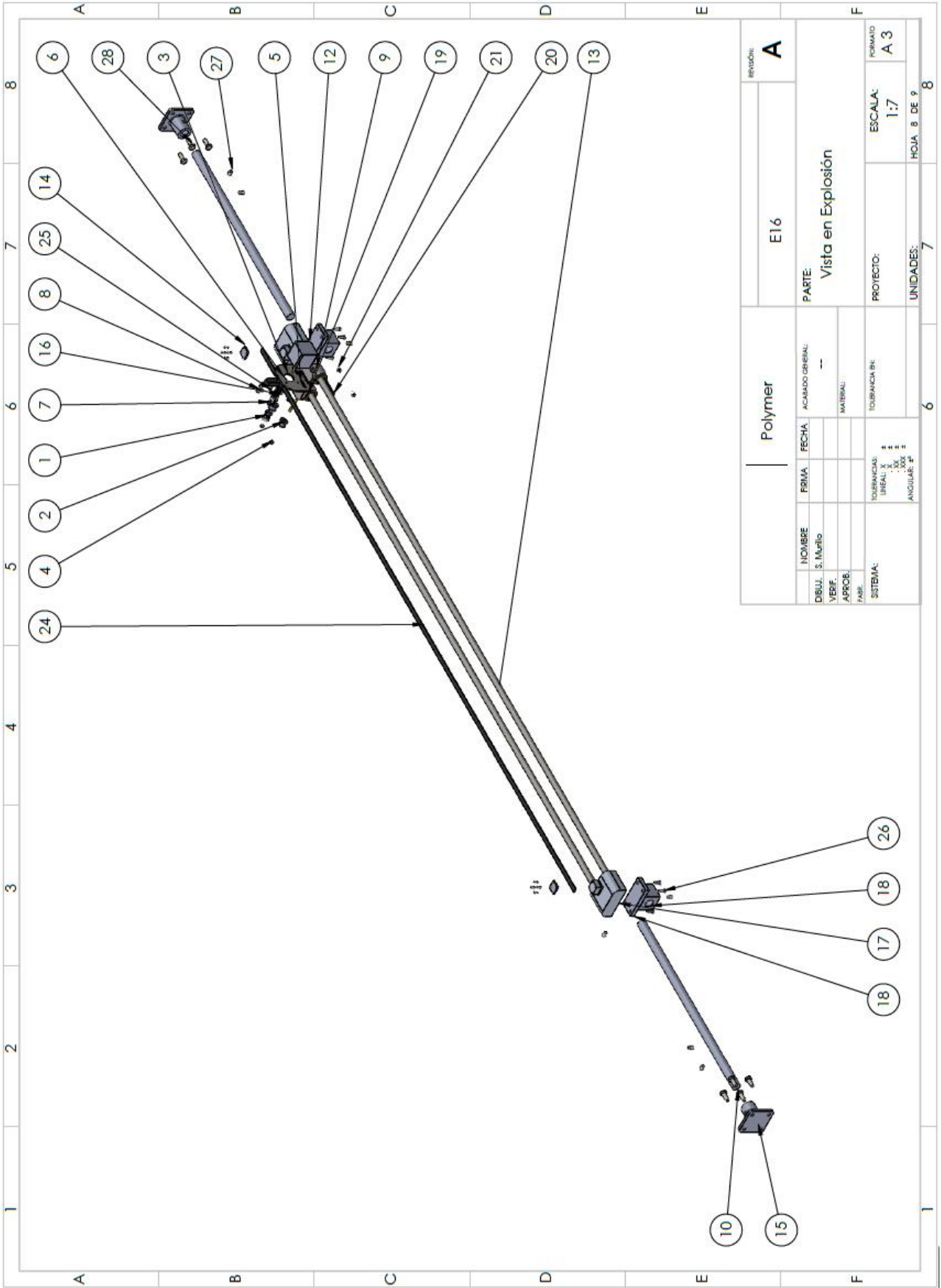


				Polymer		REVISIÓN:
				E16		<b>A</b>
<b>A</b>	NOMBRE	FIRMA	FECHA	ACABADO GENERAL:	PARTE:	
DIBUJ.	S. Murillo		01/2021		Soporte de Pared y Prensa de Cinta	
VERIF.	H. Gamboa			MATERIAL:		
APROB.	H. Gamboa			TRATAMIENTO TÉRMICO	PROYECTO:	ESCALA:
FABR.					UNIDADES:	FORMATO
SISTEMA:	TOLERANCIAS:					A 4
	LINEAL: X ±					
	- X ±					
	.XX ±					
	.XXX ±					
	ANGULAR: ±°			PESO: gramos	HOJA 5 DE 9	





NOMBRE		PRIMA		FECHA		ACABADO ORIGINAL		REVISION:	
DIBUJ. S. Murillo						---		A	
VERIF.								PARTES:	
APROB.								Ensamblaje Completo	
FABR.								PROYECTO:	
SISTEMA:								ESCALA:	
TOLERANCIAS:								1:5.5	
LINEAL: ±								FORMA:	
ANGULAR: ±								A 3	
								UNIDADES:	
								7	
								HOJA 7 DE 9	
								6	
								8	

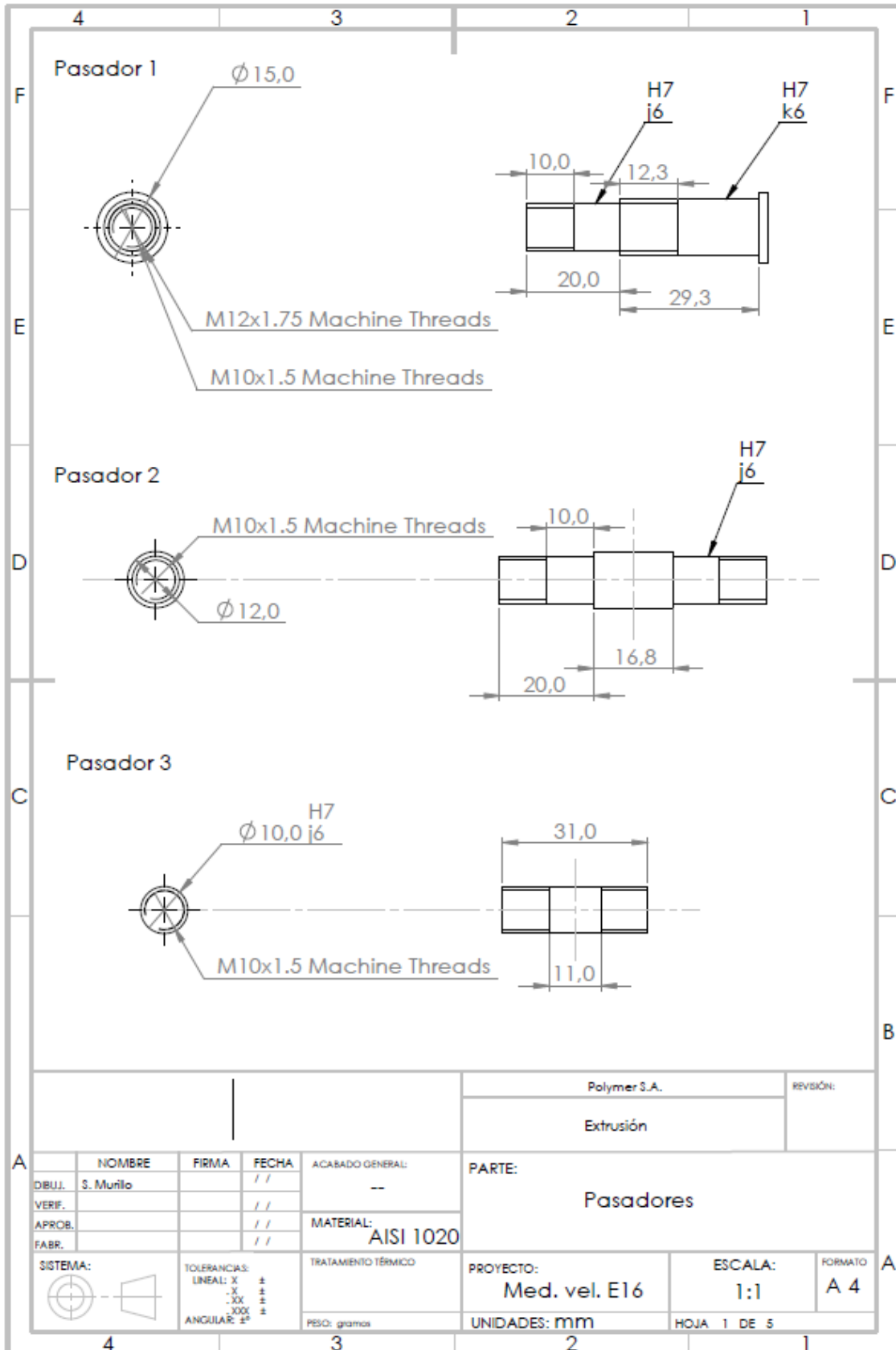


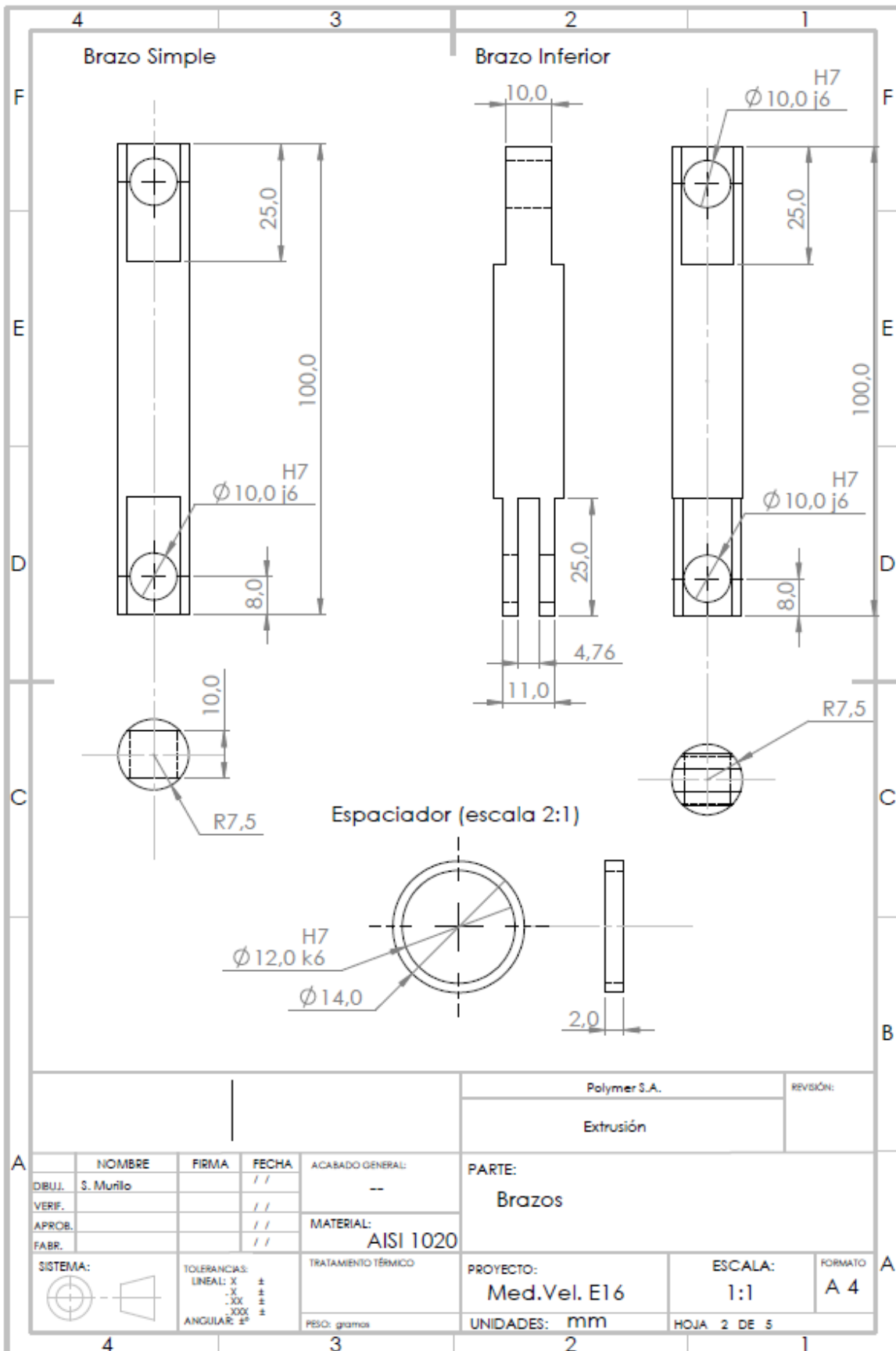
POLYMER		E16		REVISION: A	
NOMBRE		ACABADO GENERAL:		PARTE:	
DISEÑO: S. Muñoz		---		Vista en Explosión	
VERIF.:		MATERIAL:		ESCALA:	
APROB.:		TOLERANCIA EN:		1:7	
FABR.:		TOLERANCIA EN:		FORMA:	
SISTEMA:		TOL. LINEAL: ±		A 3	
		TOL. ANGULAR: ±		UNIDADES:	
				HOJA 8 DE 9	
				7	
				8	

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Polea fija/ conductora de tipo GT2	20 dientes/diámetro 10mm y calibre 5mm	1
2	Polea móvil/conducida GT2	Diámetro 10 mm y calibre 5mm	2
3	Base		1
4	B18.2.4.1M - Hex nut, Style 1, M5 x 0.8 --D-C	Tuerca M5 x 0,8	2
5	Bushing		2
6	Ojo Chino		1
7	Sensor CNY70		1
8	B18.6.7M - M5 x 0,8 x 13 Slotted PHMS --13C	Tornillo Plano M5 x 0,8 x 13	2
9	B18.2.4.5M - Hex jam nut, M5 x 0.8 --D-N	Tuerca Delgada M5 x 0,8	2
10	Barra de Soporte		2
11	Ensamblaje de soporte en voladizo		2
12	Motor Nema 17		1
13	Barra de deslizamiento 4		2
14	Lamina de prensa para cinta		2
15	Soporte para Pared		2
16	Pasador		2
17	Limite de Banda 5		2
18	Ensamblaje de Soporte en Voladizo 5		1
19	Bushing de Bronce		2
20	Soporte de Botones		1
21	Botón Push		2
22	Lamina soldada en soporte		1
23	Soporte en voladizo 4		1
24	Cinta GT2	10 mm x 1,6m	1
25	B18.3.4M - 3 x 0.5 x 10 SBHCS --N	Tornillo Allen M3 x 0,5 x 10	12
26	B18.6.7M - M5 x 0.8 x 16 Type I Cross Recessed FHMS --16N	Tornillo Avellanado M5 x 0,8 x 16	8
27	B18.3.6M - M8 x 1.25 x 12 Hex Socket Oval Pt. SS --N	Prisionero Allen M8 x 1,25 x 12	8
28	B18.2.3.2M - Formed hex screw, M8 x 1.25 x 20 --20WN	Tornillo Hexagonal M8 x 1,25 x 20	8

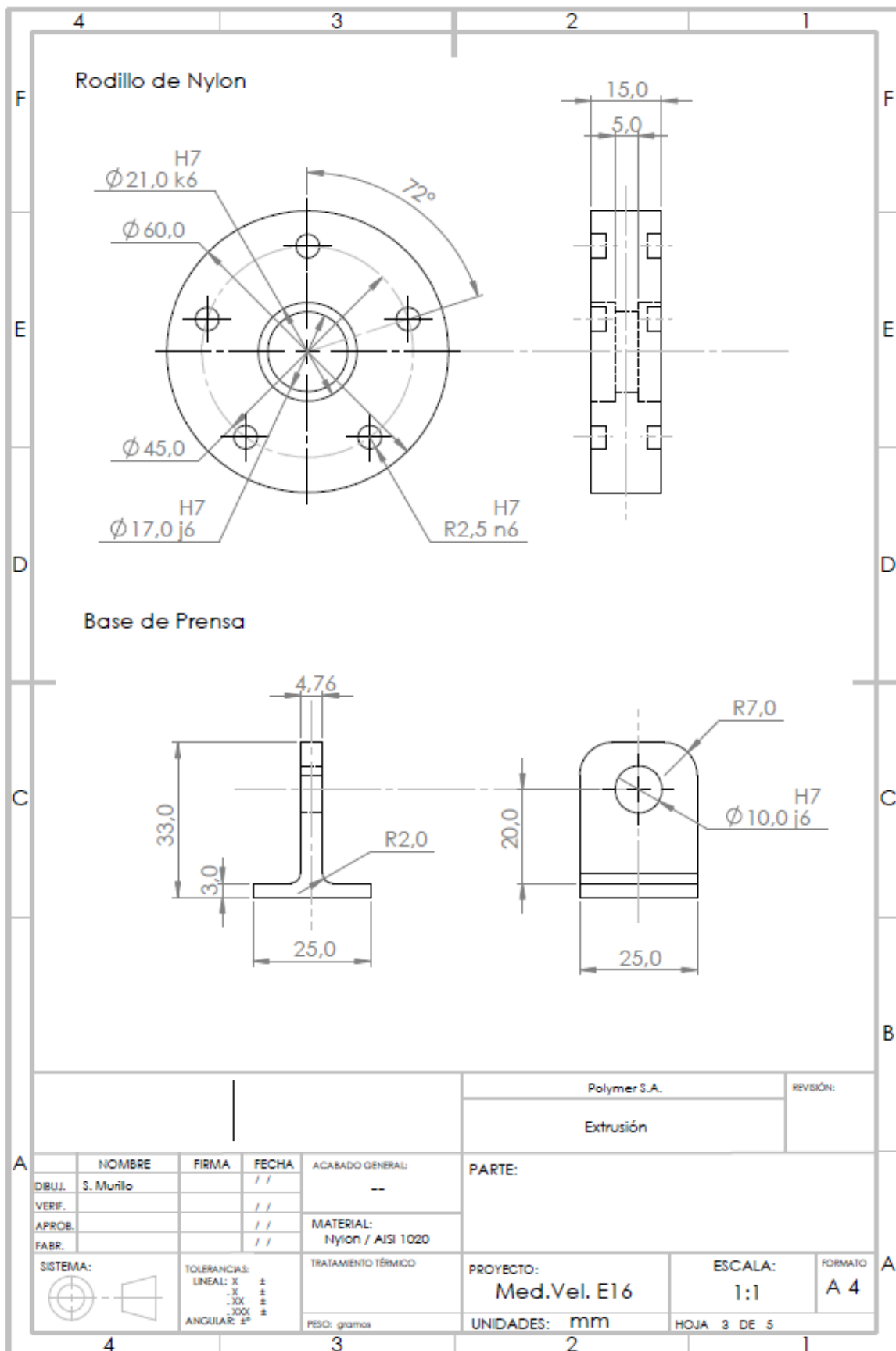


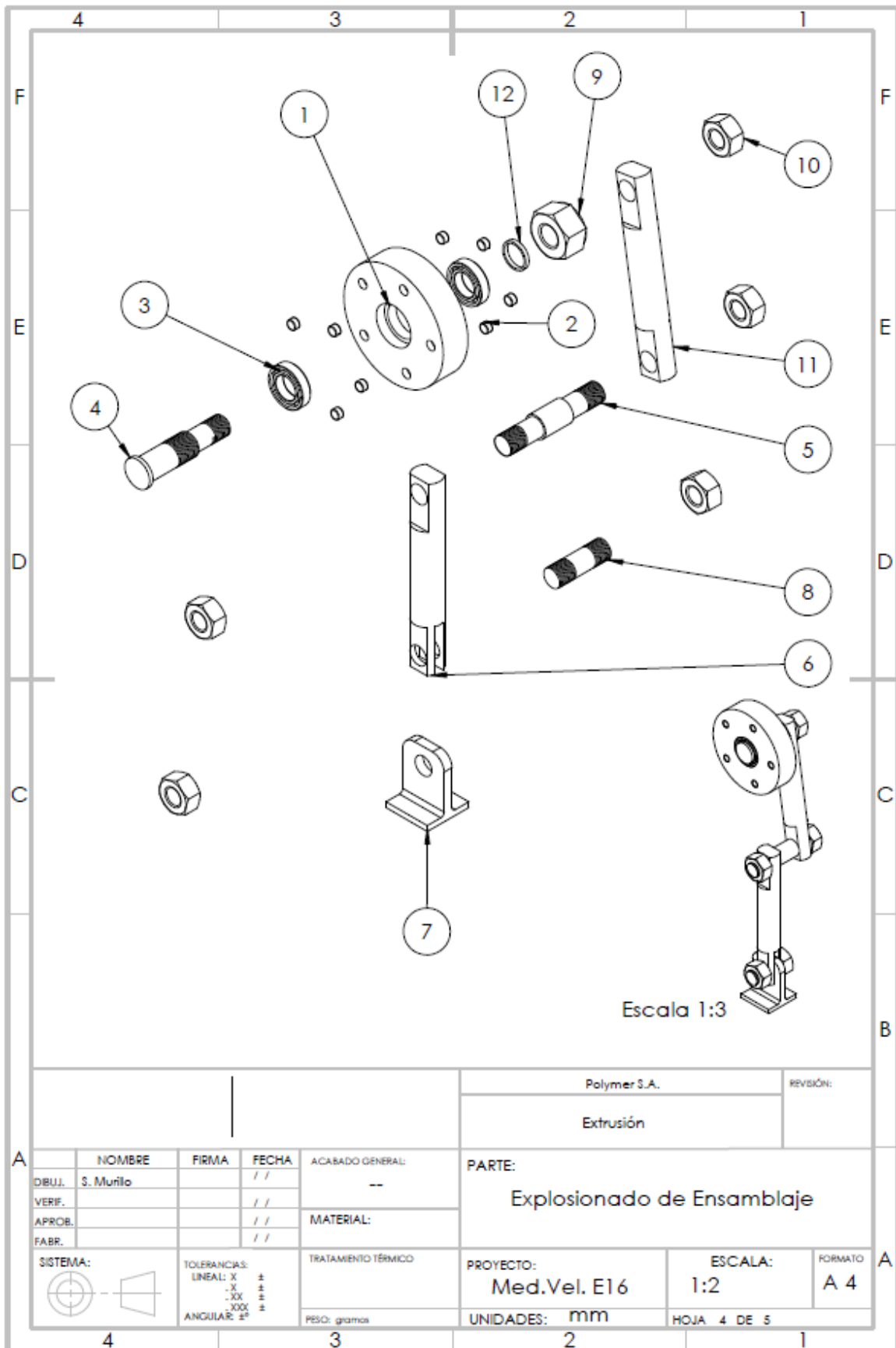
## Apéndice B – Planos del Sistema de Adquisición de Velocidad






				Polymer S.A.		REVISIÓN:
				Extrusión		
A	NOMBRE	FIRMA	FECHA	ACABADO GENERAL:	PARTE:	
	DIBUJ.	S. Murillo	///	--	Brazos	
	VERIF.		///	MATERIAL:		
	APROB.		///	AISI 1020		
FABR.		///				
SISTEMA:		TOLERANCIAS:		TRATAMIENTO TÉRMICO	PROYECTO:	ESCALA:
		LINEAL: X ± X ± .XX ± .XXX ± ANGULAR: 2° ±			Med.Vel. E16	1:1
		PESO: gramos		UNIDADES: mm	HOJA 2 DE 5	FORMATO A 4



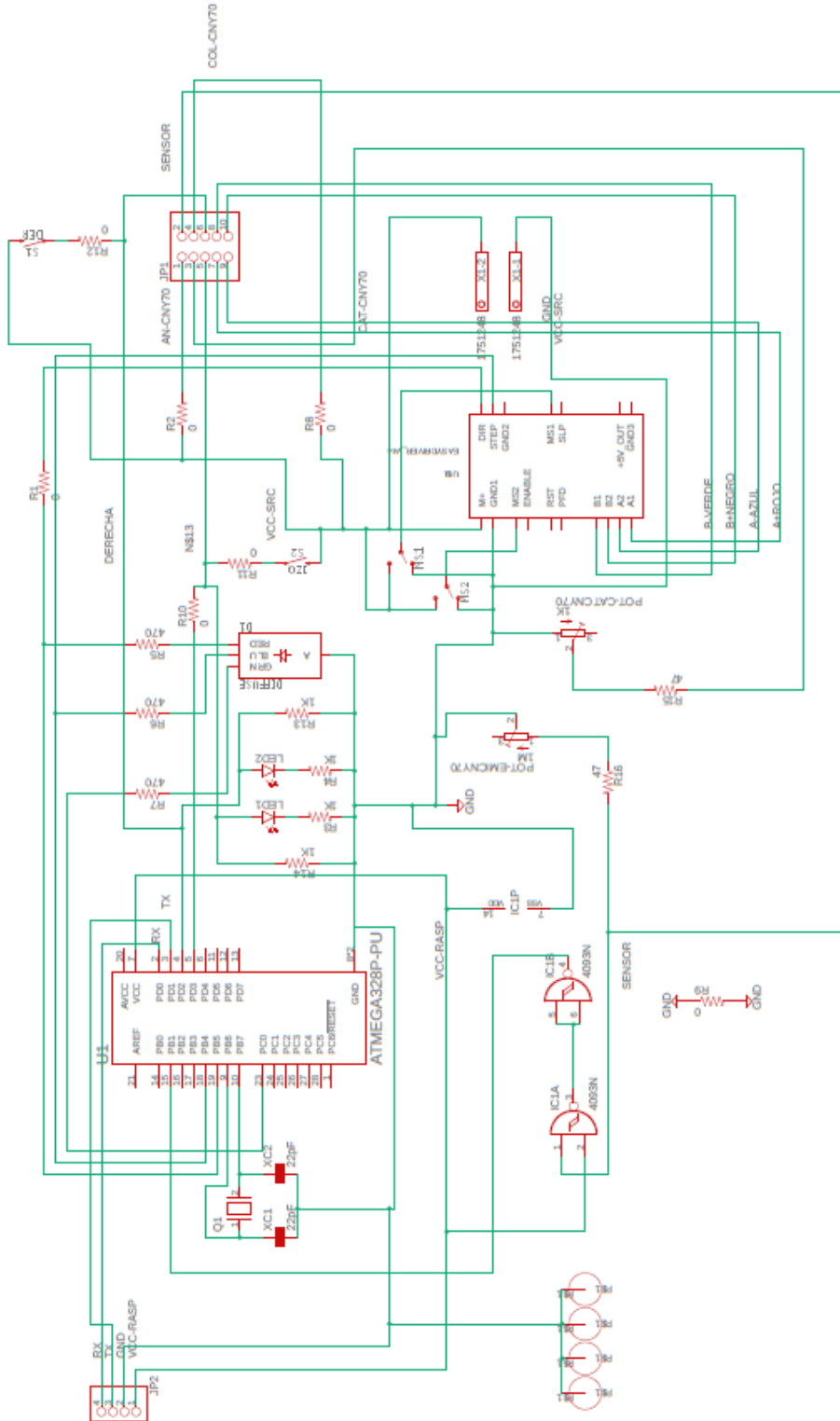


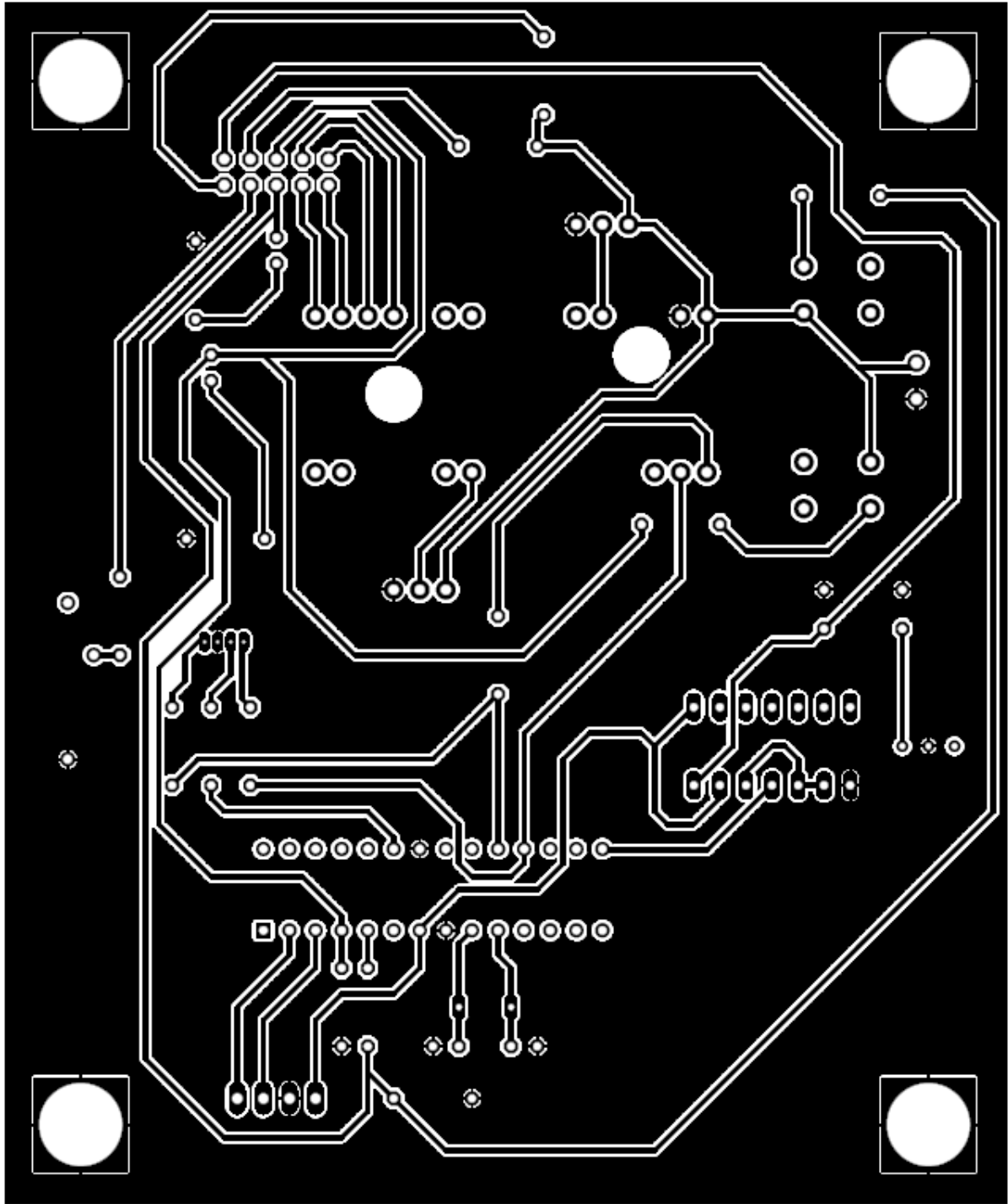
4		3		2		1	
ITEM NO.	PART NUMBER	DESCRIPTION		QTY.			
1	Rodillo	De Nylon		1			
2	Iman	Iman de neodimio 5x3mm		10			
3	Cojinete	Rodamiendo NSK 6801		2			
4	Pasador 1			1			
5	Pasador 2			1			
6	Brazo Inferior			1			
7	Base de Prensa			1			
8	Pasador 3			1			
9	B18.2.4.6M - Heavy hex nut, M12 x 1.75 -- W-C			1			
10	B18.2.4.1M - Hex nut, Style 1, M10 x 1.5, with 16mm WAF --D-C			5			
11	Brazo simple			1			
12	Espaciador			1			

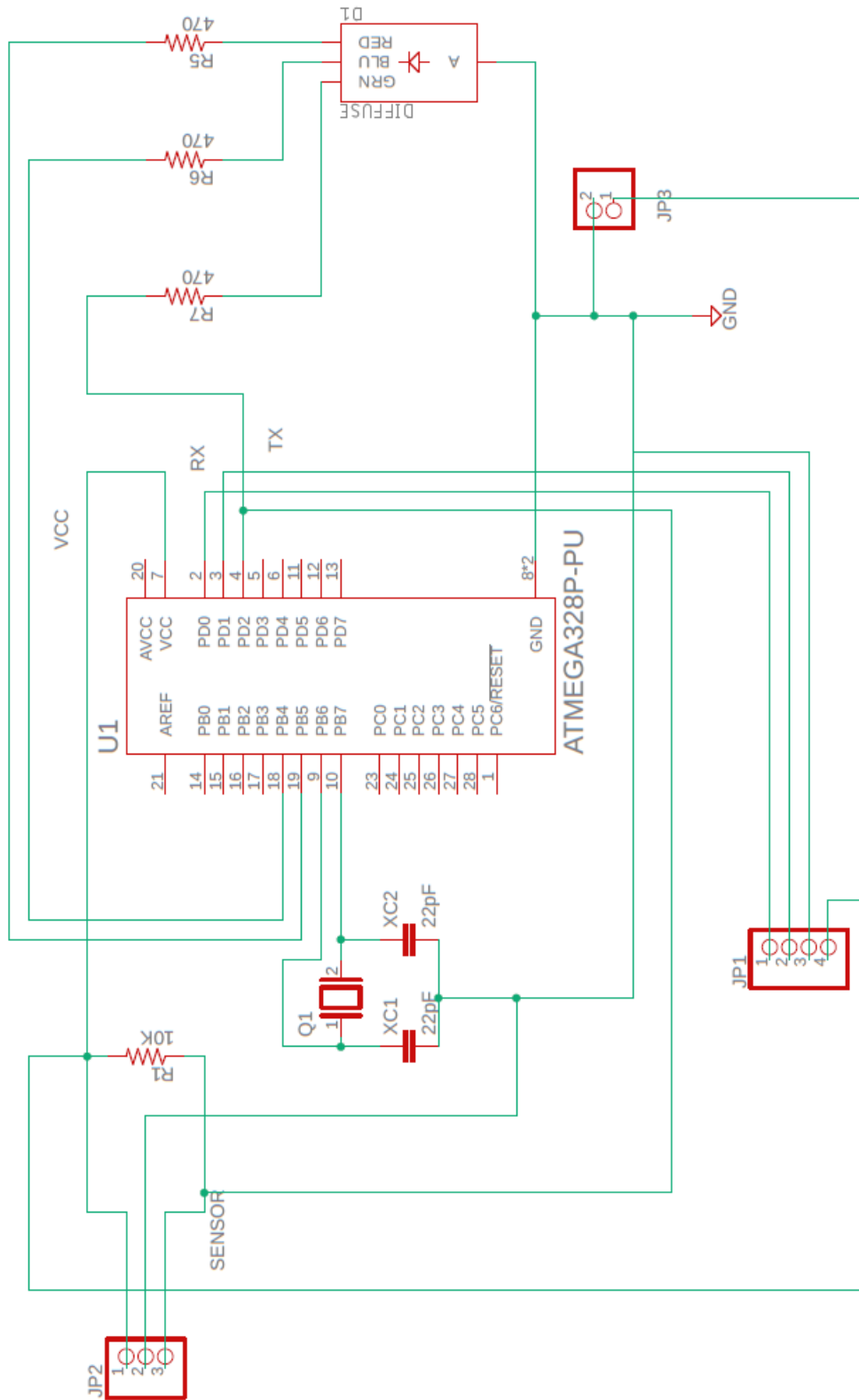
				Polymer S.A.		REVISIÓN:	
				Extrusión			
NOMBRE		FIRMA		FECHA		ACABADO GENERAL:	
DIBUJ. S. Murillo				//		--	
VERIF.				//		MATERIAL:	
APROB.				//			
FABR.				//			
SISTEMA:		TOLERANCIAS:		TRATAMIENTO TÉRMICO		PROYECTO:	
		LINEAL: X ± XX ± XXX ±				Med.Vel. E16	
		ANGULAR: ±°		PESO: gramos		UNIDADES: mm	
						ESCALA:	
						FORMATO A 4	
						HOJA 5 DE 5	

## Apéndice C – Esquemático y PCB del Sistema de Adquisición del Ancho

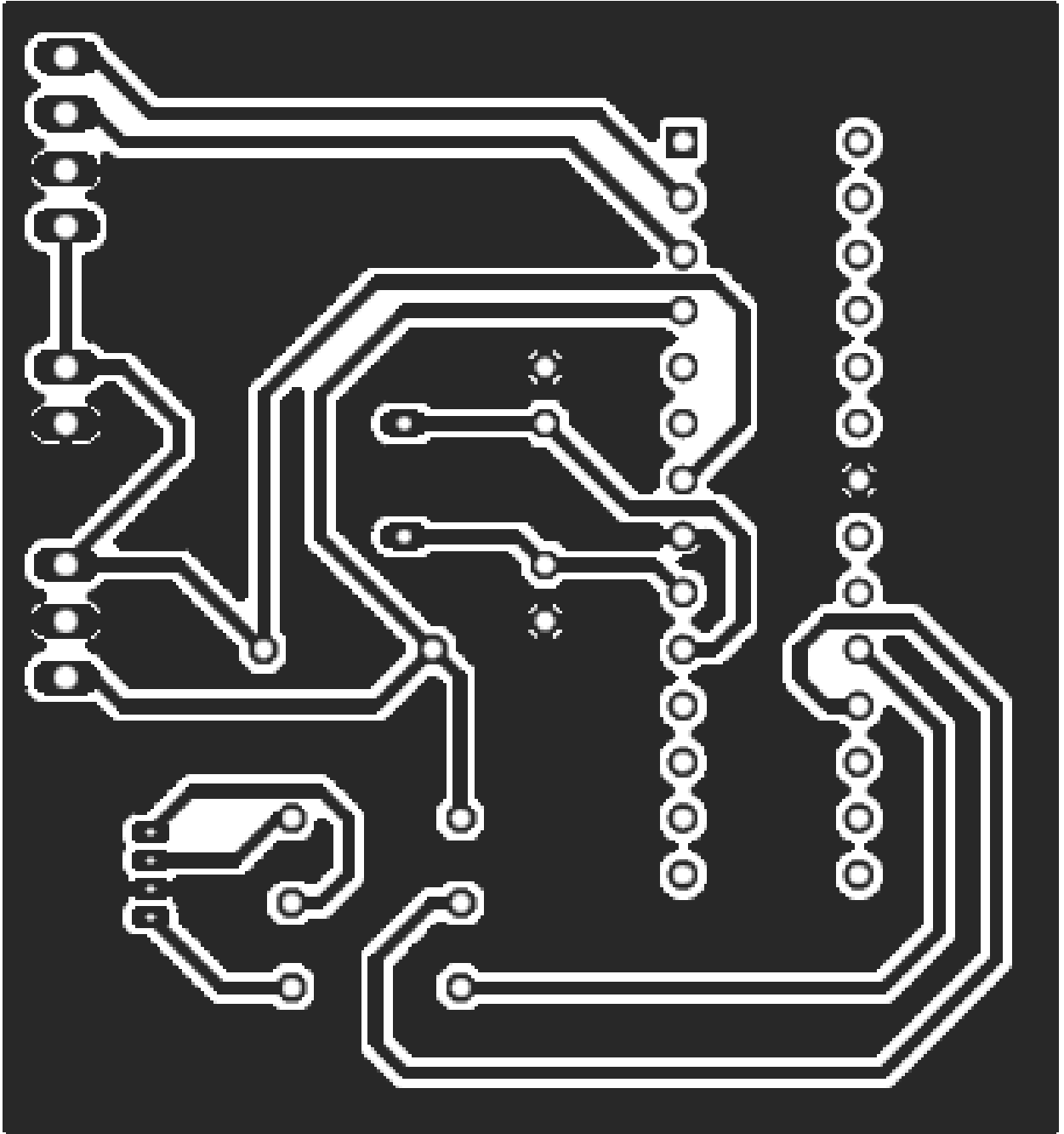




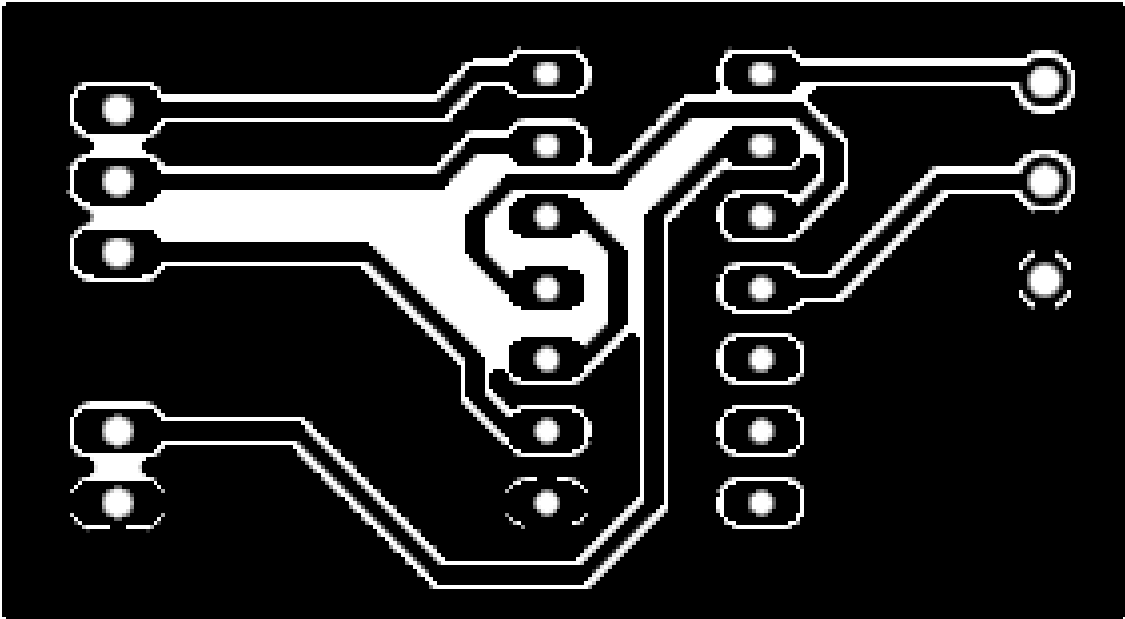
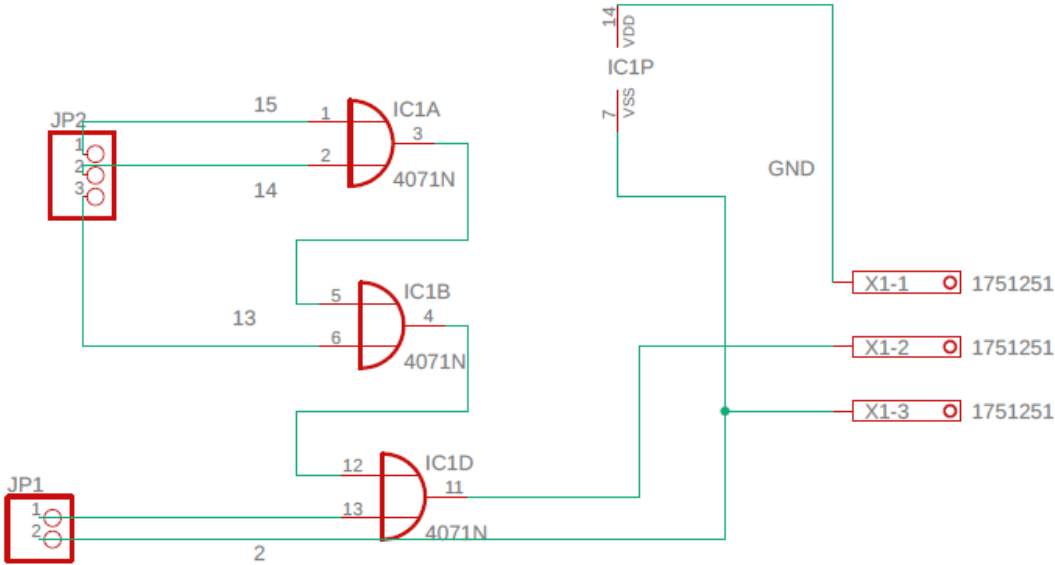
## Apéndice D- Esquemático y PCB del Sistema de Adquisición de Velocidad







Apéndice E – Esquemático y PCB del Sistema de Alarma



## Apéndice F – Código Fuente del Microprocesador del Sistema de Adquisición de Ancho

```

/*
 * main.c
 *
 * Created: 1/25/2021 4:39:54 PM
 * Author: Smurillob
 */
#define F_CPU 16000000
#include <avr/io.h>
#include <avr/interrupt.h>
#include <stdio.h>
#include <stdbool.h>
#include "USART_irq.h"
#include <stdint-gcc.h>
#define BAUD 9600
#define BRC ((F_CPU/16/BAUD) - 1)
int medicion = 0, secuencia = 0, limite =
4000;
volatile uint32_t pasos;
volatile uint32_t cont_global;
int direccion;

static FILE mystdout =
FDEV_SETUP_STREAM(USART0_Transmit_IO,NULL,_FDEV_SETUP_WRITE);

//
ISR(TIMER1_COMPA_vect){ //on duty
    PORTB |= (1<<PORTB4);
    pasos ++;
    cont_global++;
}

ISR(TIMER1_COMPB_vect){ //off duty
    PORTB &= ~(1<<PORTB4);
}
/**/

//
ISR(PCINT0_vect){ //cambio de estado
del sensor reflectivo

    if (PINB & (1 << PINB1)){
        PORTC |=
(1<<0);
        pasos = 0;
        medicion = 1;
    }else if (pasos > 10){
        }else if ((pasos > 50) &&(pasos
< limite)){
            PORTC &= ~
(1<<0);
            PORTB ^= (1
<< DDB5);
            printf("%1X\n", pasos);
            pasos = 0;
            medicion = 0;
        } else if (pasos >= limite){
            PORTC &= ~
(1<<0);
            printf("%1X\n", pasos);
            cont_global =
pasos;
            pasos = 0;
            medicion = 0;
        }
    }

ISR(INT1_vect){ //Detecta el limite de
banda izquierdo

    //cambia sentido de giro hacia
la derecha
    //if(!(PINB & (1 << PINB1))){
        //PORTC |= (1<<DDC1);
        PORTB |= (1<<DDB5);
        cont_global = 0;
    }

ISR(INT0_vect){ //Detecta el limite de
banda derecho

    /*cambia sentido de giro hacia
la izquierda
    if(!(PINB & (1 << PINB1))){*/
        PORTB &= ~(1<<DDB5);
        //PORTC &= ~(1<<DDC1);
        cont_global = 0;
    }

void PWM_Init(){ // PWM para el
movimiento

```

```

        TCCR1A |= (1<<COM1B1)|
(1<<WGM11) | (1<<WGM10);
        TCCR1B |= (1<<CS11) |
(1<<CS10) | (1<<WGM12); // prescalador
a 64 / ctc
        TIMSK1 |= (1<<OCIE1A)|
(1<<OCIE1B);
        OCR1A = 600;
        OCR1B = 200;
        //stepper a 125 rpm
    }
int main(void)
{
    //salidas
    DDRB |= (1<<DDB4);
//Movimiento
    DDRB |= (1<<DDB5);
//Direccion // 1 = se mueve a la derecha
// 0 = se mueve a la izquierda
    DDRC |= (1<<DDC0); //Cuando
mide lo enciende, cuando termina lo
apaga
    DDRC |= (1<<DDC1);

    //entradas
    DDRB &= ~(1<<DDB1); //
PCINT0 / sensor reflectivo
    DDRD &= ~(1<<DDD2); // INT0
/ push button
    DDRD &= ~(1<<DDD3); // INT1
/ push button
//    DDRD &= ~(1<<DDD6); //AIN0
para el sensor
//    DDRD &= ~(1<<DDD7); //AIN1
para referencia
    //Pin change interrupt
    PCICR |= (1<<PCIE0);
    PCMSK0 |= (1<<PCINT1);

```

```

//Ext interrupt
    EICRA |= (1<<ISC01) |
(1<<ISC00) | (1<<ISC10) | (1<<ISC11);
    EIMSK |= (1<<INT0) |
(1<<INT1);

    stdin=stdout=&mystdout;

    UBRR0H = (BRC >> 8);
    UBRR0L = (BRC);

    UCSR0B = (1 << TXEN0) | (1
<< TXCIE0);
    UCSR0C = (1 << UCSZ01) | (1
<< UCSZ00);

    stdout = &mystdout;

    PWM_Init();

    sei();

    //
    while(1){
        //if(cont_global >=
limite && !(PINB & (1 << PINB1))){
            if(cont_global
>= limite && !(PINB & (1 << PINB1))){
                PORTB ^= (1
<< DDB5);

                printf("\n");

                //printf("%u\n", cont_global);
                cont_global =
0;
            }
        }
    }

```

## Apéndice G – Código Fuente del Microprocesador del Sistema de Adquisición de Velocidad

```

#include <avr/io.h> //estado de salidas
#include <avr/interrupt.h> PORTB &= ~(1<<PORTB5);
#include <stdio.h> PORTB &= ~(1<<PORTB4);

#define F_CPU 16000000 DDRD &= ~(1 << DDD2); //
#define BAUD 9600 Clear the PD2 pin
#define BRC ((F_CPU/16/BAUD) - 1) // PD2 (PCINT0 pin) is now an
input
#include "USART_irq.h" PORTD |= (1 << PORTD2); //
turn On the Pull-up
#define LED_ON PORTB |= // PD2 is now an input with pull-
(1<<PORTB4) up enabled
#define LED_OFF PORTB &=
~(1<<PORTB4)
#define LED_Toggle PINB |= (1<<PINB4)
uint8_t cont0, cont1, send, alerta;
long int timer1;
static FILE mystdout = /* seteo del timer */
FDEV_SETUP_STREAM(USART0_Transmit_IO,NULL,_FDEV_SETUP_WRITE);
ISR(INT0_vect){ TIMSK1 |= (1<<TOIE1);
PINB |= (1<<PINB4); TCCR1A = 0b00000000;
PORTB &= ~(1<<PORTB5); TCCR1B = 0b00000101;
timer1 =TCNT1; stdin=stdout=&mystdout;
printf("%1X\n",timer1);
TCNT1H=0; UBRR0H = (BRC >> 8);
TCNT1L=0; UBRR0L = (BRC);
alerta = 0; UCSR0B = (1 << TXEN0) | (1
//cont0 = 0; << TXCIE0);
UCSR0C = (1 << UCSZ01) | (1
<< UCSZ00) ;
} stdout = &mystdout;
ISR(TIMER1_OVF_vect){ sei();
PORTB |= (1<<PORTB5); LED_OFF;
printf("g\n"); // se utiliza g while(1){
porque python recibe hex (0-F)
}
}
int main(void)
{
//salidas
DDRB |= (1<<DDB4);
DDRB |= (1<<DDB5);
}

```

## Apéndice H – Código Python para la Interfaz de Usuario

```
from guizero import*
from tkinter import*
import serial
import threading
import datetime
import requests
from Funciones import*
from time import*
from password import*
from improductivos import*
from variables import*
"FONDO DE LA APP"
fondo = "#FFFFFF"
"TAMAÑO DE LA APP"
t_width = 1920
t_height = 1100
"TAMAÑO DE CELDA"
num_w = 6 #numero de columnas
w_celda = t_width/num_w
num_h = 12 #numero de filas
h_celda = t_height/num_h
"TAMAÑO DE VENTANA DE CONTRASEÑA"
p_width = t_width/3
p_height = t_height/4
"TAMAÑO DE VENTANA DE CONFIGURACION"
c_width = t_width/2
c_height = t_height/3
"TAMAÑO DE VENTANA DE DETENCION"
d_width = t_width/3
d_height = t_height/5
"TAMAÑO DE LETRAS"
letra_leyenda_vertical=32
letra_leyenda_horizontal=36
letra_valor_numerico = 24
letra_valor_texto = 24
letra_especial = 36
"DICCIONARIO DE DATOS A ENVIAR"
data_to_send = {}
"LOG DE ACTIVIDAD"
stream = name("LOG %s" %
datetime.datetime.now().strftime("
%Y-%m-%d_%H-%M-%S"))
"VALORES
PREDETERMINADOS"
#convencion:
etapa_variable_caracteristica
#si no tiene etapa está en la
interfaz
pre_a_ini_set = 100 # cm
pre_a_fin_a_set = 94 # cm
pre_a_fin_b_set = 94 # cm
pre_produ_set = 110 # kg/h
pre_densidad_set = 0.923 #g/cm3
pre_espesor_set = 0.52 #mil/in
pre_a_ini_tol = 1 #cm
pre_a_fin_a_tol = 0.5 #cm
pre_a_fin_b_tol = 0.5 #cm
pre_produ_porcent = 10
#porcentaje
pre_produ_tol =
pre_produ_set*(pre_produ_porcen
t/100)
pre_produ_factor = 1 #se usa en
caso de ser tubo 2 o lamina 1
"INICIALIZACION DE
COMUNICACION SERIAL "
#""
vel_serial
=serial.Serial('/dev/ttyUSB3',
9600)
a_ini_serial=
serial.Serial('/dev/ttyUSB2', 9600)
a_fin_a_serial=
serial.Serial('/dev/ttyUSB1',9600)
a_fin_b_serial=
serial.Serial('/dev/ttyUSB0',9600)
vel_serial.flushInput()
a_ini_serial.flushInput()
a_fin_a_serial.flushInput()
a_fin_b_serial.flushInput()
#""
"VARIABLES DE ALARMA"
vel_alarma = 0
a_ini_alarma = 0
a_fin_a_alarma = 0
a_fin_b_alarma= 0
produ_alarma = 0
desperdicio_alarma = 0
"Funciones de ventanas"
def donothing():
pass
def ventana_contrasena():
pass_window.show()
def ventana_config():
if usuario_input.value ==
usuario and
contrasena_input.value ==
contrasena:
config_window.show()
pass_window.hide()
def actualizar_presentacion():
if (produ_presentacion.value ==
'LAMINA'):
pre_produ_factor = 1
elif (produ_presentacion.value
== 'TUBO'):
pre_produ_factor = 2
print("")
data_to_send["Presentacion"] =
produ_presentacion.value
""
print("velocidad seteada",
vel_set.value)
```

```

    print("factor de produccion:",
pre_produ_factor)

    print("produccion seteada",
produ_set.value)

    print("ancho A
seteado",a_fin_a_set.value)

    print("ancho B
seteado",a_fin_b_set.value)

    print("densidad seteada",
densidad_set.value)

    print("espesor
seteado",espesor_set.value)

print("presentacion",produ_present
acion.value)

"""

def actualizar_paro():

    tipo_paro.value =
razon_detencion.value

    tipo_paro.show()

    tipo_paro.tk.config(bg =
"yellow")

    if (tipo_paro.value != "Paro no
Justificado"):

        warning_box.hide()

        detencion_window.hide()

    else:

        warning_box.show()

    data_to_send["Razon de
Detencion"] =
razon_detencion.value

def valores_set():

    "Factor de Produccion segun la
presentacion del producto"

    #a_ini_set.value =
float(config_a_ini_set.value)

    #a_fin_a_set.value =
float(config_a_fin_a_set.value)

    #a_fin_b_set.value =
float(config_a_fin_b_set.value)

    produ_set.value =
float(config_produ_set.value)

    densidad_set.value =
float(config_densidad_set.value)

```

```

pre_a_ini_tol =
float(config_a_ini_tol.value)

pre_a_fin_a_tol=
float(config_a_fin_a_tol.value)

pre_a_fin_b_tol =
float(config_a_fin_b_tol.value)

pre_produ_porcent=
float(config_produ_tol.value)

#pre_a_ini_set =
float(config_a_ini_set.value)

#pre_a_fin_a_set =
float(config_a_fin_a_set.value)

#pre_a_fin_b_set =
float(config_a_fin_b_set.value)

pre_produ_set =
float(config_produ_set.value)

pre_densidad_set =
float(config_densidad_set.value)

pre_produ_tol =
pre_produ_set*(pre_produ_porcen
t/100)

vel_set.value =
round((100*float(produ_set.value))
/(float(pre_produ_factor)*(float(a_fi
n_a_set.value) +
float(a_fin_b_set.value))*float(den
sidad_set.value)*float(espesor_set
.value)),2)

"""

print("")

print("Tolerancia de
produccion", pre_produ_tol)

print("velocidad seteada",
vel_set.value)

print("factor de produccion:",
pre_produ_factor)

print("produccion seteada",
pre_produ_set)

print("ancho A
seteado",pre_a_fin_a_set)

print("ancho B
seteado",pre_a_fin_b_set)

print("densidad seteada",
pre_densidad_set)

print("espesor
seteado",pre_espesor_set)

```

```

print("presentacion",produ_present
acion.value)

"""

contrasena_input.value = ""

config_window.hide()

#pdb.set_trace()

def cambia_color_tipo_hora():

    if tipo_hora.value ==
"Produccion":

        tipo_hora.tk.config(bg =
"GREEN")

        if tipo_hora.value == "Set Up":

            tipo_hora.tk.config(bg =
"YELLOW")

def time():

    string = strftime("%H:%M:%S
%p")

    reloj.tk.config(text = string)

    reloj.tk.after(1000, time)

"APP"

app = App(title="Extrusora E16",
width = t_width, height = t_height,
layout = "grid")

app.bg = fondo

app.tk.config(cursor= "target")

app.tk.attributes("-fullscreen",
True)

#app.tk.protocol("WM_DELETE_
WINDOW", donothing)

"Contraseña"

pass_window = Window(app, title
= "Autorizacion", width =
int(p_width), height =
int(p_height))

pass_window.hide()

pass_window.tk.config(relief=
"sunken")

pass_window.tk.config(cursor=
"pirate")

pass_window.tk.attributes('-
topmost', True)

```

```

Text(pass_window, text
="Configuración de tolerancia
requiere contraseña")

usuario_input =
TextBox(pass_window, text=
"Usuario")

contrasena_input =
TextBox(pass_window, hide_text
= True)

intro_pass_button =
PushButton(pass_window,
command= ventana_config, text=
"Aceptar")

"Configuracion"

config_window = Window(app,
title= "Configuracion", width =
int(c_width), height = int(c_height),
layout= "grid")

config_window.tk.attributes('-
topmost', True)

config_window.hide()

tol_ini_name =
Box(config_window, grid = [0,0],
width = int(2*c_width/3), height =
int(c_height/7))

Text(tol_ini_name, text =
"Tolerancia de Ancho Inicial
(cm):", align = "left")

tol_ini_box = Box(config_window,
grid = [1,0], width = int(c_width/3),
height = int(c_height/7))

config_a_ini_tol =
TextBox(tol_ini_box, text = "",
align = "left")

config_a_ini_tol.value =
pre_a_ini_tol

tol_a_name = Box(config_window,
grid = [0,1], width =
int(2*c_width/3), height =
int(c_height/7))

Text(tol_a_name, text =
"Tolerancia de Ancho A (cm):",
align = "left")

tol_a_box = Box(config_window,
grid = [1,1], width = int(c_width/3),
height = int(c_height/7))

config_a_fin_a_tol =
TextBox(tol_a_box, text = "", align
= "left")

config_a_fin_a_tol.value =
pre_a_fin_a_tol

tol_b_name = Box(config_window,
grid = [0,2], width =
int(2*c_width/3), height =
int(c_height/7))

Text(tol_b_name, text =
"Tolerancia de Ancho B (cm):",
align = "left")

tol_b_box = Box(config_window,
grid = [1,2], width = int(c_width/3),
height = int(c_height/7))

config_a_fin_b_tol =
TextBox(tol_b_box, text = "", align
= "left")

config_a_fin_b_tol.value =
pre_a_fin_b_tol

tol_produ_name =
Box(config_window, grid = [0,3],
width = int(2*c_width/3), height =
int(c_height/7))

Text(tol_produ_name, text =
"Tolerancia de Produccion (%):",
align = "left")

tol_produ_box =
Box(config_window, grid = [1,3],
width = int(c_width/3), height =
int(c_height/7))

config_produ_tol =
TextBox(tol_produ_box, text = "",
align = "left")

config_produ_tol.value =
pre_produ_porcent

set_produ_name =
Box(config_window, grid = [0,4],
width = int(2*c_width/3), height =
int(c_height/7))

Text(set_produ_name, text =
"Produccion Set (kg/h):", align =
"left")

set_produ_box =
Box(config_window, grid = [1,4],
width = int(c_width/3), height =
int(c_height/7))

config_produ_set =
TextBox(set_produ_box, text = "",
align = "left")

config_produ_set.value =
pre_produ_set

set_densidad_name =
Box(config_window, grid = [0,5],
width = int(2*c_width/3), height =
int(c_height/7))

Text(set_densidad_name, text =
"Densidad (g/cm3):", align = "left")

set_densidad_box =
Box(config_window, grid = [1,5],
width = int(c_width/3), height =
int(c_height/7))

config_densidad_set=
TextBox(set_densidad_box, text =
"", align = "left")

config_densidad_set.value =
pre_densidad_set

PushButton(config_window, grid=
[1,6], text = "Cambiar", command
= valores_set, )

"-----"
"-----"
"-----"
"-----"

"Detencion"

detencion_window = Window(app,
title= "PARO DE LA MAQUINA",
width = int(d_width), height =
int(d_height))

razon_detencion =
Combo(detencion_window,
options = improductivos)

razon_detencion.text_size = 11

Text(detencion_window, text =
"Seleccione la razon por la que se
detuvo", size = 40).tk.config(bg =
"red")

Box(detencion_window, width =
'fill', height = int(d_height/4),
border = False, align =
'bottom').tk.config(bg = "red")

warning_box =
Box(detencion_window, align =
'bottom')

warning = Text(warning_box, text
= "Debe seleccionar una opción
distinta!", size = 35, align =
'bottom').tk.config(bg = "red", fg =
"yellow")

warning_box.hide()

Box(detencion_window, width =
'fill', height = int(d_height/4),
border = False, align =
'bottom').tk.config(bg = "red")

```



```

detencion_button=PushButton(det
encion_window, text="Registrar",
command = actualizar_paro, align
= 'bottom')

detencion_button.text_size = 40

detencion_button.tk.config(bg =
"yellow")

detencion_window.tk.protocol("W
M_DELETE_WINDOW",
do nothing)

detencion_window.tk.attributes("-
fullscreen", True)

detencion_window.tk.attributes('-
topmost', True)

detencion_window.tk.config(bg =
"red")

detencion_window.hide()

"-----"
"-----"
"-----"
"-----"
""
#Titulo

title= Box(app, width = 'fill' , grid =
[0,0], border = False)

Text(title, text= "EXTRUSORA
E16", color = "#D7263D", align =
"top").tk.config(font='Helvetica 24
bold')

"-----"
"-----"
"-----"
"-----"

#Sub Tablas

##

#Tabla Superior

##

tabla_sup= Box(app, width =
t_width, height = int(h_celda),
border = True, grid = [0,1])

""

empty_sup= Box(tabla_sup,
width= int(w_celda), height =

```

```

int(h_celda), align= "left", border
=True)

img = PhotoImage(file =
"poly2.gif")

pass_button =
Button(empty_sup.tk, command =
ventana_contrasena, text=
"Config", image = img, cursor =
"heart").pack()

#empty_sup.tk.config(cursor =
"cross")

"Variables"

var= Box(tabla_sup, width=
int(w_celda), height = 55, align=
"left", border =True)

Text(var, text= "VARIABLES",
align =
"top").tk.config(font='Helvetica 28
bold')

"Teorico/Esperado"

tees= Box(tabla_sup, width=
int(w_celda), height = 55, align=
"left", border =True)

Text(tees, text= "SETEADO", align
= "top", color =
"#DC965A").tk.config(font='Helveti
ca 28 bold')

"Medido"

med= Box(tabla_sup, width=
int(w_celda), height = 55, align=
"left", border =True)

Text(med, text= "REAL", align =
"top", color =
"#06AED5").tk.config(font='Helveti
ca 28 bold')

"%Error"

error= Box(tabla_sup, width=
int(w_celda), height = 55, align=
"left", border =True)

Text(error, text= "DIFERENCIA",
align =
"top").tk.config(font='Helvetica 28
bold')

"Alerta"

alerta= Box(tabla_sup, width=
int(w_celda), height = 55, align=
"left", border =True)

Text(alerta, text= "ALERTA", align
= "top").tk.config(font='Helvetica
28 bold')

```

```

"-----"
"-----"
"-----"
""
##
#Tabla Variables Controladas
##
""
"-----"
"-----"
"-----"
var_cont= Box(app, width =
t_width, height = 4*int(h_celda),
border = True, grid = [0,2])

"Texto variables controladas"

textvar_cont= Box(var_cont, width
= int(w_celda), height =
int(4*h_celda), border = True,
align = "left")

Text(textvar_cont, text =
"CONTROLADAS", align =
"left").tk.config(font='Helvetica 24
bold')

""
#Caja De Nombres de Variables
Controladas
""

nomb_cont= Box(var_cont, width =
int(w_celda), height =
int(4*h_celda), border = True,
align = "left")

"Ancho Inicial"

anchoini= Box(nomb_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

Text(anchoini, text = "ANCHO
INICIAL (cm)", align =
"left").tk.config(font='Helvetica 22
bold')

"Ancho Final A"

anchofina= Box(nomb_cont, width
= int(w_celda), height =
int(h_celda), border = True, align =
"top")

Text(anchofina, text = "ANCHO
FINAL A (cm)", align =
"left").tk.config(font='Helvetica 22
bold')

```

```

"Ancho Inicial"

anchofinb= Box(nomb_cont, width
= int(w_celda), height =
int(h_celda), border = True, align =
"top")

Text(anchofinb, text = "ANCHO
FINAL B (cm)", align =
"left").tk.config(font='Helvetica 22
bold')

"Velocidad"

vel= Box(nomb_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

Text(vel, text = "VELOCIDAD
(m/min)", align =
"left").tk.config(font='Helvetica 22
bold')

"-----"
"-----"
"-----"

""

#Caja de Inputs para Variables
Controladas

""

"-----"
"-----"
"-----"

input_cont= Box(var_cont, width =
int(w_celda), height =
int(4*h_celda), border = True,
align = "left")

"Input Ancho Inicial"

input_ancho_ini_box=
Box(input_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

a_ini_set=
TextBox(input_ancho_ini_box,
text= "(cm)", width = "fill", height =
"fill")

a_ini_set.text_size = ("44")

a_ini_set.tk.config(justify =
"center")

a_ini_set.tk.config(cursor="exchan
ge")

a_ini_set.value = pre_a_ini_set

"Input Ancho Final A"

input_ancho_fin_a_box=
Box(input_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

a_fin_a_set=
TextBox(input_ancho_fin_a_box,
text= "(cm)", width = "fill", height =
"fill")

a_fin_a_set.text_size = ("44")

a_fin_a_set.tk.config(justify =
"center")

a_fin_a_set.tk.config(cursor="exch
ange")

a_fin_a_set.value =
pre_a_fin_a_set

"Input Ancho Final A"

input_ancho_fin_b_box=
Box(input_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

a_fin_b_set=
TextBox(input_ancho_fin_b_box,
text= "(cm)", width = "fill", height =
"fill")

a_fin_b_set.text_size = ("44")

a_fin_b_set.tk.config(justify =
"center")

a_fin_b_set.tk.config(cursor="exch
ange")

a_fin_b_set.value =
pre_a_fin_b_set

"Velocidad"

vel_set_box= Box(input_cont,
width = int(w_celda), height =
int(h_celda), border = True, align =
"top")

vel_set = Text(vel_set_box, text =
"(m/min)", align = "left", width =
"fill", height = "fill")

vel_set.text_size = ("44")

"-----"
"-----"
"-----"

""

#Caja de Outputs para Variables
Controladas

""

"-----"
"-----"
"-----"

output_cont= Box(var_cont, width
= int(w_celda), height =
int(4*h_celda), border = True,
align = "left")

"Ancho Inicial"

ancho_ini_output_box=
Box(output_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

a_ini_real=
Text(ancho_ini_output_box, text =
"***", align = "top", size = 50)

"Ancho Final A"

ancho_fin_a_output_box=
Box(output_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

a_fin_a_real=
Text(ancho_fin_a_output_box, text
= "***", align = "top", size = 50)

"Ancho Inicial"

ancho_fin_b_output_box=
Box(output_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

a_fin_b_real=
Text(ancho_fin_b_output_box, text
= "***", align = "top", size = 50)

"Velocidad"

vel_output_box= Box(output_cont,
width = int(w_celda), height =
int(h_celda), border = True, align =
"top")

vel_real= Text(vel_output_box,
text = "***", align = "top", size = 50)

"-----"
"-----"
"-----"

""

#Caja de Error para Variables
Controladas

""

"-----"
"-----"
"-----"

```

```
error_cont= Box(var_cont, width =
int(w_celda), height =
int(4*h_celda), border = True,
align = "left")
```

"Ancho Inicial"

```
ancho_ini_error_box=
Box(error_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")
```

```
a_ini_error=
Text(ancho_ini_error_box, text =
"****", align = "top", size = 50)
```

"Ancho Final A"

```
a_fin_a_error_box=
Box(error_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")
```

```
a_fin_a_error=
Text(a_fin_a_error_box, text =
"****", align = "top", size = 50)
```

"Ancho Inicial"

```
a_fin_b_error_box=
Box(error_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")
```

```
a_fin_b_error=Text(a_fin_b_error_
box, text = "****", align = "top", size
= 50)
```

"Velocidad"

```
vel_error_box= Box(error_cont,
width = int(w_celda), height =
int(h_celda), border = True, align =
"top")
```

```
vel_error= Text(vel_error_box, text
= "****", align = "top", size = 50)
```

```
"-----"
"-----"
```

""

#Caja de Alerta para Variables Controladas

""

```
"-----"
"-----"
```

```
alarma_cont= Box(var_cont, width
= int(w_celda), height =
int(4*h_celda), border = True,
align = "left")
```

"Ancho Inicial"

```
ancho_ini_alarma_box =
Box(alarma_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")
```

```
a_ini_alarma_activa =
Drawing(ancho_ini_alarma_box,
height=int(h_celda), width="fill",
align = "top")
```

```
a_ini_alarma_activa.rectangle(0,
0, int(w_celda), int(h_celda),
color="red")
```

```
a_ini_alarma_activa.visible =
False
```

"Ancho Final A"

```
ancho_fin_a_alarma_box =
Box(alarma_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")
```

```
a_fin_a_alarma_activa =
Drawing(ancho_fin_a_alarma_box
, height=int(h_celda), width="fill",
align = "top")
```

```
a_fin_a_alarma_activa.rectangle(0
, 0, int(w_celda), int(h_celda),
color="red")
```

```
a_fin_a_alarma_activa.visible =
False
```

"Ancho Inicial"

```
ancho_fin_b_alarma_box =
Box(alarma_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")
```

```
a_fin_b_alarma_activa =
Drawing(ancho_fin_b_alarma_box
, height=int(h_celda), width="fill",
align = "top")
```

```
a_fin_b_alarma_activa.rectangle(0
, 0, int(w_celda), int(h_celda),
color="red")
```

```
a_fin_b_alarma_activa.visible =
False
```

"Velocidad"

```
vel_alarma_box =
Box(alarma_cont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")
```

```
vel_alarma_activa =
Drawing(vel_alarma_box,
height=int(h_celda), width="fill",
align = "top")
```

```
vel_alarma_activa.rectangle(0, 0,
int(w_celda), int(h_celda),
color="red")
```

```
vel_alarma_activa.visible = False
```

```
"-----"
"-----"
```

##

#Tabla Variables No Controladas y Desperdicio

##

```
"-----"
"-----"
```

```
var_ncont= Box(app, width =
t_width, height = int(2*h_celda),
border = True, grid = [0,3])
```

"Texto variables no controladas"

```
textvar_ncont= Box(var_ncont,
width = int(w_celda), height =
int(4*h_celda), border = True,
align = "left")
```

```
Text(textvar_ncont, text = "NO
CONTROLADAS", align =
"left").tk.config(font='Helvetica 24
bold')
```

```
"-----"
"-----"
```

```
"-----"
"-----"
```

```
nombre_ncont= Box(var_ncont,
width = int(w_celda), height =
int(2*h_celda), border = True,
align = "left")
```

"Espesor"

```
nombre_espesor=
Box(nombre_ncont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")
```

```
Text(nombre_espesor, text =
"ESPESOR (mil/in)", align = "left",
size = 22)
```

"Densidad"

```
nombre_densidad=
Box(nombre_ncont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")
```

```

Text(nombre_densidad, text =
"DENSIDAD (g/cm3)", align =
"left", size = 22)

"-----"
"-----"
-----"
""

#Caja de Inputs para Variables No
Controladas

""

"-----"
-----"

input_ncont= Box(var_ncont, width
= int(w_celda), height =
int(4*h_celda), border = True,
align = "left")

"Input Espesor"

input_espesor_box=
Box(input_ncont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

espesor_set=
TextBox(input_espesor_box, text=
"Digite Espesor (mil/in)", width =
"fill", height = "fill")

espesor_set.value =
pre_espesor_set

espesor_set.text_size = ("44")

espesor_set.tk.config(justify =
"center")

espesor_set.tk.config(cursor="exc
hange")

"Input Densidad"

input_densidad_box=
Box(input_ncont, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

densidad_set=
Text(input_densidad_box, text=
"(g/cm3)", width = "fill", height =
"fill")

densidad_set.value =
pre_densidad_set

densidad_set.text_size = ("44")

densidad_set.tk.config(justify =
"center")

#densidad_set.tk.config(cursor="e
xchange")

"-----"
-----"
-----"
""

#Leyenda de Desperdicio

""

nombre_desperdicio=
Box(var_ncont, width =
int(w_celda), height =
int(2*h_celda), border = True,
align = "left")

nombre_desperdicio.tk.config(bg =
"red")

Box(nombre_desperdicio, width =
int(w_celda), height =
int(2*h_celda/4), border = False,
align = "top").tk.config(bg = "red")

Text(nombre_desperdicio, text=
"DESPERDICIO", align= "top",
color= "#FFFFFF").tk.config(bg =
"red", font='Helvetica 28 bold')

Text(nombre_desperdicio, text=
"AUTOMATICO", align= "top",
color= "#FFFFFF").tk.config(bg =
"red", font='Helvetica 28 bold')

"-----"
-----"
-----"
""

#Caja para Desperdicio Real

""

desperdicio_real_box=
Box(var_ncont, width =
int(w_celda), height =
int(2*h_celda), border = True,
align = "left")

nombre_desperdicio_real=
Box(desperdicio_real_box, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

Text(nombre_desperdicio_real,
text= "REAL", align= "top", color=
"#06AED5").tk.config(font='Helvetic
a 24 bold')

#Text(nombre_produ_esperado,
text= "ESPERADO", align= "top",
color=
"#008000").tk.config(font='Helvetic
a 24 bold')

desperdicio_real_inner_box=
Box(desperdicio_real_box, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

desperdicio_real=
Text(desperdicio_real_inner_box,
text= "" (cm)", align= "top", color =
"#06AED5")

desperdicio_real.tk.config(font='He
lvetica 44 bold')

"#008000").tk.config(font='Helvetic
a 24 bold')

desperdicio_esperado_inner_box=
Box(desperdicio_esperado_box,
width = int(w_celda), height =
int(h_celda), border = True, align =
"top")

desperdicio_esperado =
Text(desperdicio_esperado_inner_
box, text= "" (cm)", align= "top",
color= "#008000")

desperdicio_esperado.tk.config(fo
nt='Helvetica 44 bold')

"-----"
-----"
-----"
-----"
""

#Caja para Desperdicio Esperado

""

desperdicio_esperado_box=
Box(var_ncont, width =
int(w_celda), height =
int(2*h_celda), border = True,
align = "left")

nombre_desperdicio_esperado=
Box(desperdicio_esperado_box,
width = int(w_celda), height =
int(h_celda), border = True, align =
"top")

Text(nombre_desperdicio_espera
do, text= "ESPERADO", align=
"top", color=

```

```

"-----"
-----"

##

#Espacio en Blanco

##

"-----"
-----"

tabla_blanco= Box(app, width =
int(t_width), height = int(h_celda),
border = False, grid = [0,4])

select_tipo_nombre_box=
Box(tabla_blanco, width =
int(w_celda), height = 'fill', border
=False, align = 'left')

Text(select_tipo_nombre_box, text
= "PRODUCTO", align=
'left').tk.config(font='Helvetica 24
bold')

select_tipo_box= Box(tabla_blanco,
width = int(2*w_celda), height =
'fill', border =True, align = 'left')

produ_tipo= TextBox
(select_tipo_box, text= "DIGITE
DATO", width = 'fill', height = 'fill')

produ_tipo.tk.config(cursor="excha
nge")

produ_tipo.text_size = "24"

select_presentacion_box =
Box(tabla_blanco, width =
int(3*w_celda), height = 'fill',
border =False, align = 'right')

select_presentacion_box.tk.config(
cursor="exchange")

""

presentacion_produ_legenda_box
= Box(select_presentacion_box,
width = int(((3*w_celda))/2), height
= "fill", border =False, align = "left")

presentacion_produ_legenda =
Text(presentacion_produ_legenda
_box, text = "Selección", size =
"25", align = "right")

""

produ_presentacion =
ButtonGroup(select_presentacion_
_box, options = ["LAMINA",
"TUBO"], selected = "top", align =
"left", command =
actualizar_presentacion, horizontal
= True)

produ_presentacion.text_size =
"48"

cont_button = 0

for radio_button in
produ_presentacion.children:

    tk_radio_button =
radio_button.tk

    if cont_button == 0:

        radio_button.tk.config(bg =
"#06AED5")

        elif cont_button == 1:

            radio_button.tk.config(bg =
"#808080")

            cont_button +=1

"-----"
-----"

##

#Tabla Operador y Produccion

##

"-----"
-----"

op_prod_box= Box(app, width =
t_width, height = int(2*h_celda),
border = True, grid = [0,5])

"Texto Operador"

text_op= Box(op_prod_box, width
= int(w_celda), height =
int(4*h_celda), border = True,
align = "left")

Text(text_op, text =
"OPERADOR", align =
"left").tk.config(font='Helvetica 24
bold')

"-----"
-----"

"-----"
-----"

""

#Caja para Leyendas de Operador

""

leyenda_operador_box=
Box(op_prod_box, width =
int(w_celda), height =
int(2*h_celda), border = True,
align = "left")

leyenda_nombre=
Box(leyenda_operador_box, width
= int(w_celda), height =
int(h_celda), border = True, align =
"top")

Text(leyenda_nombre, text=
"Nombre:", align= "left", size = 24)

leyenda_numero=
Box(leyenda_operador_box, width
= int(w_celda), height =
int(h_celda), border = True, align =
"top")

Text(leyenda_numero, text=
"Numero:", align= "left", size = 24)

"-----"
-----"

"-----"
-----"

""

#Caja para Datos de Operador

""

datos_operador_box =
Box(op_prod_box, width =
int(w_celda), height =
int(2*h_celda), border = True,
align = "left")

nombre_operador_box =
Box(datos_operador_box, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

nombre_operador =
TextBox(nombre_operador_box,
text= "", width = "fill", height =
"fill")

nombre_operador.tk.config(cursor
="man")

nombre_operador.text_size =
("24")

nombre_operador.tk.config(justify
= "center")

numero_operador_box =
Box(datos_operador_box, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

```

```

numero_operador =
TextBox(numero_operador_box,
text= "***", width = "fill", height =
"fill")

numero_operador.tk.config(cursor
="man")

numero_operador.text_size =
("24")

numero_operador.tk.config(justify
= "center")

"-----"
"-----"

"-----"
"-----"

""

#Leyenda de Produccion
""

nombre_produ=
Box(op_prod_box, width =
int(w_celda), height =
int(2*h_celda), border = True,
align = "left")

Text(nombre_produ, text= "kg/h",
align="top", size = 90, color=
"#E6AF2E")

"-----"
"-----"

"-----"
"-----"

""

#Caja para Produccion Esperado
""

produ_esperado_box=
Box(op_prod_box, width =
int(w_celda), height =
int(2*h_celda), border = True,
align = "left")

nombre_produ_esperado=
Box(produ_esperado_box, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

Text(nombre_produ_esperado,
text= "ESPERADO", align= "top",
color=

"#008000").tk.config(font='Helvetic
a 24 bold')

produ_set_box=
Box(produ_esperado_box, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

produ_set= Text(produ_set_box,
text= "***", align= "top", color =
"#008000")

produ_set.tk.config(font='Helvetica
44 bold')

produ_set.value = pre_produ_set

"-----"
"-----"

"-----"
"-----"

""

#Caja para Produccion Real
""

produ_real_box=
Box(op_prod_box, width =
int(w_celda), height =
int(2*h_celda), border = True,
align = "left")

nombre_produ_real=
Box(produ_real_box, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

Text(nombre_produ_real, text=
"REAL", align= "top", color =
"#06AED5").tk.config(font='Helveti
ca 24 bold')

produ_calc_box=
Box(produ_real_box, width =
int(w_celda), height = int(h_celda),
border = True, align = "top")

produ_real= Text(produ_calc_box,
text= "*** (kg/h)", align= "top", color
= "#06AED5")

produ_real.tk.config(font='Helvetic
a 44 bold')

"-----"
"-----"

"-----"
"-----"

#
#Informacion Adicional
#

"-----"
"-----"

info_box = Box(app, width=
int(t_width), height = int(h_celda),
border = True, grid = [0,6])

"-----"
"-----"

tipo_hora_leyenda_box = Box
(info_box, width= int(3*w_celda/2),
height= 'fill', border = False, align
= "left")

Text(tipo_hora_leyenda_box, text
= "TIPO DE HORA:", size = "22")

tipo_hora_box = Box (info_box,
width = int(w_celda), height = 'fill',
border = False, align = "left")

tipo_hora =
Combo(tipo_hora_box, options =
["Set Up", "Produccion"])

tipo_hora.tk.config(bg = "GREEN")

#tipo_hora.tkinter.OptionMenu.con
fig(command =
cambia_color_tipo_hora)

tipo_hora.text_size = 20

"-----"
"-----"

tipo_paro_leyenda_box = Box
(info_box, width= int(w_celda),
height= 'fill', border = False, align
= "left")

Text(tipo_paro_leyenda_box, text
= "RAZON DE PARO:", size =
"22")

#""

tipo_paro_box = Box (info_box,
width = 'fill', height = 'fill', border =
False, align = "left")

tipo_paro = Text(tipo_paro_box,
text = "")

tipo_paro.text_size = 22

tipo_paro.hide()

#""

```

```

reloj_outter_box = Box (info_box,
width = int(w_celda), height = 'fill',
border = False, align = "right")

reloj_inner_box =
Box(reloj_outter_box, width =
int(w_celda), height =
int(h_celda/2), border = True, align
= "bottom")

reloj = Text(reloj_inner_box, text =
"")

reloj.text_size = 22

time()

vel_set.value =
round((100*float(produ_set.value)
/(1.524*(float(a_fin_a_set.value) +
float(a_fin_b_set.value))*float(den
sidad_set.value)*float(espesor_set
.value))),2)

"-----"
"-----"

#

#Threads

#

"-----"
"-----"

""

#""

threading.Thread(target =
velocidad, args =
(produ_presentacion,
pre_produ_factor, vel_real,
vel_set, vel_serial, vel_error,
data_to_send, vel_alarma_activa,
tipo_paro, detencion_window,
tipo_hora, config_produ_tol,
razon_detencion, stream,)).start()

threading.Thread(target =
ancho_ini, args = (a_ini_real,

```

```

a_ini_serial, a_ini_set, a_ini_error,
a_ini_alarma_activa,
config_a_ini_tol, data_to_send,
stream,)).start()

threading.Thread(target =
ancho_fin_a, args = (a_fin_a_real,
a_fin_a_serial, a_fin_a_set,
a_fin_a_error,
a_fin_a_alarma_activa,
config_a_fin_a_tol, data_to_send,
stream,)).start()

threading.Thread(target =
ancho_fin_b, args = (a_fin_b_real,
a_fin_b_serial, a_fin_b_set,
a_fin_b_error,
a_fin_b_alarma_activa,
config_a_fin_b_tol, data_to_send,
stream,)).start()

threading.Thread(target =
envio_datos, args = (produ_tipo,
nombre_operador,
numero_operador, data_to_send,
variables, stream, tipo_hora,
)).start()

threading.Thread(target =
temp_setup, args = (tipo_hora,
detencion_window,
nombre_operador,
numero_operador, stream)).start()

threading.Thread(target =
calculo_produccion, args
=(produ_set, produ_real,
a_fin_a_set, a_fin_b_set, vel_set,
a_fin_a_real, a_fin_b_real,
produ_presentacion, vel_real,
densidad_set, espesor_set,
produ_calc_box, config_produ_tol,
data_to_send)).start()

threading.Thread(target =
calculo_desperdicio, args
=(a_ini_set, a_fin_a_set,
a_fin_b_set, a_ini_real,
a_fin_a_real, a_fin_b_real,
desperdicio_esperado,
desperdicio_real,

```

```

data_to_send, config_a_ini_tol,
config_a_fin_a_tol,
config_a_fin_b_tol,
produ_presentacion,
desperdicio_real_inner_box)).start
()

#""

app.display()

pdb.set_trace()

```

## Apéndice I – Código Python de la Biblioteca Funciones

```

from guizero import*
from time import*
from gpiozero import LED
import serial
import threading
import datetime
import requests
import pdb

led_1 = LED(2)
led_2 = LED(3)
led_3 = LED(14)
led_4 = LED(15)
led_1.off()
led_2.off()
led_3.off()
led_4.off()

"""VELOCIDAD"""

def velocidad(produ_presentacion,
pre_produ_factor,
vel_real,vel_set,vel_serial,vel_erro
r, data_to_send,

        vel_alarma_activa,
tipo_paro, detencion_window,
tipo_hora, config_produ_tol,
razon_detencion, stream):

    cont0 = 0

    cont1 = 0

    radio = 4

    promedio = 0

    cont_alarma = 0

    while True:

        file = open(stream, 'a+')

        vel_Bytes =
vel_serial.readline()

        vel_line =
vel_Bytes.decode('utf-8').strip()

        if (produ_presentacion.value
== "LAMINA"):

            pre_produ_factor = 1;

            elif
(produ_presentacion.value ==
"TUBO"):

                pre_produ_factor = 2;

            if vel_line != 'g':

                if cont0 == 0:

                    tipo_hora.value = "Set
Up"

                    cont0 +=1

                    detencion_window.hide()

                    tipo_paro.hide()

                    tipo_hora.show()

                    data_to_send["Razon de
Paro"] = ""

                    data_to_send["Tipo Hora"]
= tipo_hora.value

                    cont_alarma = 0

                    #vel_alarma_activa.visible
= False

                    #led_1.off()

                    vel_line = int(vel_line,16)

                    tiempo = vel_line/15625

                    try:

                        velocidad =
((2*3.1416*radio*60)/(100*5*tiemp
o))

                    except:

                        cont1 -=1

                        file.write("La VELOCIDAD
es de %s m/min. \n" %
str(velocidad))

                        #print('La velocidad es de
',velocidad, ' m/min.')
```

```

                promedio =
round(promedio,2)

                except:

                    ""

                    vel_tol =
float(vel_set.value)*(float(config_pr
odu_tol.value)/100)

                    if cont1 == 5:

                        file.write("El promedio
de VELOCIDAD de los ultimos %s
conteos es de " % str(cont1))

                        file.write("%s m/min. \n"
% str(promedio))

                        #print('El promedio de
los ultimos ', cont1, 'conteos para
la VELOCIDAD es de ',promedio, '
m/min.')
```



```

        #print("El margen es:
", float(vel_set.value)-vel_tol)

        #print("\n")

        file.write("IA
VELOCIDAD SE SALIO DEL
MARGEN POR %s. M/MIN\n" %
str(float(vel_set.value)-vel_tol-
promedio))

        file.write("El margen
de la VELOCIDAD es: %s.
m/min\n" %
str(float(vel_set.value)-vel_tol))

        elif promedio >
(float(vel_set.value)-vel_tol):

            vel_real.tk.config(bg =
"#FFFFFF")

            vel_real.tk.config(fg =
"black")

            vel_alarma_activa.visible = False

            led_1.off()

            try:

                vel_error.value =
round((promedio-
float(vel_set.value)),2)

            except:

                ""

            data_to_send["Velocidad
Seteada"] =
round(float(vel_set.value),2)

            elif vel_line == 'g':

                cont0 = 0

                if cont_alarma == 0:

                    detencion_window.show()

                    tipo_paro.value = "Paro
no Justificado"

                    razon_detencion.value =
"Paro no Justificado"

                    tipo_paro.tk.config(bg =
"red")

                    tipo_paro.show()

                    tipo_hora.hide()

                    data_to_send["Razon
Paro"] = tipo_paro.value

```

```

            data_to_send["Tipo
Hora"] = "Improductiva"

            data_to_send["date"]=str(datetime.
datetime.now())

            r=
requests.post("https://hook.integro
mat.com/xwymmmjgdovqwlvygq14
kyeuajdi4383",datetime.datetime.n
ow().strftime("%Y-%m-%d
%H:%M:%S"))

            if (r.status_code ==
200):

                file.write("Se envio la
alerta por detencion a
https://hook.integromat.com/xwym
mmjgdovqwlvygq14kyeuajdi4383
a las %s." %
str(datetime.datetime.now().strftim
e("%Y-%m-%d %H:%M:%S")))

                print ('La rueda se
detuvo')

                file.write("LA RUEDA SE
DETUVO A LAS %s. \n" %
str(datetime.datetime.now().strftim
e("%Y-%m-%d %H:%M:%S")))

                #print('\n')

                cont_alarma +=1

                elif cont_alarma !=0:

                    vel_alarma_activa.visible = True

                    led_1.on()

                    cont_alarma += 1

                    vel_real.tk.config(bg =
"red")

                    vel_real.tk.config(fg =
"#FFFFFF")

                    vel_real.value = "0 m/min"

                    vel_error.value =
vel_set.value

                    data_to_send["Velocidad
Real"]= 0

                    data_to_send["Diferencia
Velocidad"]=
round(float(vel_error.value),2)

                    try:

```

```

            data_to_send["Velocidad
Seteada"] =
round(float(vel_set.value),2)

            except:

                ""

                #pdb.set_trace()

                file.write("\n")

                file.close()

                "" ANCHO INICIAL ""

                def ancho_ini(a_ini_real,
a_ini_serial, a_ini_set, a_ini_error,
a_ini_alarma_activa,

                    config_a_ini_tol,
data_to_send, stream):

                        contador = 0;

                        pasos = 200

                        circunferencia = 4.1

                        puerto_ancho= a_ini_serial

                        ancho = 0

                        alarma = 0

                        while True:

                            file = open(stream, 'a+')

                            ini_Bytes =
puerto_ancho.readline()

                            try:

                                ini_line =
ini_Bytes.decode('utf-8').strip()

                                try:

                                    ini_line = int(ini_line,16)

                                    #print('El numero de
pasos para el ANCHO INICIAL es
', str(int(ini_line)))

                                    file.write("El numero de
pasos para el ANCHO INICIAL es
%s \n"% str(ini_line))

                                    if ini_line < 10:

                                        ""

                                    else:

                                        contador = contador +
1

```

```

        medida =
((ini_line/pasos)*circunferencia)

        #print ("La medida del
ANCHO INICIAL es de ", medida,
" cm")

        file.write("La medida
del ANCHO INICIAL es de %s
cm\n" % str(medida))

        ancho =
(ancho*(contador-1) +
medida)/contador

        ancho =
round(ancho,2)

        #print("")

        #print ("El promedio
de las ultimas ", contador, "
medidas del ANCHO INICIAL es ",
ancho, " cm")

        file.write("El promedio
de las ultimas %s "%
str(contador))

        file.write("medidas
para el ANCHO INICIAL es %s cm
\n" % str(ancho))

        if contador == 2:

            a_ini_real.value =
ancho

            #ancho = medida

            contador = 0

            try:

                error =
round((ancho -
float(a_ini_set.value)),2)

                a_ini_error.value
= error

                if (abs(error) >
float(config_a_ini_tol.value)):

                    led_2.on()

            a_ini_alarma_activa.visible = True

            a_ini_real.tk.config(bg = "red")

            a_ini_real.tk.config(fg =
"#FFFFFF")

            alarma = 1

            file.write("\n")

            file.write("EL
ANCHO INICIAL SE SALIO DEL
MARGEN POR %s cm " %
str(error))

            file.write("A
LAS %s \n" %
str(datetime.datetime.now().strftim
e("%Y-%m-%d %H:%M:%S")))

            file.write("La
tolerancia para el ANCHO INICIAL
es de %s cm \n" %
str(config_a_ini_tol.value))

            file.write("\n")

            elif (abs(error)
<= float(config_a_ini_tol.value)):

                led_2.off()

            a_ini_alarma_activa.visible =
False

            a_ini_real.tk.config(bg =
"#FFFFFF")

            a_ini_real.tk.config(fg = "black")

            file.write("El
ANCHO INICIAL esta dentro del
margen.")

            alarma = 0

            except:

                #print("El
ANCHO INICIAL no tiene formato
Hex")

                file.write("El
ANCHO INICIAL no tiene formato
Hex")

                #print("El ANCHO
INICIAL esperado es
",a_ini_set.value, " cm")

                file.write("El
ANCHO INICIAL esperado es %s
cm\n"% str(a_ini_set.value))

                #print('El ANCHO
INICIAL medido es ', ancho, ' cm')

                file.write("El
ANCHO INICIAL medido es %s
cm\n"% str(ancho))

                #print("")

                #pdb.set_trace()

            data_to_send["Ancho Inicial
Real"]= float(a_ini_real.value)

            data_to_send["Ancho Inicial Set"]=
float(a_ini_set.value)

            data_to_send["Diferencia Ancho
Inicial"]= float(a_ini_error.value)

            data_to_send["Tolerancia Ancho
Inicial"]=
float(config_a_ini_tol.value)

            data_to_send["date"] =
datetime.datetime.now().strftime("
%Y-%m-%d %H:%M:%S")

            #r=
requests.post("https://hook.integro
mat.com/a6tmoarb9bimlgl91nj3nyf
rqme25t5e",data_to_send)

            #print(r.status_code)

            except:

                ""

                if (ini_line) == "I":

                    #print("El sensor para
el ANCHO INICIAL no ha
encontrado nada\n")

                    file.write("El sensor
para el ANCHO INICIAL no ha
encontrado nada\n")

                    a_ini_real.value =
0.00

                    try:

                        error =
round((float(a_ini_real.value) -
float(a_ini_set.value)),2)

                        a_ini_error.value
= error

                        if (abs(error) >
float(config_a_ini_tol.value)):

                            led_3.on()

                    a_ini_alarma_activa.visible = True

                    a_ini_real.tk.config(bg = "red")

```

```

a_ini_real.tk.config(fg =
"#FFFFFF")

        alarma = 1

        file.write("\n")

        file.write("EL
ANCHO INICIAL SE SALIO DEL
MARGEN POR %s cm " %
str(error))

        file.write("A
LAS %s \n" %
str(datetime.datetime.now().strftim
e("%Y-%m-%d %H:%M:%S")))

        file.write("La
tolerancia para el ANCHO INICIAL
es de %s cm \n" %
str(config_a_ini_tol.value))

        file.write("\n")

        elif (abs(error)
<= float(config_a_ini_tol.value)):

                led_3.off()

a_ini_alarma_activa.visible =
False

a_ini_real.tk.config(bg =
"#FFFFFF")

a_ini_real.tk.config(fg = "black")

        file.write("El
ANCHO INICIAL esta dentro del
margen.")

data_to_send["Ancho Inicial
Real"]= float(a_fin_a_real.value)

data_to_send["Ancho Inicial Set"]=
float(a_fin_a_set.value)

data_to_send["Diferencia Ancho
Inicial"]= float(a_fin_a_error.value)

data_to_send["Tolerancia Ancho
Inicial"]=
float(config_a_fin_a_tol.value)

data_to_send["date"] =
datetime.datetime.now().strftime("
%Y-%m-%d %H:%M:%S")

        #print("El
ANCHO INICIAL esperado es
",a_ini_set.value, " cm")

        file.write("El
ANCHO INICIAL esperado es %s
cm\n"% str(a_ini_set.value))

        #print("El
ANCHO INICIAL medido es ',
ancho, ' cm')

        file.write("El
ANCHO INICIAL medido es %s
cm\n"% str(a_fin_a_real.value))

        #print("La informacion
del ANCHO INICIAL no se puede
interpretar")

        #file.write("La
informacion del ANCHO INICIAL
no se puede interpretar")

        except:

                ""

        except:

                ""

        #print("La entrada en el
puerto de ANCHO INICIAL no
tiene formato de linea")

        #file.write("El ANCHO
INICIAL no tiene formato de
linea")

        file.close()

        ""ANCHO FINAL EN A""

def ancho_fin_a(a_fin_a_real,
a_fin_a_serial, a_fin_a_set,
a_fin_a_error,
a_fin_a_alarma_activa,

        config_a_fin_a_tol,
data_to_send, stream):

        contador = 0;

        pasos = 200

        circunferencia = 4.1

        puerto_ancho= a_fin_a_serial

        ancho = 0

        alarma = 0

        while True:

                file = open(stream, 'a+')

                ini_Bytes =
puerto_ancho.readline()

                try:

                        ini_line =
ini_Bytes.decode('utf-8').strip()

                try:

                        ini_line = int(ini_line,16)

                        #print('El numero de
pasos para el ANCHO INICIAL es
', str(int(ini_line)))

                        file.write("El numero de
pasos para el ANCHO FINAL A es
%s \n"% str(ini_line))

                        if ini_line < 10:

                                ""

                        else:

                                contador = contador +
1

                                medida =
((ini_line/pasos)*circunferencia)

                                #print ("La medida del
ANCHO INICIAL es de ", medida,
" cm")

                                file.write("La medida
del ANCHO FINAL A es de %s
cm\n" % str(medida))

                                ancho =
(ancho*(contador-1) +
medida)/contador

                                ancho =
round(ancho,2)

                                #print("")

                                #print ("El promedio
de las ultimas ", contador, "
medidas del ANCHO INICIAL es ",
ancho, " cm")

                                file.write("El promedio
de las ultimas %s "%
str(contador))

                                file.write("medidas
para el ANCHO FINAL A es %s
cm \n" % str(ancho))

                                #pdb.set_trace()

                                if contador == 2:

                                        a_fin_a_real.value
= ancho

```

```

#ancho = medida
contador = 0

try:
    error =
round((ancho -
float(a_fin_a_set.value)),2)

a_fin_a_error.value = error

    if (abs(error) >
float(config_a_fin_a_tol.value)):
        led_3.on()

a_fin_a_alarma_activa.visible =
True

a_fin_a_real.tk.config(bg = "red")

a_fin_a_real.tk.config(fg =
"#FFFFFF")

        alarma = 1

        file.write("\n")

        file.write("EL
ANCHO FINAL A SE SALIO DEL
MARGEN POR %s cm " %
str(error))

        file.write("A
LAS %s \n" %
str(datetime.datetime.now().strftim
e("%Y-%m-%d %H:%M:%S")))

        file.write("La
tolerancia para el ANCHO FINAL
A es de %s cm \n" %
str(config_a_ini_tol.value))

        file.write("\n")

        elif (abs(error)
<= float(config_a_fin_a_tol.value)):

            led_3.off()

a_fin_a_alarma_activa.visible =
False

a_fin_a_real.tk.config(bg =
"#FFFFFF")

a_fin_a_real.tk.config(fg = "black")

```

```

file.write("El
ANCHO FINAL A esta dentro del
margen.")

        alarma = 0

        except:

            ""

            #print("El
ANCHO INICIAL no tiene formato
Hex")

            #file.write("El
ANCHO FINAL A no tiene formato
Hex")

data_to_send["Ancho Final A
Real"]= float(a_fin_a_real.value)

data_to_send["Ancho Final A
Set"]= float(a_fin_a_set.value)

data_to_send["Diferencia Ancho
A"]= float(a_fin_a_error.value)

data_to_send["Tolerancia Ancho
Final A"]=
float(config_a_fin_a_tol.value)

data_to_send["date"] =
datetime.datetime.now().strftime("
%Y-%m-%d %H:%M:%S")

        #print("El ANCHO
INICIAL esperado es
",a_ini_set.value, " cm")

        file.write("El
ANCHO FINAL A esperado es %s
cm\n"% str(a_ini_set.value))

        #print('El ANCHO
INICIAL medido es ', ancho, ' cm')

        file.write("El
ANCHO FINAL A medido es %s
cm\n"% str(ancho))

        #print("")

        #pdb.set_trace()

        #r=
requests.post("https://hook.integro
mat.com/a6tmoarb9bimlg191nj3nyf
rqme25t5e",data_to_send)

#print(r.status_code)

        except:

```

```

""

        if (ini_line) == "I":

            print("El sensor para
el ANCHO FINAL A no ha
encontrado nada\n")

            file.write("El sensor
para el ANCHO FINAL A no ha
encontrado nada\n")

            a_fin_a_real.value =
0.00

            try:

                error =
round((float(a_fin_a_real.value) -
float(a_fin_a_set.value)),2)

                a_fin_a_error.value = error

                if (abs(error) >
float(config_a_fin_a_tol.value)):

                    led_3.on()

                a_fin_a_alarma_activa.visible =
True

                a_fin_a_real.tk.config(bg = "red")

                a_fin_a_real.tk.config(fg =
"#FFFFFF")

                alarma = 1

                file.write("\n")

                file.write("EL
ANCHO FINAL A SE SALIO DEL
MARGEN POR %s cm " %
str(error))

                file.write("A
LAS %s \n" %
str(datetime.datetime.now().strftim
e("%Y-%m-%d %H:%M:%S")))

                file.write("La
tolerancia para el ANCHO FINAL
A es de %s cm \n" %
str(config_a_ini_tol.value))

                file.write("\n")

                elif (abs(error)
<= float(config_a_fin_a_tol.value)):

                    led_3.off()

                a_fin_a_alarma_activa.visible =
False

```

```

a_fin_a_real.tk.config(bg =
"#FFFFFF")

a_fin_a_real.tk.config(fg = "black")

file.write("El
ANCHO FINAL A esta dentro del
margen.")

data_to_send["Ancho Final A
Real"]= float(a_fin_a_real.value)

data_to_send["Ancho Final A
Set"]= float(a_fin_a_set.value)

data_to_send["Diferencia Ancho
A"]= float(a_fin_a_error.value)

data_to_send["Tolerancia Ancho
Final A"]=
float(config_a_fin_a_tol.value)

data_to_send["date"] =
datetime.datetime.now().strftime("
%Y-%m-%d %H:%M:%S")

#print("El
ANCHO INICIAL esperado es
",a_ini_set.value, " cm")

file.write("El
ANCHO FINAL A esperado es %s
cm\n"% str(a_ini_set.value))

#print('El
ANCHO INICIAL medido es ',
ancho, ' cm')

file.write("El
ANCHO FINAL A medido es %s
cm\n"% str(a_fin_a_real.value))

except:
    ""

#print("La informacion
del ANCHO INICIAL no se puede
interpretar")

#file.write("La
informacion del ANCHO FINAL A
no se puede interpretar")

except:
    ""

#print("La entrada en el
puerto de ANCHO INICIAL no
tiene formato de linea")

#file.write("El ANCHO
FINAL A no tiene formato de
linea")

file.close()

""ANCHO FINAL EN B""

def ancho_fin_b(a_fin_b_real,
a_fin_b_serial, a_fin_b_set,
a_fin_b_error,
a_fin_b_alarma_activa,

config_a_fin_b_tol,
data_to_send, stream):

    contador = 0;

    pasos = 200

    circunferencia = 4.1

    puerto_ancho= a_fin_b_serial

    ancho = 0

    alarma = 0

    while True:

        file = open(stream, 'a+')

        ini_Bytes =
puerto_ancho.readline()

        try:

            ini_line =
ini_Bytes.decode('utf-8').strip()

            #print('El numero de pasos
para el ANCHO FINAL B es ',
str(int(ini_line)))

            try:

                ini_line = int(ini_line,16)

                file.write("El numero de
pasos para el ANCHO FINAL B es
%s \n"% str(ini_line))

                if ini_line < 10:

                    ""

                else:

                    contador = contador +
1

                    medida =
((ini_line/pasos)*circunferencia)

                    #print ("La medida del
ANCHO FINAL B es de ", medida,
" cm")

                    file.write("La medida
del ANCHO FINAL B es de %s
cm\n" % str(medida))

                    ancho =
(ancho*(contador-1) +
medida)/contador

                    ancho =
round(ancho,2)

                    #print("")

                    #print ("El promedio
de las ultimas ", contador, "
medidas del ANCHO FINAL B es
", ancho, " cm")

                    file.write("El promedio
de las ultimas %s "%
str(contador))

                    file.write("medidas
para el ANCHO FINAL B es %s
cm \n" % str(ancho))

                    if contador == 2:

                        a_fin_b_real.value
= ancho

                        contador = 0

                        try:

                            error =
round((ancho -
float(a_fin_b_set.value)),2)

                            a_fin_b_error.value = error

                            if (abs(error) >
float(config_a_fin_b_tol.value)):

                                led_4.on()

                                a_fin_b_alarma_activa.visible =
True

                                a_fin_b_real.tk.config(bg = "red")

                                a_fin_b_real.tk.config(fg =
"#FFFFFF")

                                alarma = 1

                                file.write("\n")

```

```

file.write("EL
ANCHO FINAL B SE SALIO DEL
MARGEN POR %s cm " %
str(error))

file.write("A
LAS %s \n" %
str(datetime.datetime.now().strftim
e("%Y-%m-%d %H:%M:%S")))

file.write("La
tolerancia para el ANCHO FINAL
B es de %s cm \n" %
str(config_a_ini_tol.value))

file.write("\n")

elif (abs(error)
<= float(config_a_fin_b_tol.value)):

led_4.off()

a_fin_b_alarma_activa.visible =
False

a_fin_b_real.tk.config(bg =
"#FFFFFF")

a_fin_b_real.tk.config(fg = "black")

file.write("El
ANCHO FINAL B esta dentro del
margen.\n")

alarma = 0

except:

#print("El
ANCHO FINAL B no tiene formato
Hex\n")

file.write("El
ANCHO FINAL B no tiene formato
Hex\n")

data_to_send["Ancho Final B
Real"]= float(a_fin_b_real.value)

data_to_send["Ancho Final B
Set"]= float(a_fin_b_set.value)

data_to_send["Diferencia Ancho
B"]= float(a_fin_b_error.value)

data_to_send["Tolerancia Ancho
Final B"]=
float(config_a_fin_b_tol.value)

```

```

data_to_send["date"] =
datetime.datetime.now().strftime("
%Y-%m-%d %H:%M:%S")

#print("El ANCHO
FINAL B esperado es
",a_ini_set.value, " cm")

file.write("El
ANCHO FINAL B esperado es %s
cm\n"% str(a_ini_set.value))

#print('El ANCHO
FINAL B medido es ', ancho, ' cm')

file.write("El
ANCHO FINAL B medido es %s
cm\n"% str(ancho))

#print("")

#pdb.set_trace()

#r=
requests.post("https://hook.integro
mat.com/a6tmoarb9bimlgl91nj3nyf
rqme25t5e",data_to_send)

#print(r.status_code)

except:

"""

if (ini_line) == "I":

print("El sensor para
el ANCHO FINAL B no ha
encontrado nada\n")

file.write("El sensor
para el ANCHO FINAL B no ha
encontrado nada\n")

a_fin_b_real.value =
0.00

try:

error =
round((float(a_fin_b_real.value) -
float(a_fin_b_set.value)),2)

a_fin_b_error.value = error

if (abs(error) >
float(config_a_fin_b_tol.value)):

led_3.on()

a_fin_b_alarma_activa.visible =
True

```

```

a_fin_b_real.tk.config(bg = "red")

a_fin_b_real.tk.config(fg =
"#FFFFFF")

alarma = 1

file.write("\n")

file.write("EL
ANCHO FINAL B SE SALIO DEL
MARGEN POR %s cm " %
str(error))

file.write("A
LAS %s \n" %
str(datetime.datetime.now().strftim
e("%Y-%m-%d %H:%M:%S")))

file.write("La
tolerancia para el ANCHO FINAL
B es de %s cm \n" %
str(config_a_ini_tol.value))

file.write("\n")

elif (abs(error)
<= float(config_a_fin_b_tol.value)):

led_3.off()

a_fin_b_alarma_activa.visible =
False

a_fin_b_real.tk.config(bg =
"#FFFFFF")

a_fin_b_real.tk.config(fg = "black")

file.write("El
ANCHO FINAL B esta dentro del
margen.")

data_to_send["Ancho Final B
Real"]= float(a_fin_a_real.value)

data_to_send["Ancho Final B
Set"]= float(a_fin_a_set.value)

data_to_send["Diferencia Ancho
B"]= float(a_fin_a_error.value)

data_to_send["Tolerancia Ancho
Final B"]=
float(config_a_fin_a_tol.value)

data_to_send["date"] =
datetime.datetime.now().strftime("
%Y-%m-%d %H:%M:%S")

```

```

        #print("El
ANCHO INICIAL esperado es
",a_ini_set.value, " cm")

        file.write("El
ANCHO FINAL B esperado es %s
cm\n"% str(a_ini_set.value))

        #print('El
ANCHO INICIAL medido es ',
ancho, ' cm')

        file.write("El
ANCHO FINAL B medido es %s
cm\n"% str(a_fin_a_real.value))

        #print("La informacion
del ancho FINAL B no se puede
interpretar")

        #file.write("La
informacion del ANCHO FINAL B
no se puede interpretar\n")

        except:
            ""

        except:
            ""

        #print("La entrada en el
puerto de ancho FINAL B no tiene
formato de linea")

        #file.write("El ANCHO
FINAL B no tiene formato de
linea\n")

"ALARMAS"
""

def alarmas(data_to_send,
tolerancia_vel, vel_error,
vel_alarma_activa,
tolerancia_a_ini, a_ini_error,
ancho_ini_alarma_activa,
tolerancia_fin_a, a_fin_a_error,
ancho_fin_a_alarma_activa,
tolerancia_fin_b, a_fin_b_error,
ancho_fin_b_alarma_activa):

    while True:

        sleep(1)

        try:

float(tolerancia_a_ini.value)

            except:

                tolerancia_a_ini.value =
0.5

            try:

float(a_ini_error.value)

            except:

                a_ini_error.value = 0

            if
(float(tolerancia_a_ini.value) <
abs(float(a_ini_error.value))):

                ancho_ini_alarma_activa.visible =
not
ancho_ini_alarma_activa.visible

            elif
(float(tolerancia_a_ini.value) >
abs(float(a_ini_error.value))):

                ancho_ini_alarma_activa.visible =
False

            try:

float(tolerancia_fin_a.value)

            except:

                tolerancia_fin_a.value =
0.5

            try:

float(a_fin_a_error.value)

            except:

                a_fin_a_error.value = 0

            if
(float(tolerancia_fin_a.value) <
abs(float(a_fin_a_error.value))):

                ancho_fin_a_alarma_activa.visible
= not
ancho_fin_a_alarma_activa.visible

            elif
(float(tolerancia_fin_a.value) >
abs(float(a_fin_a_error.value))):

                ancho_fin_a_alarma_activa.visible
= False

            ancho_fin_a_alarma_activa.visible
= False

float(tolerancia_fin_b.value)

            except:

                tolerancia_fin_b.value =
0.5

            try:

float(a_fin_b_error.value)

            except:

                a_fin_b_error.value = 0

            if
(float(tolerancia_fin_b.value) <
abs(float(a_fin_b_error.value))):

                ancho_fin_b_alarma_activa.visible
= not
ancho_fin_b_alarma_activa.visible

            elif
(float(tolerancia_fin_b.value) >
abs(float(a_fin_b_error.value))):

                ancho_fin_b_alarma_activa.visible
= False

            ancho_fin_b_alarma_activa.visible
= False

            try:

float(tolerancia_vel.value)

            except:

                tolerancia_vel.value = 5

            try:

float(vel_error.value)

            except:

                vel_error.value = 0

                if (float(tolerancia_vel.value)
< abs(float(vel_error.value))):

                    vel_alarma_activa.visible = not
vel_alarma_activa.visible

                elif
(float(tolerancia_vel.value) >
abs(float(vel_error.value))):

                    vel_alarma_activa.visible = False

                data_to_send["Tolerancia de
Velocidad"] = tolerancia_vel.value

                data_to_send["Tolerancia de
Ancho Inicial"] =
tolerancia_a_ini.value

                data_to_send["Tolerancia de
Ancho Final A"] =
tolerancia_fin_a.value

                data_to_send["Tolerancia de
Ancho Final B"] =
tolerancia_fin_b.value

```

```

"""
"""PRODUCCION"""
def calculo_produccion(produ_set,
    produ_calc, a_fin_a_set,
    a_fin_b_set, vel_set, a_fin_a_real,
        a_fin_b_real,
    presentacion_produ, vel_real,
    densidad_set, espesor_set,
    produ_calc_box, config_produ_tol,
        data_to_send):
    factor_produ = 1
    cont_factor = 0
    while True:
        #valores teoricos
        "Factor de Produccion segun
        la presentacion del producto"
        try:
            t_ancho_a =
            float(a_fin_a_set.value)
        except:
            t_ancho_a = 94
        try:
            t_ancho_b =
            float(a_fin_b_set.value)
        except:
            ""
        try:
            t_velocidad =
            float(vel_set.value)
        except:
            t_velocidad = 0
        try:
            t_densidad =
            float(densidad_set.value)
        except:
            t_densidad = 0.923
        try:
            t_espesor =
            float(espesor_set.value)
        except:
            t_espesor = 0.52

#valores medidos
try:
    r_ancho_a =
    float(a_fin_a_real.value)
except:
    r_ancho_a = 0
try:
    r_ancho_b =
    float(a_fin_b_real.value)
except:
    r_ancho_b = 0
try:
    r_velocidad =
    float(vel_real.value)
except:
    r_velocidad = 0
    "Factor de Produccion segun
    la presentacion del producto"
    factor_produ = 1
    if (presentacion_produ.value
    == "LAMINA"):
        factor_produ = 1
        cont_factor = 0
    elif
    (presentacion_produ.value ==
    "TUBO"):
        factor_produ = 2
        if cont_factor == 0:
            a_fin_a_set.value = 0
            a_fin_b_set.value = 0
            t_ancho_a = 0
            t_ancho_b = 0
            cont_factor = 1
        """Calculo de Valor de
        Produccion Teorico"""
        try:
            #vel_set.value =
            round(((65.617*(float(produ_set.val
            ue)/(float(densidad_set.value)*flea
            t(espesor_set.value)*factor_produ*
            (float(a_fin_a_set.value) +
            float(a_fin_b_set.value))))),2)
            vel_set.value =
            round(((65.617*(float(produ_set.val
            ue)/(t_densidad*t_espesor*factor_
            produ*(t_ancho_a +
            t_ancho_b))))),2)
        except:
            ""
            #pdb.set_trace()
            "Calculo de Valor de
            Produccion Real"
            r_produccion =
            ((factor_produ*(r_ancho_a +
            r_ancho_b)* r_velocidad *
            t_espesor * t_densidad*
            1.524)/100)
            r_produccion_provisional =
            (((t_ancho_a + t_ancho_b)*
            r_velocidad * t_espesor *
            t_densidad* 1.524)/100)
            produ_calc.value =
            round(r_produccion,2)
            if (float(produ_calc.value)<
            (float(produ_set.value)*(float(confi
            g_produ_tol.value)/100))):
                sleep(0.3)
            """desperdicio_real.tk.config(bg =
            "green")
            desperdicio_real_inner_box.tk.con
            fig(bg = "blue")"""
            produ_calc.tk.config(fg =
            "#FFFFFF")
            produ_calc.tk.config(bg =
            "red")
            produ_calc_box.tk.config(bg =
            "red")
            sleep(0.3)
            produ_calc.tk.config(fg =
            "red")
            produ_calc.tk.config(bg =
            "#FFFFFF")
            produ_calc_box.tk.config(bg =
            "#FFFFFF")

```



```

elif (float(produ_calc.value)>
(float(produ_set.value)*(float(config_produ_tol.value)/100))):

    produ_calc.tk.config(fg =
"#06AED5")

    produ_calc.tk.config(bg =
"#FFFFFF")

produ_calc_box.tk.config(bg =
"#FFFFFF")

    data_to_send["Espesor"] =
t_espesor

    data_to_send["Densidad"] =
t_densidad

    data_to_send["Produccion
Esperada"]= float(produ_set.value)

    data_to_send["Produccion
Real"]= float(produ_calc.value)

    data_to_send["Tolerancia
Produccion"] =
float(config_produ_tol.value)

    data_to_send["Presentacion"]
= presentacion_produ.value

    #pdb.set_trace()

"""DESPERDICIO"""

def calculo_desperdicio( a_ini_set,
a_fin_a_set, a_fin_b_set,
a_ini_real, a_fin_a_real,
a_fin_b_real,

desperdicio_esperado,
desperdicio_real, data_to_send,
config_a_ini_tol,

    config_a_fin_a_tol,
config_a_fin_b_tol,
presentacion_produ,
desperdicio_real_inner_box):

    while True:

        try:

            t_ancho_ini =
float(a_ini_set.value)

        except:

            t_ancho_ini = 0

        try:

            t_ancho_a =
float(a_fin_a_set.value)

        except:

            t_ancho_a = 0

            t_ancho_b = 0

            try:

                t_ancho_b =
float(a_fin_b_set.value)

            except:

                t_ancho_b = 0

            "Factor de Produccion segun
la presentacion del producto"

            factor_produ = 1

            if (presentacion_produ.value
== "LAMINA"):

                factor_produ = 1

            elif
(presentacion_produ.value ==
"TUBO"):

                factor_produ = 2

            t_desperdicio = 2 *
t_ancho_ini - (t_ancho_a +
t_ancho_b)

            desperdicio_esperado.value
= round(t_desperdicio,2)

            try:

                r_ancho_ini =
float(a_ini_real.value)

            except:

                r_ancho_ini = 0

            try:

                r_ancho_a =
float(a_fin_a_real.value)

            except:

                r_ancho_a = 0

            try:

                r_ancho_b =
float(a_fin_b_real.value)

            except:

                r_ancho_b = 0

            r_desperdicio = 2 *
r_ancho_ini - (r_ancho_a +
r_ancho_b)

            desperdicio_real.value =
round(r_desperdicio,2)

            margen_desperdicio =
float(desperdicio_esperado.value)

            +float(config_a_ini_tol.value)+float(
config_a_fin_a_tol.value)+float(
config_a_fin_b_tol.value)

            if
(float(desperdicio_real.value)>
margen_desperdicio):

                sleep(0.3)

            desperdicio_real.tk.config(fg =
"#FFFFFF")

            desperdicio_real.tk.config(bg =
"red")

            desperdicio_real_inner_box.tk.con
fig(bg = "red")

                sleep(0.3)

            desperdicio_real.tk.config(bg =
"#FFFFFF")

            desperdicio_real_inner_box.tk.con
fig(bg = "#FFFFFF")

            desperdicio_real.tk.config(fg =
"red")

            elif
(float(desperdicio_real.value)<
margen_desperdicio):

            desperdicio_real.tk.config(bg =
"#FFFFFF")

            desperdicio_real.tk.config(fg =
"#06AED5")

            desperdicio_real_inner_box.tk.con
fig(bg = "#FFFFFF")

            data_to_send["Desperdicio
Esperado"] =
float(desperdicio_esperado.value)

            data_to_send["Desperdicio
Real"] =
float(desperdicio_real.value)

            #pdb.set_trace()

            """ENVIO DE LOS DATOS A
INTEGROMAT"""

            def envio_datos(produ_tipo,
nombre_operador,
numero_operador, data_to_send,
variables, stream, tipo_paro):

```

```

minutos= 2

cont = 0

nombre_anterior =
name(datetime.datetime.now()).strftime("%Y-%m-%d_%H")

#nombre_anterior =
name(datetime.datetime.now()).strftime("%Y-%m-%d_%H-%M")

nombre =
name(datetime.datetime.now()).strftime("%Y-%m-%d_%H")

#nombre=
name(datetime.datetime.now()).strftime("%Y-%m-%d_%H-%M")

cont_a = 0

while True:

    data_to_send["Nombre del
Operador"] =
nombre_operador.value

    data_to_send["Numero de
Operador"] =
numero_operador.value

    data_to_send["Producto"] =
produ_tipo.value

    data_to_send["date"] =
datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")

    #nombre =
name(datetime.datetime.now()).strftime("%Y-%m-%d_%H-%M")

    nombre =
name(datetime.datetime.now()).strftime("%Y-%m-%d_%H")

    file = open(nombre, "a+")

    stream_f = open(stream,
"a+")

    if nombre_anterior != nombre:

        attachment =
open(nombre_anterior, "rb")

        print ("Se envia email con
archivo", nombre_anterior, ".\n")

        r =
requests.post("https://hook.integro
mat.com/t9i17x83fnu5t8oegfignhtr
w4e1vk5p",attachment)

        if (r.status_code == 200):

            stream_f.write("Se han
enviado los datos por correo a

```

```

https://hook.integromat.com/t9i17x
83fnu5t8oegfignhtrw4e1vk5p a las
%s \n" %
str(datetime.datetime.now()).strftim
e("%Y-%m-%d %H:%M:%S"))

    nombre_anterior = nombre

    cont_a = 0

    attachment.close()

    if cont_a == 0:

        for var in variables:

            file.write("%s, " %
str(var))

            #print("%s," % str(var))

            file.write("\n")

            file.close()

        else:

            for var in variables:

                try:

                    file.write("%s, " %
str(data_to_send[var]))

                except:

                    file.write(" , ")

                    file.write("\n")

                    file.close()

                #print
(datetime.datetime.now()).strftime("
%Y-%m-%d_%H-%M")

                cont_a += 1

                sleep(5)

                cont += 1

                #print (cont)

                if cont >= (minutos*60/5):

                    r=
requests.post("https://hook.integro
mat.com/a6tmoarb9bimlgl91nj3nyf
rqme25t5e",data_to_send)

                    if (r.status_code == 200):

                        stream_f.write("Se han
enviado los datos a
https://hook.integromat.com/a6tmo
arb9bimlgl91nj3nyfrqme25t5e a
las %s \n" %
str(datetime.datetime.now()).strftim
e("%Y-%m-%d %H:%M:%S"))

```

```

stream_f.write("\n")

    stream_f.write("El tipo de
producto es %s \n" %
str(produ_tipo.value))

    stream_f.write("El nombre
del operador es %s \n" %
str(nombre_operador.value))

    stream_f.write("El numero
del operador es %s \n" %
str(numero_operador.value))

    try:

        stream_f.write("La
presentacion del producto es %s
\n" %
str(data_to_send["Presentacion"]))

    except:

        stream_f.write("No se ha
seleccionado la presentacion \n")

    try:

        stream_f.write("El tipo
de hora es %s \n" %
str(data_to_send["Tipo Hora"]))

    except:

        stream_f.write("No se ha
seleccionado el tipo de hora \n")

    try:

        stream_f.write("La razon
de paro es %s \n" %
str(data_to_send["Razon de
Detencion"]))

    except:

        stream_f.write("No se ha
seleccionado la razon de paro \n")

        stream_f.write("\n")

        stream_f.close()

        print (data_to_send)

        print ("")

        cont = 1

        #pdb.set_trace()

        #pdb.set_trace()

def name(s):

    return s + ".txt"

```

```

def temp_setup (tipo_hora,
detencion_window,
nombre_operador,
numero_operador, stream):

    cont_setup = 0

    minutos = 1

    if (tipo_hora.value == "Set Up")
and cont_setup == 0:

        file = open(stream, "a+")

        sleep(minutos*60)

        detencion_window.show()

        file.write("\n")

        file.write("El setup por el
operador %s " %
str(nombre_operador.value))

        file.write("numero %s " %
str(numero_operador.value))

        file.write("debia finalizar a las
%s. \n" %
str(datetime.datetime.now().strftim
e("%Y-%m-%d %H:%M")))

        file.write("Se contaba con %s
minutos para finalizar \n" %
str(minutos))

        file.write("\n")

        print("El setup duró mucho")

        print("")

        cont_setup = 1

        file.close()

        if (tipo_hora.value ==
"Produccion"):

            cont_setup = 0

            #pdb.set_trace()

```

## Apéndice J – Código Python de la Biblioteca Variables

```

variables = ["date", "Velocidad Seteada", "Velocidad
Real", "Diferencia Velocidad", "Ancho Inicial Set", "Ancho
Inicial Real",
"Diferencia Ancho Inicial", "Tolerancia Ancho
Inicial", "Ancho Final A Set", "Ancho Final A
Real", "Diferencia Ancho A",
"Tolerancia Ancho Final A", "Ancho Final B
Set", "Ancho Final B Real", "Tolerancia Ancho Final
B", "Densidad", "Espesor",
"Desperdicio Esperado", "Desperdicio
Real", "Produccion Esperada", "Produccion
Real", "Tolerancia Produccion",
"Nombre del Operador", "Numero de
Operador", "Producto", "Presentacion", "Tipo Hora", "Razon
de Detencion"]

```

```

"Falta de Tintas", "Falta o Espera de
Montajes", "Foto Polímero Dañado (Produc)",
"Foto Polímero defectuoso(Arte)", "Limpieza de
Dado", "Limpieza de equipo",
"Mal Montaje de Planchas", "Mant Correctivo
Eléctrico/Elec", "Mant Correctivo Mecánico",
"Paro no programación", "Paro por Control de
Calidad", "Problema Materia Prima/Insumo",
"Problema con Tintas", "Problemas con
Rollo", "Problemas con adhesivo", "Reuniones"]

```

## Apéndice K – Código Python de la Biblioteca Improductivos

```

improductivos = ["Paro no Justificado", "Ajustes máquina en
Set Up", "Camb/Ajuste Matr/ Dado Set Up",
"Entonación de tintas (Set Up)", "Inspección de
calidad (Set Up)", "Montaje Mecánico",
"Ajuste de Imprenta en Proceso", "Ajuste de
maquina por pruebas", "Aprobación del cliente",
"Arranque de máquina", "Calentamiento de
Máquina", "Cambio de Dado", "Cambio de Filtros",
"Cambio de Rollo", "Cambio de Tesa", "Choque de
mangas programado", "Comidas",
"Corte electricidad", "Cuadre de Horas
Improductivas", "Desmontar rollos",
"Error Configuración Producto", "Error en
Configuración (I&D)", "Espera Entonación de Tintas",
"Falla- Falta consumibles, pist", "Falta Materia
Prima/Insumos", "Falta Equipo Alisto",
"Falta de Coproducto", "Falta de Jumbos", "Falta de
Personal", "Falta de Repuesto",

```

## Capítulo 9: Anexos

### Anexo A – Código Python para Identificación de Puertos Seriales [35]

```
#!/usr/bin/python

import os
import time

def GetUsbList():
    return
os.popen("lsusb").read().strip().split("\n")

def GetDevList():
    return os.listdir("/dev")

def Changed(old,now):
    add = []
    rem = []
    for this in now:
        if not this in old:
            add.append(this)
    for this in old:
        if not this in now:
            rem.append(this)
    return add, rem

try:
    print("Monitoring for USB changes
and changes in /dev directory")
    usbOld = GetUsbList()
    devOld = GetDevList()
    while True:
        time.sleep(1)
        usbNow = GetUsbList()
        devNow = GetDevList()
        usbAdd, usbRem =
Changed(usbOld,usbNow)
        devAdd, devRem =
Changed(devOld,devNow)
        if len(usbAdd) + len(usbRem) +
len(devAdd) + len(devRem) > 0:
            print("-----")
            t = time.strftime("%Y-%m-%d
%H:%M:%S - ")
            for this in usbAdd : print(t +
"Added  :" + this)
            for this in usbRem : print(t +
"Removed : " + this)
            for this in devAdd : print(t +
"Added  : /dev/" + this)
            for this in devRem : print(t +
"Removed : /dev/" + this)
            usbOld = usbNow
            devOld = devNow
except KeyboardInterrupt:
    print("")
```

## Anexo B – Biblioteca USART.h

```
/*
 * IncFile1.h
 *
 * Created: 21/01/2021 10:49:40 a.m.
 * Author: Smurillob
 */

#ifndef INCFILE1_H_
#define INCFILE1_H_

/*UART Buffer Defines*/
#define USART_RX_BUFFER_SIZE
128 /* 2,4,8,16,32,64,128 or 256
bytes */
#define USART_TX_BUFFER_SIZE
128 /* 2,4,8,16,32,64,128 or 256
bytes */
#define USART_RX_BUFFER_MASK
(USART_RX_BUFFER_SIZE - 1)
```

```
#define USART_TX_BUFFER_MASK
(USART_TX_BUFFER_SIZE - 1)

/* Prototypes */
void USART0_Init(unsigned int
baudrate);
unsigned char
USART0_Receive(void);
void USART0_Transmit(unsigned
char data);
void USART_putstring(char*
StringPtr);
int USART0_Transmit_IO(char data,
FILE *stream);
void msg(unsigned msg, FILE
stream);

// todo funciona antes de esta linea

#endif /* INCFILE1_H_ */
```

## Anexo C- Biblioteca USART.c

```

/*
 * USART_irq.c
 *
 * Created: 21/01/2021 10:51:21 a.m.
 * Author: Smurillob
 */
#include <avr/interrupt.h>
#include <avr/io.h>
// nueva linea
#include <stdio.h>
//
#include "USART_irq.h"
/* Static Variables */
static unsigned char
USART_RxBuf[USART_RX_BUFFER_SIZE];
static volatile unsigned char
USART_RxHead;
static volatile unsigned char USART_RxTail;
static unsigned char
USART_TxBuf[USART_TX_BUFFER_SIZE];
static volatile unsigned char USART_TxHead;
static volatile unsigned char USART_TxTail;
void msg(unsigned msg,FILE stream){
    printf("\n %u\n", msg);
}
void USART0_Init(unsigned int baudrate)
{
    unsigned char x;

    /* Set the baud rate */
    UBRROH = (unsigned char)
(baudrate>>8);
    UBRROL = (unsigned char)
baudrate;
    /* Enable USART receiver and
transmitter */
    UCSR0B = ((1 << RXCIE0) | (1 <<
RXEN0) | (1 << TXEN0));
    /* For devices in which
UBRRH/UCSRC shares the same location
* eg; ATmega16, URSEL should be
written to 1 when writing UCSRC
*/
    /*
    /* Set frame format: 8 data 1stop */
    UCSR0C = (1 << UCSZ01) | (1 <<
UCSZ00);
    /* Flush receive buffer */
    x = 0;

    USART_RxTail = x;
    USART_RxHead = x;
    USART_TxTail = x;

```

```

    USART_TxHead = x;
}

ISR (USART_RX_vect)
{
    unsigned char data;
    unsigned char tmphead;
    /* Read the received data */
    data = UDR0;
    /* Calculate buffer index */
    tmphead = (USART_RxHead + 1)
& USART_RX_BUFFER_MASK;
    /* Store new index */
    USART_RxHead = tmphead;

    if (tmphead == USART_RxTail) {
        /* ERROR! Receive
buffer overflow */
    }
    /* Store received data in buffer */
    USART_RxBuf[tmphead] = data;
}

ISR(USART_UDRE_vect)
{
    unsigned char tmptail;

    /* Check if all data is transmitted */
    if (USART_TxHead !=
USART_TxTail) {
        /* Calculate buffer index
*/
        tmptail = (USART_TxTail
+ 1) & USART_TX_BUFFER_MASK;
        /* Store new index */
        USART_TxTail = tmptail;
        /* Start transmission */
        UDR0 =
USART_TxBuf[tmptail];
    } else {
        /* Disable UDRE
*/
        UCSR0B &=
~(1<<UDRIE0);
    }
}
unsigned char USART0_Receive(void)
{
    unsigned char tmptail;

    /* Wait for incoming data */
    while (USART_RxHead ==
USART_RxTail);
    /* Calculate buffer index */
    tmptail = (USART_RxTail + 1) &
USART_RX_BUFFER_MASK;

```

```

        /* Store new index */
        USART_RxTail = tmptail;
        /* Return data */
        return USART_RxBuf[tmptail];
    }
void USART0_Transmit(unsigned char data)
{
    unsigned char tmphead;

    /* Calculate buffer index */
    tmphead = (USART_TxHead + 1)
    & USART_TX_BUFFER_MASK;
    /* Wait for free space in buffer */
    while (tmphead ==
    USART_TxTail);
    /* Store data in buffer */
    USART_TxBuf[tmphead] = data;
    /* Store new index */
    USART_TxHead = tmphead;
    /* Enable UDRE interrupt */
    UCSR0B |= (1<<UDRIE0);
}
unsigned char DataInReceiveBuffer(void){
    return (USART_RxHead !=
    USART_RxTail); // Return 0 if the receive
    buffer is empty
}
void USART_putstring(char* StringPtr){
    while(*StringPtr != 0x00){

        USART0_Transmit(*StringPtr);
        StringPtr++;
    }
}
//unsigned char USART_Receive(void)
//{
//    /*wait for data to be
//    received */
//    while (!(UCSR0A &
//    (1<<RXC0)));
//    //
//    /* Get and return
//    received data from buffer */
//    //return UDR0;
//}
//
//void USART_Transmit( unsigned char data)
//{
//    /* wait for empty transmit buffer */
//    while ( !(UCSR0A &
//    (1<<UDRE0)));
//}

        /*Puts the data into buffer, sends
        the data*/
        //    UDR0 = data;
        //}

//void USART_Init( unsigned int ubrr)
//{
//    /*Set baud rate */
//    //UBRR0H = (unsigned char)
//    (ubrr>>8);
//    //UBRR0L = (unsigned char) ubrr;
//    /*Enable receiver and transmitter */
//    //UCSR0B =
//    (1<<RXEN0)|(1<<TXEN0);
//    /*Set frame format: 8 data, 2 stop
//    bit */
//    //UCSR0C = (3<<UCSZ00);
//    //}

//int main(void)
//{
//    // USART_Init(MYUBRR);
//    //while (1){

//        //USART_Transmit(USART_Receive
//    e());
//    //}
//}

//static int uart_putchar(char c, FILE *stream);

int USART0_Transmit_IO(char data, FILE
*stream)
{
    unsigned char tmphead;

    /* Calculate buffer index */
    tmphead = (USART_TxHead + 1)
    & USART_TX_BUFFER_MASK;
    /* Wait for free space in buffer */
    while (tmphead ==
    USART_TxTail);
    /* Store data in buffer */
    USART_TxBuf[tmphead] = data;
    /* Store new index */
    USART_TxHead = tmphead;
    /* Enable UDRE interrupt */
    UCSR0B |= (1<<UDRIE0);
}

```