

| | | | |
|--|--------------------------------------|---------|------------|
|  Institución Universitaria | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-27 |

Desarrollo de un ruteador duplicador 4D automatizado

Ref.: TGP-ME-1325

John Camilo Giraldo López

Tecnología en sistemas electromecánicos

Jorge Andrés Sierra del Rio

INSTITUTO TECNOLÓGICO METROPOLITANO

Mayo 15 de 2022

| | | | |
|---|--------------------------------------|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

RESUMEN

La estandarización y automatización es una necesidad creciente dentro de las industrias productivas. Los robots son cada vez máquinas más requeridas para la realización de tareas que requieren de altos grados de precisión y repetibilidad. En la industria de la generación energética, se hace necesario la creación de herramientas que proporcionen dichas condiciones en la reparación de instrumentos claves como lo son las turbinas Pelton, que por causas como el desgaste por corrosión y abrasiones causadas por elementos con los que entran en contacto sufren diversas averías. Se plantea entonces la creación de un sistema embebido que mediante el sensado por medio de un sensor de carga y el fresado simultaneo, recree la geometría de una pieza patrón en otra con material adicionado o de corte; todo controlado mediante un microcontrolador que implemente un control retroalimentado que detecte la pieza.

Palabras clave: **Recreación, geometría, sensor de carga, fresado, simultáneo, microcontrolador, sistema embebido.**

| | | | |
|---|--------------------------------------|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

RECONOCIMIENTOS

Agradezco primeramente que me ha apoyado y sostenido a lo largo del proceso; a mis compañeros que me acompañaron y trabajaron conmigo durante muchas jornadas; y finalmente a la institución y su personal docente, administrativo y espacios que estuvieron siempre disponibles para las consultas y prácticas durante mi estancia.

Y en segunda instancia agradezco a los trabajos, proyectos y guías que inspiraron este proyecto especialmente a: Erik (el profe) García, Clone UK, Instructables, Arduino, Allegro y demás compañías proveedoras de componentes mecánicos, eléctricos y electrónicos.

| | | | |
|--|--------------------------------------|---------|------------|
|  Institución Universitaria | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

ACRÓNIMOS

CNC Control numérico por computadora.

PCB Printed Circuit Board (placa de circuito impreso)

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

TABLA DE CONTENIDO

| | |
|---|----|
| 1. INTRODUCCIÓN | 6 |
| 2. MARCO TEÓRICO | 8 |
| 3. METODOLOGÍA..... | 11 |
| 4. RESULTADOS Y DISCUSIÓN..... | 15 |
| 5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO | 20 |
| REFERENCIAS | 21 |
| APÉNDICE..... | 22 |

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

1. INTRODUCCIÓN

Esta sección se divide en tres partes: *Generalidades*, donde se presenta de manera resumida la pertinencia, justificación y problema abordado; *objetivos*, donde se escribe textualmente los objetivos, general y específicos, aprobados en el anteproyecto; y *organización de la tesis*, donde se describe en no más de cinco párrafos, lo que aborda cada uno de los capítulos de la tesis.

1.1 Generalidades

En la industria (en general) se hace necesario el uso de máquinas y mecanismos complejos para la producción de bienes y servicios. Su uso implícitamente conlleva la probabilidad de que los mismos sufran averías y/o fallas por desgaste, o se vuelvan obsoletos. Se hace necesario entonces, el reemplazo o reparación de uno o varios de sus componentes a lo largo de su vida útil. Sin embargo, para piezas de gran envergadura y/o geometrías complejas (y ya que muchas veces dichos mecanismos son comprados a compañías extranjeras especializadas) los procesos de: embarque, desembarque y recepción se vuelven lentos, complicados y costosos. Debido a lo anterior es aceptable entonces pensar en la reproducción o reparación de estas piezas mediante el maquinado manual y artesanal local de las mismas; Lo que finalmente reduce costos, pero minimiza la precisión y mantiene un tiempo prolongado de espera, debido a que las piezas remanufacturadas deben mantener estándares de calidad mínimos para el óptimo funcionamiento de los mecanismos. Es por esto por lo que, planteando una solución para la repetibilidad y el óptimo desarrollo de las actividades, se proponen mecanismos autónomos que mejoren la calidad.

1.2 Objetivos

1.2.1 General

Diseñar a nivel de detalle una máquina ruteadora 4D automática capaz de replicar piezas de diversos materiales y geometrías a partir de una pieza patrón.

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

1.2.2 Específicos

- Establecer los criterios de diseño requeridos para la concepción del dispositivo ruteador 4D.
- Diseñar y seleccionar los componentes que integran la estructura mecánica del ruteador 4D.
- Modelar y simular el circuito de control necesario para realizar movimientos automáticos de la ruteadora 4D.

1.3 Organización de la tesis

El proyecto consiste entonces, en generar una metodología de automatizado para los procesos de reparación y/o remanufactura de piezas con geometrías medianamente complejas que deban ser replicadas con precisión para un óptimo funcionamiento. Utilizando como referencia para la medición celdas de carga y para el control un microcontrolador ATMEGA2560 programado mediante el IDE Arduino.

| | | | |
|---|--|---------|------------|
|  | <p style="text-align: center;">INFORME FINAL DE TRABAJO DE GRADO</p> | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

2. MARCO TEÓRICO

2.1 Pantógrafo

Algunos de los grandes avances de las tecnologías actuales están basados en recrear comportamientos de la naturaleza o conocimientos de civilizaciones anteriores en desuso. Antes de la imprenta y la computación los conocimientos debían ser plasmados a mano y tendían a ser imprecisos especialmente cuando de graficas se trataba. Es allí donde aparece la importancia de la pantografía y su herramienta principal el pantógrafo. El pantógrafo de dibujo es un aparato de dibujo cuyo principio es usar una imagen guía a efectos de ampliarla. Generalmente usado en arquitectura, consta de un pivote y un cruce de varillas de madera o metal. El aparato se basa en el principio del que recibe el nombre, y consiste en un paralelogramo articulado que sirve para dibujar una figura homotética a una usada de referencia, teniendo como objeto la ampliación de un dibujo o geometría.

2.2 Ingeniería inversa

“La ingeniería inversa busca obtener información acerca de un objeto o sistema de referencia. Dicha información es obtenida bajo un proceso de análisis del objeto, y puesto que todo análisis hecho por la ingeniería inversa debe ser guiado, entonces la información obtenida debe ser específica con dos claros propósitos; esto es, por un lado, la información es referida al objeto inicial o de referencia o, mejor dicho, debe caracterizar al objeto y, por otro lado, la información guía a la síntesis hacia la obtención de un modelo del objeto.” (Jiménez López, et al., 2010)

Con el reciente avance de la tecnología de escaneo 3D existen múltiples soluciones para adquirir un escaneo de alta calidad de objetos ordinarios. Aun con el mejor escáner

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

disponible, el objeto debe ser escaneado desde múltiples puntos de vista y luego superpuesto en una figura coordinada para su procesamiento.

Los modelos CAD pueden ser representados como combinaciones de formas primitivas. La idea de utilizar representaciones paramétricas para hacer ingeniería a la inversa tiene bastante tiempo. Basada en el conocimiento del diseño CAD y los pasos de representación los trabajos tempranos empezaron las representaciones de frontera o parches paramétricos. (I. D. Lee, J. H. Seo, Y. M. Kim, J. Choi, S. Han & B. Yoo, 2019).

2.3 Pantógrafos 3D-4D

También existen pantógrafos 3D, nacidos del desarrollo informático con los que, los objetos dibujados en el plano, se pueden reproducir en volumen; aunque estos nuevos instrumentos podrían ser también considerados como un "torno" con mayor capacidad de reproducción, con el que se puede reproducir, por ejemplo, una cabeza humana, a escala, tomando la información de una base de datos digital procedente de un análisis métrico realizado con un equipo explorador infográfico (escáner) y sustituyendo el trabajo de un escultor. Esta técnica se utiliza frecuentemente para el modelado de objetos o piezas; la realización de maquetas de automóviles, diseños arquitectónicos, objetos publicitarios personalizados, etc.

2.4 CNC

El control numérico (CN) o control decimal numérico es un sistema de automatización de máquinas herramienta que son operadas mediante comandos programados en un medio de almacenamiento, en comparación con el mando manual mediante volantes o palancas. Las primeras máquinas de control remoto numérico se construyeron en los años 40 y 50 por el ingeniero John T. Parsons, basadas en las máquinas existentes con motores des modificados cuyos números se relacionan manualmente siguiendo las instrucciones dadas

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

en un microscopio de tarjeta perforada. Estos servomecanismos iniciales se desarrollaron rápidamente con los equipos analógicos y digitales. El abaratamiento y miniaturización de los procesadores ha generalizado la electrónica digital en los todos los tipos herramienta, lo que dio lugar a la denominación control decimal numérico, control numérico por computadora, control numérico por computador o control numérico computarizado (CNC), para diferenciarlas de las máquinas que no tenían computadora. En la actualidad se usa el término control numérico para referirse a este tipo de sistemas, con o sin computadora.

Para mecanizar una pieza se usa un sistema de coordenadas que especificarán el movimiento de la herramienta de corte. El sistema se basa en el control de los movimientos de la herramienta de trabajo con relación a los ejes de coordenadas de la máquina, usando un programa informático ejecutado por una computadora. En el caso de un torno, hace falta controlar los movimientos de la herramienta en dos ejes de coordenadas: el eje de las X para los desplazamientos longitudinales del carro y el eje de las Z para los desplazamientos transversales de la torre. En el caso de las fresadoras se controlan también los desplazamientos verticales, que corresponden al eje Y. Para ello se incorporan servomotores en los mecanismos de desplazamiento del carro y la torreta, en el caso de los tornos y en la mesa en el caso de la fresadora; dependiendo de la capacidad de la máquina, esto puede no ser limitado únicamente a tres ejes. (Wikipedia, 2021)

2.5 Celda de carga.

Son transductores electrónicos que transforman o trasladan de fuerza o peso a cambios de voltaje. Por lo que este cambio de voltaje produce en la instrumentación de salida una deflexión repetible o Indicación que puede ser calibrado directamente en términos de la carga aplicada a la celda. El principio de operación depende sobre la deflexión de filamento del Strain – Gages(Galgas extensiométricas), creando un cambio en su resistencia y por lo tanto un desbalance en el circuito puente. Como resultado, se tiene que para una señal de voltaje de entrada dada, el voltaje de salida del puente varía proporcionalmente con la carga. (Espinoza Esquivel, 1995)

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

3. METODOLOGÍA

ETAPAS DE REALIZACIÓN DEL PROYECTO

3.1 CONSULTA

Como fase preliminar del desarrollo del proyecto se desempeñó una fase de consulta de proyectos o desarrollos comerciales similares en forma o funcionamiento al proyecto formulado, utilizando como referentes especiales el trabajo realizado por el profesor Andrés García en el desarrollo de una maquina CNC casera, documentado mediante su canal de YouTube: El Profe García, también los planos y procesos que proveen en la web mediante la página de instructables, y como más fuerte influencia el desarrollo comercial patentado por Clone duplicators UK: Clone 4D.

3.2 DISEÑO

Como primera decisión para la fase de diseño, se escogió emplear ejes prismáticos para el diseño general del sistema, se usó además el recurso que se dispone en la institución (INVENTOR Professional) para el diseño de las piezas a utilizar para la creación de la máquina, posterior al diseño se realizó el ajuste para conseguir que las piezas diseñadas estuvieran ajustadas al contexto en el que se vive, llegando a un diseño final con piezas que se pudiesen conseguir en el mercado local.

Se planteo entonces una máquina de 3 ejes prismáticos controlada por un microcontrolador ATMEGA2560 sobre una placa de desarrollo Arduino MEGA, que contempla la cantidad de salidas que se requiere para emplear: 2 celdas de carga con su respectivo amplificador

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

HX711 para la lectura de fuerza/torque, 3 motores paso a paso NEMA17 con sus respectivos driver A4988, 6 finales de carrera NC, un sensor de corriente SCT013-100 y un relevo conmutador de marcha de fresadora, todo alimentado mediante una fuente conmutada de 120 W (12 V - 10 A) que provea la corriente y el voltaje necesario de funcionamiento a todos los componentes electrónicos.

Para el controlador se diseña entonces, un programa de control retroalimentado planar, en el cual se utilicen los pines análogos y digitales del Arduino mega para conocer el estado de la máquina con una filosofía de control similar a una máquina de estados, que contemple el contacto con la pieza patrón sin transgredir su geometría y una región de histéresis para que simultáneamente se recorte el material de exceso que posee la pieza de trabajo, mientras se reconoce la geometría de la pieza.

Durante el proceso se tienen en cuentas los siguientes estados posibles:

- a. Home: Es el estado en que la maquina retrae todos sus ejes a una posición conocida para comenzar su proceso de retiro de material.
- b. Reconocimiento de pieza: Al comenzar, la máquina es ignorante de la ubicación y dimensiones de las piezas en el espacio de trabajo, por tanto, debe realizar una calibración del mismo para ubicar al menos un sector de la pieza para comenzar el desbaste del material de exceso de la pieza a procesar.
- c. Barrido de capa plana: La situación ideal para la máquina de ejes prismáticos es que la pieza tuviese solo caras planas perpendiculares a la punta ruteadora, de modo que se dé un deslizamiento suave sobre la superficie.
- d. Esquivar obstáculo: A medida que las geometrías de los objetos son más complejas, se debe activar sistema de control y es aquí donde actúa el grueso del control al tener en cuenta distintos tipos de obstáculos:

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

- Huecos: En este caso se pierde el contacto con la pieza patrón, para este caso en particular se podrá elegir ignorar el hueco y continuar con el desbaste de la capa planar. O por el contrario continuar el contacto con la pieza y seguir bordeando su geometría. Esta decisión dependerá completamente de lo que el desee el operador.
 - Pendientes/montañas: En este caso se procura seguir la geometría manteniendo el contacto con la pieza patrón hasta volver a la posición en z correspondiente a la capa.
 - Muro: Este caso corresponde a las pendientes abruptas en la geometría con ángulos cercanos o superiores a los 90° que son muy difíciles de detectar por parte de los sensores (en el caso de caídas) o que pueden sobre pasar los límites de histéresis contemplados por el sistema. En estos casos se procura retroceder ligeramente para después subir o bajar hasta llegar al límite inferior de histéresis.
- e. Capa Completa: Cuando se llega al límite articular de alguno de los ejes prismáticos la maquina debe reconocer que no puede continuar avanzando en la misma dirección.
- f. Pieza completa: Cuando se llega al límite del espacio de trabajo de la máquina.

3.3 MANUFACTURA

En la etapa de manufactura como primera etapa, y tras la contingencia causada por la pandemia de la COVID-19, se hizo una búsqueda de campo de los componentes que se podían adquirir en el mercado nacional, dadas las falencias en despacho y transporte internacional, además de los sobrecostos generados en algunos materiales e insumos se tuvieron falencias respecto al diseño original. Se inicia la elección de los materiales para el despiece del prototipo a implementar. Se eligió madera de 9 mm para la realización de los paneles principales. Varillas roscadas de 3/8” para la realización de los ejes móviles, y se

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

usaron acoples, varillas, y rodamientos de 8 mm que son ampliamente utilizadas en el desarrollo de impresoras 3D y CNC de pequeña escala. Para realización de piezas específicas del diseño se usaron impresoras 3D presentes en la institución. Las uniones se fijaron con tornillería convencional de 5/32", para las piezas presentes en los ejes que propician los desplazamientos, se optó por usar acoples y carcasas de aluminio que se encuentran fácilmente en el contexto nacional y además son indicadas para la implementación para mecanismos como las CNC e impresoras 3D.

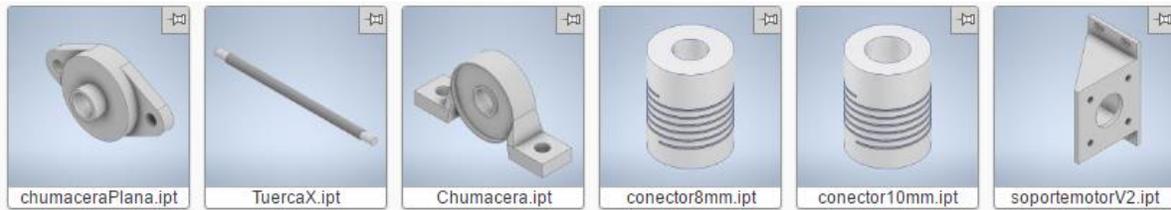


Figura 1. Piezas adquiridas en mercado nacional.

| | | | |
|---|--------------------------------------|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

4. RESULTADOS Y DISCUSIÓN

Se genera entonces el modelo CAD del prototipo a implementar y diversas versiones de este a medida que se realizaban modificaciones a los diseños por parte del docente asesor del proyecto.

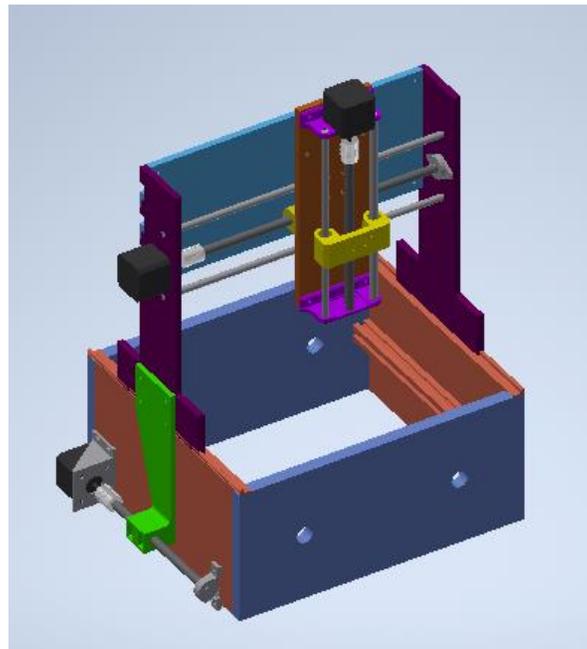


Figura 2. Ensamblaje CAD.

Para comenzar la implementación se hicieron pruebas de control a los distintos elementos que posee la máquina, también siguiendo varias recomendaciones dadas por el referenciado Erik García y las hojas de datos de los componentes no fabricados.

4.1. Prueba de sensores de carga.

Se investiga entonces la forma de obtener la magnitud del contacto de una pieza con un palpador sensor, encontrando en las celdas de carga una alternativa para sensor mediante la fuerza y el torque este contacto. Se emplean entonces dos pequeñas celdas de carga, que mediante un convertidor HX711 den una buena estimación de la fuerza que se ejerce en un sentido de movimiento. Para probar este hecho se realizó una prueba en la cual se obtuviesen los valores de dichos sensores mediante un microcontrolador con la misma

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

arquitectura del controlador a utilizar (En este caso Arduino UNO y Arduino MEGA respectivamente). Comprobando así que estos sensores tenían cambios proporcionales al poner o quitar carga en un extremo.

4.2. Prueba de motores.

Seguidamente se realizó la configuración de los motores siguiendo en este caso la configuración del driver controlador de motores descrita por Erik García y que se aparece en la hoja de datos del Driver A4988 de Allegro, donde se especifica el valor del voltaje de referencia que debe tener el potenciómetro del chip para proveer la corriente necesaria del motor, que en este caso dado que los motores a usar son NEMA17 es de 1.5 A se sigue la siguiente fórmula:

$$I_{TripMAX} = V_{REF} / (8 \times R_S)$$

Donde, $I_{TripMAX}$ es la corriente del motor, V_{REF} es el voltaje de referencia en el potenciómetro, y R_S es la resistencia de sensado del chip. Se calibra como se describe en el video de Erik García:

Teniendo en cuenta que se va a trabajar en el modo de Full-step y por tanto se debe tener como referencia la siguiente grafica de la hoja de datos.

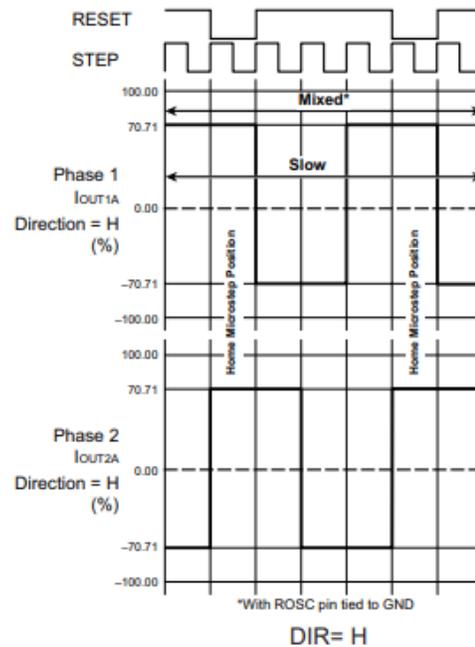


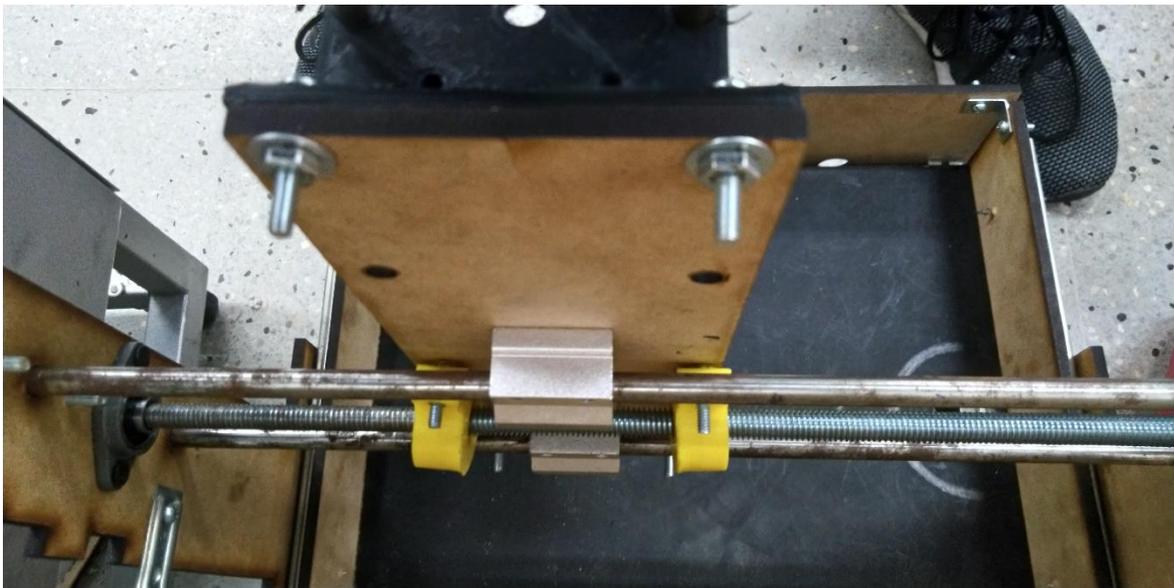
Figura 4. Comportamiento motor paso a paso Full-step.

De donde se concluye que para este modo se requiere el 70% de la corriente. Posteriormente se procedió a realizar control simultaneo de los motores por medio de la librería StepperDriver del A4988 disponible en el repositorio de Arduino.

4.3 Implementación de Mecanismos.

Ya diseñado, se procede a la implementación del prototipo CAD a partir de las piezas adquiridas en el mercado nacional, y con ayuda de partes impresas mediante impresora 3D para la creación de piezas de propósito específico para la aplicación.

| | | | |
|--|--------------------------------------|---------|------------|
|  Institución Universitaria | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |



Figuras 5,6 y 7. Implementación del mecanismo real.

Además, para integrar todos los componentes del circuito electrónico que se requiere, se diseñó en la plataforma de Autodesk EAGLE una placa de circuito impreso que contemple

todos los componentes para la implementación. Dado que las placas disponibles en el mercado son compatibles con software de propósito específico como lo es GRBL para el ruteado vectorizado y la impresión 3D.

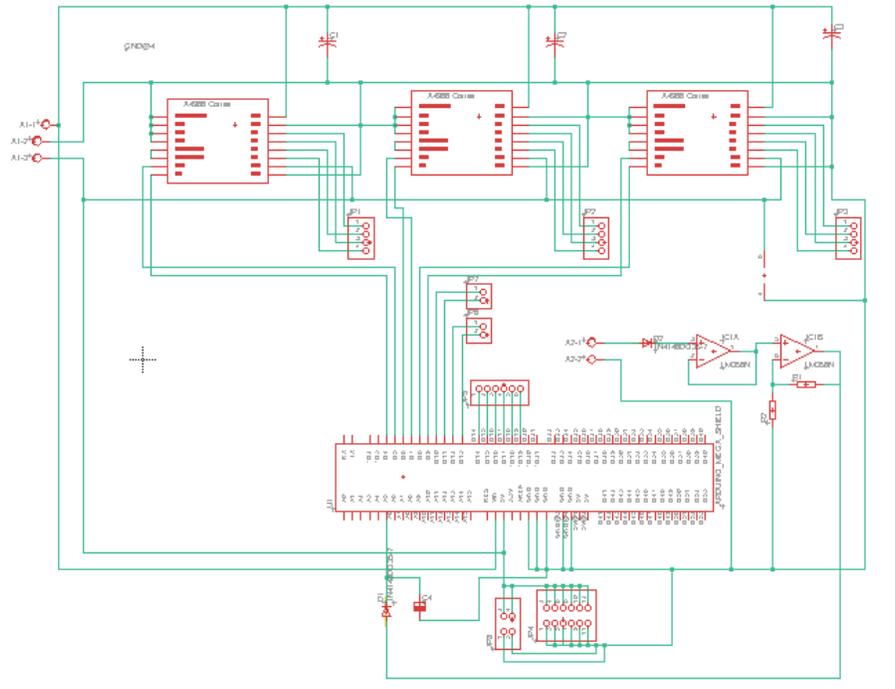


Figura 8. Circuito lógico diseñado

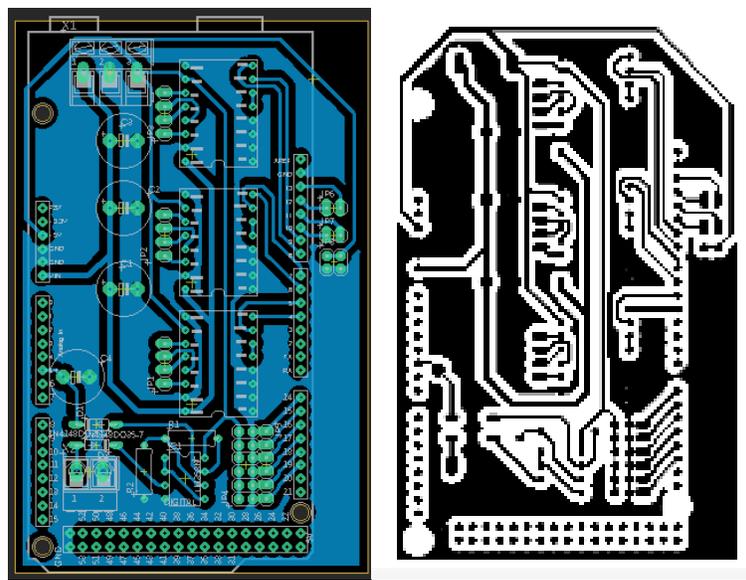


Figura 9. Circuito impreso diseñado

| | | | |
|---|--------------------------------------|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO

- Se logró diseñar e implementar una ruteadora 4D con ejes controlados automáticamente y un sistema que le provee datos para el ruteado de geometrías medianamente complejas.
- Se diseñaron, además, los componentes de dicho mecanismo partiendo de suposiciones razonables.
- Se diseñó además un sistema de control que integre el funcionamiento de los componentes mecánicos y electrónicos de la maquina para un optimo desempeño de su función.
- Se logró la implementación de un PCB de diseño propio que implementa la cantidad de salidas requeridas en una ubicación optima para el desarrollo de la aplicación deseada.
- En general se concluye que se requiere mayor desarrollo en sistemas que puedan realizar trabajos que requieren de algún tipo de presión y/o contacto, y al mismo tiempo la sensibilidad y repetibilidad para la realización de procesos ágiles y cuidadosos.
- En el futuro se debería propender por la realización de sistemas robustos que controlen suavemente altas cargas de trabajo y sean sensibles a los leves cambios.

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

REFERENCIAS

- Clone Duplicators. (16 de Mayo de 2021). About: Clone Duplicators. Obtenido de Clone Duplicators : <https://cloneduplicators.uk/>
- García, E. (Dirección). (2016). CNC Facil de hacer en Casa [Película]. Obtenido de https://www.youtube.com/watch?v=X4BLydtpllo&list=PLnwu2s7SlakR0Gs5vAO_1sfWDOy9mBEO
- Glosario Gráfico. (s.f.). Pantógrafo. Obtenido de Glosario gráfico: <http://www.glosariografico.com/pantografo>
- Instructables. (2021). Arduino Controlled CNC 3D Printer. Obtenido de Instructables: <https://www.instructables.com/Arduino-Controlled-CNC-3D-Printer/>
- Jiménez López, E., Luna Bracamontes, A., García Velázquez, L., Martínez Molina, V., Luna Sandoval, G., Delfín Vázquez, J., & Arellano Rivera, L. (2010). La ingeniería inversa como metodología para potenciar la enseñanza de la metrología. Simposio de Metrología 2010 (págs. 1-8). Sonora, Mexico: CINNTRA.
- Quatrano, A., De Simone, M. C., Rivera, Z. B., & Guida, D. (2017). Development and Implementation of a. FME Transactions, 565-571. doi:10.5937/fmet1704565Q Wikipedia. (11 de Marzo de 2021). Control Numérico. Obtenido de Wikipedia la enciclopedia libre: https://es.wikipedia.org/wiki/Control_num%C3%A9rico
- Golnaraghi, F. , Kuo, B. C., "Automatic Control Systems", 10ª ed., McGraw Hill, 2017 Yu Yang, Wei Shengmin and Liu Ping, "A Universal Kinesiology modeling of multi-axis CNC machine," 2011 Second International Conference on Mechanic Automation and Control Engineering, 2011, pp. 45-48, doi: 10.1109/MACE.2011.5986853.
- I. D. Lee, J. H. Seo, Y. M. Kim, J. Choi, S. Han and B. Yoo, "Automatic Pose eneration for Robotic 3-D Scanning of Mechanical Parts," in IEEE Transactions on Robotics, vol. 36, no. 4, pp. 1219-1238, Aug. 2020, doi: 10.1109/TRO.2020.2980161. Z. Yu, T. Wang, P. Wang, Y. Tian and H. Li, "Rapid and Precise Reverse Engineering of Complex Geometry Through Multi-Sensor Data Fusion," in IEEE Access, vol. 7, pp. 165793-165813, 2019, doi: 10.1109/ACCESS.2019.2948124.

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

- Espinoza Esquivel, M. (1995). *Diseño y construcción de una celda de carga*. San Nicolás de los Garza: Universidad autónoma de Nuevo Leon.

APÉNDICE

A. Código de controlador

/*

Proyecto de grado: Ruteadora 4D automatizada

Modalidad: Producto de laboratorio

Tutor: Jorge Andrés Sierra del Rio

Por: John Camilo Giraldo Lopez

Estudiante de Tecnología en sistemas electromecánicos

Departamento de Mecatrónica y Electromecánica

Instituto tecnológico Metropolitano

Medellín, Antioquia, Colombia

Junio 2022

*/

// Librerías a usar

#include <Arduino.h>

#include "BasicStepperDriver.h" // Librería de Laurentiu Badea- Stepper Driver ver
ejemplo de MultiAxis

#include "MultiDriver.h"

#include "SyncDriver.h"

#include <HX711_ADC.h> // Librería de Olav - ADC for multiple load cells

#if defined(ESP8266) || defined(ESP32) || defined(AVR)

#include <EEPROM.h>

| | | | |
|---|--------------------------------------|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

#endif

//-----MOTORES-----

// Motor steps per revolution. Most steppers are 200 steps or 1.8 degrees/step}

// Pasos por vuelta del motor.

#define MOTOR_STEPS 200

// Target RPM for X axis motor

// Velocidad para el motor del eje X

#define MOTOR_X_RPM 150

// Target RPM for Y axis motor

// Velocidad para el motor del eje Y

#define MOTOR_Y_RPM 150

// Target RPM for Y axis motor

// Velocidad para el motor del eje Z

#define MOTOR_Z_RPM 150

// X motor

#define DIR_X 8

#define STEP_X 9

// Y motor

#define DIR_Y 7

#define STEP_Y 6

// Z motor

#define DIR_Z 4

#define STEP_Z 5

| | | | |
|--|--------------------------------------|---------|------------|
|  Institución Universitaria | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```

// If microstepping is set externally, make sure this matches the selected mode
// 1=full step, 2=half step etc.
#define MICROSTEPS 2

// Defining position maximum and minimum
#define MAXX 1000
#define MAXY 1500
#define MAXZ 1200
#define MINX 0
#define MINY 0
#define MINZ 0

// 2-wire basic config, microstepping is hardwired on the driver
// Other drivers can be mixed and matched but must be configured individually
BasicStepperDriver stepperX(MOTOR_STEPS, DIR_X, STEP_X);
BasicStepperDriver stepperY(MOTOR_STEPS, DIR_Y, STEP_Y);
BasicStepperDriver stepperZ(MOTOR_STEPS, DIR_Z, STEP_Z);

// Pick one of the two controllers below
// each motor moves independently, trajectory is a hockey stick
MultiDriver controller(stepperX, stepperY, stepperZ);

// OR
// synchronized move, trajectory is a straight line
//SyncDriver controller(stepperX, stepperY, stepperZ);

//-----SENSORES DE CARGA-----
-----

const int HX711_dout_1 = 2; //mcu > HX711 no 1 dout pin
const int HX711_sck_1 = 3; //mcu > HX711 no 1 sck pin
const int HX711_dout_2 = 10; //mcu > HX711 no 2 dout pin
const int HX711_sck_2 = 11; //mcu > HX711 no 2 sck pin

```

| | | | |
|--|--------------------------------------|---------|------------|
|  Institución Universitaria | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```
//HX711 constructor (dout pin, sck pin)
```

```
HX711_ADC LoadCell_1(HX711_dout_1, HX711_sck_1); //HX711 1
```

```
HX711_ADC LoadCell_2(HX711_dout_2, HX711_sck_2); //HX711 2
```

```
const int calVal_eepromAdress_1 = 0; // eeprom adress for calibration value load cell 1 (4 bytes) direccion de la memoria interna para la calibración de la celda 1
```

```
const int calVal_eepromAdress_2 = 4; // eeprom adress for calibration value load cell 2 (4 bytes) direccion de la memoria interna para la calibración de la celda 2
```

```
unsigned long t = 0;
```

```
const float refA = 1.3;
```

```
const float refB = 1.5;
```

```
const float refChoque=200.0;
```

```
int SENTIDOX=1;
```

```
int SENTIDOY=1;
```

```
int SENTIDOZ=1;
```

```
const int AVANCEX = 100;
```

```
const int AVANCEY = 150;
```

```
const int AVANCEZ = 120;
```

```
bool patron=false;
```

```
bool initial=false;
```

```
float a=0.0;
```

```
float b=0.0;
```

```
int PAX=10;
```

```
int PAY=10;
```

```
int PAZ=10;
```

```
int PDZ=0;
```

```
//-----FINALES DE CARRERA-----
```

```
-----
```

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```

#define B_X 14 // Backward X
#define F_X 15 // Forward X
#define B_Y 16
#define F_Y 17
#define B_Z 18
#define F_Z 19

void setup() {
//-----Iniciación FINALES DE CARRERA-----
-----

pinMode(B_X, INPUT); // Estados HIGH y LOW para ambos
pinMode(B_Y, INPUT);
pinMode(B_Z, INPUT);
pinMode(F_X, INPUT);
pinMode(F_Y, INPUT);
pinMode(F_Z, INPUT);

//-----Iniciación MOTORES-----
Serial.begin(57600); delay(10); // inicializamos el puerto serie a 57600 baudios
stepperX.begin(MOTOR_X_RPM, MICROSTEPS); // Se inicializan los motores
stepperY.begin(MOTOR_Y_RPM, MICROSTEPS);
stepperZ.begin(MOTOR_Z_RPM, MICROSTEPS);

//-----Iniciación SENSORES de Carga-----
-----

Serial.println(); // Imprimimos línea vacía
Serial.println("Starting...");

```

| | | | |
|---|--------------------------------------|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```

float calibrationValue_1; // calibration value load cell 1
float calibrationValue_2; // calibration value load cell 2

//calibrationValue_1 = 696.0; // uncomment this if you want to set this value in the
sketch

//calibrationValue_2 = 733.0; // uncomment this if you want to set this value in the
sketch

#if defined(ESP8266) || defined(ESP32)
  //EEPROM.begin(512); // uncomment this if you use ESP8266 and want to fetch the
value from eeprom
#endif

//EEPROM.get(calVal_eepromAdress_1, calibrationValue_1); // uncomment this if you
want to fetch the value from eeprom

//EEPROM.get(calVal_eepromAdress_2, calibrationValue_2); // uncomment this if you
want to fetch the value from eeprom

LoadCell_1.begin(); // Iniciamos la celda de carga 1
LoadCell_2.begin(); // Iniciamos la celda de carga 2

unsigned long stabilizingtime = 2000; // tare precision can be improved by adding a few
seconds of stabilizing time

boolean _tare = true; //set this to false if you don't want tare to be performed in the next
step

byte loadcell_1_rdy = 0;
byte loadcell_2_rdy = 0;

while ((loadcell_1_rdy + loadcell_2_rdy) < 2) { //run startup, stabilization and tare, both
modules simultaneously

  if (!loadcell_1_rdy) loadcell_1_rdy = LoadCell_1.startMultiple(stabilizingtime, _tare);
  if (!loadcell_2_rdy) loadcell_2_rdy = LoadCell_2.startMultiple(stabilizingtime, _tare);
}

```

| | | | |
|--|--------------------------------------|---------|------------|
|  Institución Universitaria | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```

if (LoadCell_1.getTareTimeoutFlag()) {
  Serial.println("Timeout, check MCU>HX711 no.1 wiring and pin designations");
}
if (LoadCell_2.getTareTimeoutFlag()) {
  Serial.println("Timeout, check MCU>HX711 no.2 wiring and pin designations");
}
//LoadCell_1.setCalFactor(calibrationValue_1); // user set calibration value (float)
//LoadCell_2.setCalFactor(calibrationValue_2); // user set calibration value (float)
Serial.println("Startup is complete");

}

void loop() {
  //Análisis de casos
  //Caso 1: Volver a casa-> Retraer totalmente Z, LUEGO retraer a Y y retraer a X a un final
  de carrera definido
  while (initial==false){
    while(digitalRead(B_Z==HIGH)){
      controller.rotate(0, 0,-90);
      delay(10);
    }
    PAZ=0;//Posición Actual Z
    while(digitalRead(B_X==HIGH)){
      controller.rotate(-90, 0,0);
      delay(10);
    }
    PAX=0;//Posición Actual X
  }
}

```

| | | | |
|--|--------------------------------------|---------|------------|
|  Institución Universitaria | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```

while(digitalRead(B_Y==HIGH)){
  controller.rotate(0, -90,0);
  delay(10);
}
PAY=0;//Posición Actual Y
initial=true; //Ya iniciado el proceso automatico desde (0,0,0)
}

//Caso 2: NO HAY contacto ni con la pieza, ni con el bloque pasado el eje de referencia->
Desplazar en X hasta que se detecte contacto en alguno de los dos (Primera vez), sino baja
Z 2 milímetros invertir dirección de X y repetir
//CREAR VARIABLES PARA LA POSICION EN CADA EJE

//if (SensorCorriente<refC){ //Caso 3: Hay contacto de herramienta de corte, pero aun no
se localiza la pieza patrón en Y o Z -> realizar AVANCE uniforme (Desbaste) a baja
velocidad, hasta encontrar patrón
//
contact();

while (initial==true and patron==false and a<refA and b<refB){//buscar pieza patrón
(mientras se busca la pieza patron también se desbasta el bloque de material)
  controller.rotate(0,SENTIDOY*AVANCEY,0); // Empezar barrido en Y
  PAY=PAY+SENTIDOY*AVANCEY;//Actualizar posicion Actual en Y
  if( PAY> MAXY or PAY<MINY){ // Limites de Y
    SENTIDOY=-SENTIDOY; // Si se llega a un limite se debe invertir el SENTIDO de
AVANCE
    controller.rotate(SENTIDOX*AVANCEX,SENTIDOY*AVANCEY,0);// Además de invertir
el SENTIDO se debe avanzar en otra dirección (X) para cubrir el área de una superficie

```

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```

PAY=PAY+SENTIDOY*AVANCEY; // Actualizamos la posicion en Y
PAX=PAX+SENTIDOX*AVANCEX ;// Actualizamos la posicion en X
if(PAX>=MAXX or PAX<=MINX){ // Si llegamos al límite en X
    SENTIDOX=-SENTIDOX; //Invertir el giro en X
    controller.rotate(SENTIDOX*AVANCEX,SENTIDOY*AVANCEY,SENTIDOZ*AVANCEZ);
    PAY=PAY+SENTIDOY*AVANCEY;
    PAX=PAX+SENTIDOX*AVANCEX;
    PAZ=PAZ+SENTIDOZ*AVANCEZ;
    if(PAZ>MAXZ){
        Serial.println("No se encontró pieza ni patrón. Reiniciando...");
        initial=false;
    }
}
}
contact();
}

if(initial=true and patron==false and (a>refA or b>refB)){//Pieza patrón encontrada!!
    patron=true;
    PDZ=PAZ;
}

//Caso 4: Hay contacto de herramienta de corte, y la pieza patrón es plana perpendicular
a la herramienta de corte -> Sostener control para tener contacto en Z sin aumentar el
contacto en Y

if(patron==true and (a>refA or b>refB)){
    float vector=sqrt(a*a+b*b);

```

| | | | |
|---|--------------------------------------|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

if (vector>refChoque){//Caso 5: Hay contacto de herramienta de corte, y la pieza patrón es una curva ascendente-> Sostener control para tener contacto con la pieza patrón en Z sin aumentar el contacto en Y

```
controller.rotate(0,0,-SENTIDOZ*AVANCEZ); // Empezar Z
```

```
PAZ=PAZ-SENTIDOZ*AVANCEZ;
```

```
}
```

```
else{
```

```
controller.rotate(0,SENTIDOY*AVANCEY,0);
```

```
PAY=PAY+SENTIDOY*AVANCEY;
```

```
}
```

```
if( PAY >= MAXY or PAY<=MINY){ // Limites de Y
```

```
SENTIDOY=-SENTIDOY; // Si se llega a un limite se debe invertir el SENTIDO de
```

AVANCE

```
controller.rotate(SENTIDOX*AVANCEX,SENTIDOY*AVANCEY,0);// Además de invertir
```

el SENTIDO se debe avanzar en otra dirección (X) para cubrir el área de una superficie

```
PAY=PAY+SENTIDOY*AVANCEY; // Actualizamos la posición en Y
```

```
PAX=PAX+SENTIDOX*AVANCEX; // Actualizamos la posición en X
```

```
if(PAX>MAXX or PAX<MINX){ // Si llegamos al límite en X
```

```
SENTIDOX=-SENTIDOX; //Invertir el giro en X
```

```
controller.rotate(SENTIDOX*AVANCEX,SENTIDOY*AVANCEY,SENTIDOZ*AVANCEZ);//
```

Además de invertir el SENTIDO se debe avanzar en otra dirección (Z) para cubrir el área de una superficie

```
PAY=PAY+SENTIDOY*AVANCEY;
```

```
PAX=PAX+SENTIDOX*AVANCEX;
```

```
PAZ=PAZ+SENTIDOZ*AVANCEZ;
```

```
if(PAZ>MAXZ){
```

```
Serial.println("No se encontró pieza ni patrón. Reiniciando...");
```

```
initial=false;
```

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```

    }
  }
}
}
else if(patron==true and (a<refA and b<refB)){ // No se está tocando la pieza patrón
  if(PAZ==PDZ){//si la posicion actual en Z corresponde con la capa que se está
desvastando, entonces continúe el desbaste sobre dicha capa
  controller.rotate(0,SENTIDOY*AVANCEY,0);
  PAY=PAY+SENTIDOY*AVANCEY;
  if(PAY >= MAXY or PAY <= MINY){ // Limites de Y
    SENTIDOY=-SENTIDOY; // Si se llega a un limite se debe invertir el SENTIDO de
AVANCE
    controller.rotate(SENTIDOX*AVANCEX,SENTIDOY*AVANCEY,0);// Además de invertir
el SENTIDO se debe avanzar en otra dirección (X) para cubrir el área de una superficie
    PAY=PAY+SENTIDOY*AVANCEY; // Actualizamos la posición en Y
    PAX=PAX+SENTIDOX*AVANCEX;// Actualizamos la posición en X
    if(PAX>MAXX or PAX<MINX){ // Si llegamos al límite en X
      SENTIDOX=-SENTIDOX; //Invertir el giro en X

controller.rotate(SENTIDOX*AVANCEX,SENTIDOY*AVANCEY,SENTIDOZ*AVANCEZ);//
Además de invertir el SENTIDO se debe avanzar en otra dirección (Z) para cubrir el área de
una superficie
    PAY=PAY+SENTIDOY*AVANCEY;
    PAX=PAX+SENTIDOX*AVANCEX;
    PAZ=PAZ+SENTIDOZ*AVANCEZ;
    PDZ=PAZ;
    if(PAZ>MAXZ){
      Serial.println("Proceso terminado") ;

```

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```

    initial=false;
  }
}
}
}
else{
  controller.rotate(0,0,int((PDZ-PAZ)*0.1)); // Empezar Z
  PAZ=PAZ+int((PDZ-PAZ)*0.1);
}
}

//Caso 6: Hay contacto de herramienta de corte, y la pieza patrón es una curva
descendente-> Sustener control para tener contacto con la pieza patrón en Z sin aumentar
el contacto en Y

//Caso 7: Hay contacto de herramienta de corte, y la pieza patrón tiene un escalón
ascendente-> Subir el eje hasta que disminuya el indicador de contacto Y bajo el límite
crítico

//Caso 8: Hay contacto de herramienta de corte, y la pieza patrón tiene un escalón
descendente-> Bajar el eje hasta la altura a la que inicio la pasada para no esforzar la
profundidad de corte

//Caso 9: NO HAY contacto de herramienta de corte, pero si con la pieza patrón -> El
Bloque fue cortado satisfactoriamente O El bloque no cubría las dimensiones minimas
(Acciones casos 4-8)

//Caso 10: Final de carrerera de Y -> Avanzar en X y cambiar la dirección de Y
//Caso 11: Finales de carrera X y Y -> Bajar en Z, Cambiar la dirección de X y realizar
nueva pasada

//Caso 12: Finales de carrera X,Y y Z -> Alarma de Proceso terminado en una cara,
Reubicar la pieza para más procesar otra Cara o Terminar(Volver a Home)
}

```

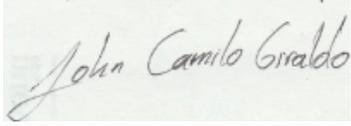
| | | | |
|--|--------------------------------------|---------|------------|
|  Institución Universitaria | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |

```

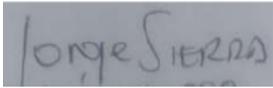
void contact(){
  boolean newDataReady = 0;
  const int serialPrintInterval = 0; //increase value to slow down serial print activity
  // check for new data/start next conversion:
  if (LoadCell_1.update()) newDataReady = true;
  LoadCell_2.update();
  //get smoothed value from data set
  if ((newDataReady)) {
    if (millis() > t + serialPrintInterval) {
      a = LoadCell_1.getData();
      b = LoadCell_2.getData();
      // Serial.print("Load_cell 1 output val: ");
      // Serial.print(a);
      // Serial.print("  Load_cell 2 output val: ");
      // Serial.println(b);
      newDataReady = 0;
      t = millis();
    }
  }
}

```

| | | | |
|---|--|---------|------------|
|  | INFORME FINAL DE TRABAJO DE GRADO | Código | FDE 089 |
| | | Versión | 03 |
| | | Fecha | 2015-01-22 |



FIRMA ESTUDIANTES _____



FIRMA ASESOR _____

FECHA ENTREGA: 07/06/2022

FIRMA COMITÉ TRABAJO DE GRADO DE LA FACULTAD _____

RECHAZADO___ ACEPTADO___ ACEPTADO CON MODIFICACIONES___

ACTA NO. _____

FECHA ENTREGA: _____

FIRMA CONSEJO DE FACULTAD _____

ACTA NO. _____

FECHA ENTREGA: _____