

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

INGENIERÍA EN ELECTRÓNICA Y CONTROL

**PROGRAMACIÓN E IMPLEMENTACIÓN DE UN PANTÓGRAFO
CNC PARA OXICORTE MEDIANTE EL USO DE MATLAB
BASADO EN LA IMPORTACIÓN DE UN ARCHIVO EN FORMATO
GRÁFICO**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA Y CONTROL**

DIEGO MANUEL GUALLICHICO HIDALGO
gualidiegomh@hotmail.com

CHRISTIAN ANDRÉS ZAMBRANO SANTAMARÍA
c.andres.z@hotmail.es

DIRECTOR: DR. JORGE ANDRÉS ROSALES ACOSTA
androsaco@yahoo.com

Quito, agosto 2012

DECLARACIÓN

Nosotros, Diego Manuel Guallichico Hidalgo y Christian Andrés Zambrano Santamaría, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Diego Manuel Guallichico Hidalgo

Christian Andrés Zambrano Santamaría

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Diego Manuel Guallichico Hidalgo y Christian Andrés Zambrano Santamaría, bajo mi supervisión.

Dr. Andrés Rosales

DIRECTOR DEL PROYECTO

AGRADECIMIENTO

Diego Guallichico

Al Doctor Andrés Rosales, quien dirigió este proyecto de titulación.

Al señor Marco Zambrano, por haber puesto su confianza y brindado su apoyo durante la realización y definición de este proyecto.

Christian Zambrano

A mis padres por haberme abrigado bajo su manto en este sendero de mi vida, a mis profesores por transmitirme su conocimiento y a mis amigos por cada momento compartido.

DEDICATORIA

Diego Guallichico

A nuestro Padre creador, por habernos brindado oportunidades cada día, lecciones y la sabiduría para elegir el mejor sendero.

A mis padres Manuel y Aurora, por ser el principal pilar e inspiración durante este camino; a mis hermanos, sobrinos y amigos quienes con sus ánimos forman parte de este logro.

Christian Zambrano

A mi madre por ser la luz de mis ojos.

A mi padre por su apoyo y fortaleza.

CONTENIDO

CONTENIDO.....	v
RESUMEN.....	viii
PRESENTACIÓN.....	ix

CAPÍTULO 1 MARCO TEÓRICO 1

1.1	INTRODUCCIÓN.....	1
1.2	EXPLICACIÓN GENERAL DEL PROCESO	2
1.3	CONTROL NUMÉRICO COMPUTARIZADO	3
1.3.1	DEFINICIÓN	3
1.3.2	CARACTERÍSTICAS DEL CONTROL NUMÉRICO COMPUTARIZADO	3
1.3.3	MÁQUINAS HERRAMIENTAS DE CONTROL NUMÉRICO	4
1.3.3.1	Tipos De Máquina Herramienta	5
1.3.3.2	Principales Máquinas Herramientas	5
1.3.3.3	Elementos De Una Máquina Herramienta De Control Numérico.....	8
1.3.3.4	Principios De Mecanizado En Control Numérico	9
1.3.3.5	Ejes De Coordenadas En Control Numérico	9
1.4	UTILIZACIÓN DE MATLAB	11
1.4.1	PROCESAMIENTO DE IMÁGENES	11
1.4.1.1	Herramienta Imtool	13
1.4.2	UTILIZACIÓN DE DAQ.....	16
1.4.2.1	Características Generales.....	16
1.4.2.2	Compatibilidad De Daq 6501 Con Pc	17
1.4.2.3	Compatibilidad De Daq 6501 Con Matlab	19
1.4.2.4	Datos Que Maneja La Daq	20
1.4.3	HERRAMIENTA GUIDE	21
1.4.3.1	Descripción De Los Principales Componentes De Una Ventana Gui.....	24

CAPÍTULO 2 DESCRIPCIÓN DEL ALGORITMO Y MÉTODO DE CONTROL..... 28

2.1	DESCRIPCIÓN GENERAL	28
2.2	IMPORTACIÓN DE IMÁGENES	29
2.2.1	CREACIÓN DE IMÁGENES PREDISEÑADAS	31

2.2.1.1	Análisis De Imágenes Mediante La Herramienta Imtool.....	34
2.2.2	IDENTIFICACIÓN DE CONTORNOS.....	36
2.2.3	ALGORITMO DE CONTEO DE PERÍMETROS EXISTENTES	39
2.2.4	GENERACIÓN DE MATRICES DE PERÍMETROS.....	42
2.2.5	GENERACIÓN DE VECTORES INDIVIDUALES DE CADA CONTORNO EXISTENTE EN LA FIGURA.....	45
2.2.6	GRÁFICA DE PERÍMETROS.....	48
2.3	GENERACIÓN DE CÓDIGO G	50
2.4	CONTROL DE MOTORES.....	65
2.4.1	DESCRIPCIÓN DEL HARDWARE DE INTERFAZ.....	66
2.4.1.1	Acondicionamiento De Señal De Encoders.....	67
2.4.1.2	Realimentación Hacia Daq6501	72
2.4.2	ALGORITMO DE CONTROL DE MOTORES	79
2.5	CREACIÓN DE ARCHIVO EJECUTABLE, ARCHIVO .EXE.....	82
2.6	INSTALACIÓN DE ARCHIVO EJECUTABLE	83

CAPÍTULO 3 DISEÑO DEL SISTEMA Y MONTAJE..... 87

3.1	DESCRIPCIÓN GENERAL DEL PROCESO.....	87
3.1.1	MECANISMO TORNILLO TUERCA.....	87
3.1.2	TORNILLOS SIN FIN	88
3.1.3	ACOPLAMIENTO DE PIÑONES.....	89
3.1.4	EJES DE MOVIMIENTO	91
3.2	DISEÑO DEL SISTEMA MECÁNICO	92
3.2.1	PRIMER PROTOTIPO MECÁNICO	93
3.2.2	SEGUNDO PROTOTIPO MECÁNICO	96
3.2.3	TERCER PROTOTIPO MECÁNICO.....	98
3.2.4	CUARTO PROTOTIPO MECÁNICO.....	100
3.2.4.1	Fabricación De Mesa.....	100
3.2.4.2	Incorporación Del Eje Longitudinal	102
3.2.4.3	Incorporación Del Eje Transversal.....	105
3.2.5	DESPLAZAMIENTO EN EJES LONGITUDINAL Y TRANSVERSAL	108
3.2.6	SISTEMA DE MONTAJE PARA MOTORES.....	113
3.2.6.1	Soporte Para El Motor.....	114
3.2.6.2	Base Para El Motor.....	114
3.2.7	SISTEMA DE MONTAJE PARA ENCODERS.....	116
3.2.8	SOPORTE PARA SOPLETE DE MÁQUINA OXICORTE	118

CAPÍTULO 4 PRUEBAS Y ANÁLISIS DE RESULTADOS ... 121

4.1	SISTEMA DE ARRASTRE PARA LA ANTORCHA	121
4.1.1	RIEL EN “V” CON RUEDA EN “V”	121
4.1.2	TUBO CIRCULAR CON RODAMIENTOS.....	122
4.1.3	TUBO CUADRADO CON RODAMIENTOS.....	123
4.1.4	CORRECCIÓN DE AVANCE MUERTO	125
4.2	ACONDICIONAMIENTO DE ENCODERS INCREMENTALES	126
4.2.1	OBTENCIÓN DE LA SEÑAL.....	126
4.2.2	CIRCUITO DE ANCONDICIONAMIENTO.....	127
4.2.3	VELOCIDAD DE GIRO DEL ENCODER.....	130
4.2.3.1	Primer Acoplamiento	130
4.2.3.2	Segundo Acoplamiento	133
4.3	REALIMENTACIÓN HACIA DAQ6501	135
4.3.1	TIEMPO DE RESPUESTA DEL PROGRAMA CON ANIMACIÓN DEL PROCESO ...	135
4.3.2	REDUCCIÓN DE CICLOS DE MÁQUINA	136
4.4	CONTROL DE LA TRAYECTORIA DE CORTE	138
4.4.1	TRAYECTORIA DIAGONAL	138
4.4.2	TRAYECTORIA EN ESCALONES.....	139
4.5	CALIBRACIÓN DEL SISTEMA	141
4.5.1	PERPENDICULARIDAD ENTRE EJES	142
4.5.2	DETECCIÓN DEL DESPLAZAMIENTO DE 1 mm.....	143

CAPÍTULO 5 CONCLUSIONES Y RECOMENDACIONES .. 145

5.1	CONCLUSIONES.....	145
5.2	RECOMENDACIONES	147

REFERENCIAS BIBLIOGRÁFICAS 149

ANEXOS

RESUMEN

Este proyecto de titulación trata la implementación de un pantógrafo para una antorcha oxicorte, así como su automatización mediante el uso de un software de control CNC (Control Numérico Computarizado) desarrollado en MATLAB, que mediante la importación de un archivo en formato gráfico, previamente diseñado, reproducirá en una plancha de metal de grosor considerable por medio de una herramienta oxicorte. Los archivos gráficos son obtenidos mediante el uso de un software que convierte archivos DWF (pertenecientes a una plataforma CAD) a un formato gráfico como png o jpg.

El proyecto contempla el diseño de un sistema microprocesado para el control de los motores de DC con caja reductora, que brinda el movimiento necesario a los ejes transversal y longitudinal del pantógrafo; también se realiza el diseño de interfaces electrónicas para comunicar la PC con el microcontrolador, por medio de una tarjeta de adquisición de datos DAQ6501 de la NI (National Instruments), así como el diseño de la etapa de acondicionamiento de las señales de los encoders, ubicados en cada eje, para detectar el desplazamiento.

Como parte del sistema CNC se desarrolla una Interfaz Hombre-Máquina (HMI), la cual se encarga de realizar la importación del archivo a reproducir con el oxicorte, mostrar las trayectorias que la antorcha recorre, enviar los datos hacia el microcontrolador mediante la configuración de la DAQ6501, y permitir el inicio del proceso que abarca el posicionamiento y el corte como tal. Cuenta además con un botón virtual de paro de emergencia y con cuatro pulsadores para el reposicionamiento o empleo del oxicorte sin el uso de una PC, es decir, netamente para cortes rectos.

Asimismo, el proyecto contempla la implementación de un tablero general que contiene fuentes de alimentación para los sistemas de control y potencia, circuitos de control, circuitos de potencia, circuitos acondicionadores de encoders y las interfaces con el pantógrafo.

PRESENTACIÓN

La tecnología actual brinda muchas ventajas y soluciones a diversos procesos, pero en cuanto al sector industrial ha avanzado considerablemente en estos últimos años brindando maquinarias y componentes que, en muchos casos, incluso llegan a remplazar a un operador, optimizando al proceso, reduciendo costos y elevando la productividad. Esta evolución tecnológica aún no se vive con énfasis en nuestro país, debido a la gran inversión que necesita. Sin embargo, algunas industrias han empezado a automatizarse, lo que ha llevado a la mejora de servicios y por ende a la competencia, es por esto que muchos sectores han tomado como una solución la automatización de algunas etapas de sus procesos, para en un futuro lograr incorporar totalmente nuevas tecnologías.

Una opción de automatización de un proceso es una máquina CNC, que como principales ventajas brinda precisión, rapidez y sobre todo, incremento de la productividad.

El presente proyecto está distribuido de la siguiente manera:

En el Capítulo 1 se detallan las definiciones básicas sobre el control numérico computarizado, las características de una máquina CNC, la naturaleza del proceso a automatizar y los elementos principales para el uso del MATLAB, tales como funciones y sentencias, además de los componentes para crear una interfaz gráfica. Se hace referencia al uso de una tarjeta de adquisición de datos de la NI y su compatibilidad con MATLAB.

En el Capítulo 2 se desarrolla el software para convertir las coordenadas de la figura a cortar en señales eléctricas comandadas por un microcontrolador, esto mediante una interfaz de usuario por medio de la DAQ6501. También se realiza el diseño de los circuitos de control, de potencia y acondicionamiento de señales de los encoders.

En el Capítulo 3 se explica la evolución del sistema mecánico así como sus componentes y el resultado final con la incorporación de la antorcha.

En el Capítulo 4 se exponen las pruebas que se realizan durante el desarrollo del proyecto, los problemas presentados y las soluciones dadas, así como los resultados obtenidos, y como punto principal, el resultado final del sistema.

En el Capítulo 5 se presentan las conclusiones y recomendaciones del proyecto.

Finalmente, en la sección de anexos se presenta el manual de usuario, los diagramas circuitales, hojas de datos de los componentes electrónicos y los planos del sistema mecánico.

CAPÍTULO 1

MARCO TEÓRICO

1.1 INTRODUCCIÓN

La necesidad de obtener productos de buena calidad y en grandes cantidades fué uno de los parámetros que dió lugar a los procesos de automatización, los mismos que con el pasar del tiempo permitieron fabricar productos complejos imposibles de realizar por un operador humano, es por esto que las máquinas controladas desde un computador son, hoy en día, indispensables para la mayoría de procesos industriales.

El control numérico computarizado busca mejorar la operación de la maquinaria en lo que se refiere a rapidez, precisión, y calidad, además permite realizar una variedad de productos debido a la flexibilidad que otorga un programa, que con el simple hecho de cambiar un código, o introducir una secuencia de datos diferentes en el procedimiento a seguir, permite cambiar totalmente las características del producto terminado.

Las ventajas que una máquina controlada desde un ordenador ofrece son innumerables, más su alto costo impide que las pequeñas industrias puedan acceder a la misma; es por esto que el presente proyecto muestra en forma detallada el proceso realizado para automatizar una máquina oxicorte, considerando el diseño del software y la implementación del hardware.



Figura 1.1 Trabajo de una máquina oxicorte

1.2 EXPLICACIÓN GENERAL DEL PROCESO

Mediante el uso de MATLAB se escribe un archivo .m, que incluye una interfaz gráfica creada en GUIDE (herramienta gráfica de MATLAB), el cual permite procesar una imagen prediseñada, generando un código (código G), el mismo que por medio de una DAQ NI 6501 USB es enviado a un microcontrolador ATTINY 2313, para ser interpretado y posteriormente enviar señales a un puente H (L298N). Dicho puente envía señales a dos motores acoplados a una máquina para transformar el movimiento circular en lineal y permitir el desplazamiento de un soplete (parte de la oxicorte) en cualquier punto de un plano.



Figura 1.2 Descripción general del proceso

1.3 CONTROL NUMÉRICO COMPUTARIZADO

El origen de las primeras máquinas de control numérico data de los años 1940 y 1950, dichas máquinas usaban motores, cuyos mandos se accionaban automáticamente siguiendo las instrucciones dadas en un sistema de tarjeta perforada. Los microcontroladores han permitido el dominio de las máquinas dando lugar a la denominación de control numérico por computadora (CNC), para diferenciarlas de las máquinas que no tenían computadora. [1]

1.3.1 DEFINICIÓN

El control numérico computarizado (CNC) es un sistema de automatización por lo general para máquinas herramientas, las mismas que normalmente se manejan por un computador, mediante comandos programados en un medio de almacenamiento. Los comandos son susceptibles a cambios dependiendo del proceso que se desee realizar, debido a su flexibilidad el control numérico es usado en diferentes tipos de máquinas, mejorando en cada una la precisión y rapidez de fabricación de un producto.[1]

1.3.2 CARACTERÍSTICAS DEL CONTROL NUMÉRICO COMPUTARIZADO

Entre las principales características que posee una máquina con control numérico computarizado se consideran las siguientes:

- Mayor precisión y mejor calidad de productos.
- Uniformidad en los productos fabricados.
- No se requiere un operario a tiempo completo.
- Rápido procesamiento de productos de complicada fabricación.
- Se requiere poco tiempo para diseñar un nuevo producto.
- Reducción de costos debido a la rápida y eficiente fabricación.
- El manejo de la maquinaria no requiere previos conocimientos.
- Se disminuye el trabajo del operador.
- Se proporciona mayor seguridad a los trabajadores.

- Eficientes cronogramas de producción, es posible cumplir en su totalidad metas establecidas.
- Alto costo de la maquinaria.
- En caso de falla se requiere personal especializado para reparar el equipo.
- Es indispensable fijar en forma correcta la secuencia de operación para un eficiente funcionamiento.[2]

1.3.3 MÁQUINAS HERRAMIENTAS DE CONTROL NUMÉRICO

El término máquina herramienta indica herramientas que no utilizan la operación humana. Se puede decir que el primer torno considerado máquina herramienta fue el inventado alrededor de 1751 por Jacques de Vaucanson, puesto que fue el primero que incorporó el instrumento de corte en una cabeza ajustable mecánicamente, quitándolo de las manos del operario.

La máquina herramienta es una máquina que se utiliza para dar forma a materiales sólidos, por lo general metales. Suelen ser máquinas estacionarias. El moldeado de la pieza se realiza por eliminación de material, que se puede ser por arranque de viruta, por estampado, corte o electroerosión.

Las primeras máquinas usaban volantes para estabilizar su movimiento y poseían sistemas complejos de engranajes y palancas para controlar la máquina y las piezas en que trabajaba.

Las máquinas de control numérico utilizaban una serie de números perforados en una cinta de papel o tarjetas perforadas para controlar su movimiento. En los años 1960 se añadieron computadoras para aumentar la flexibilidad del proceso. Tales máquinas se comenzaron a llamar máquinas CNC, o máquinas de Control Numérico por Computadora. Las máquinas de control numérico y CNC pueden repetir secuencias una y otra vez con precisión, y pueden producir piezas mucho más complejas que las que pueda hacer el operario más experimentado.[3]

1.3.3.1 Tipos De Máquina Herramienta

Por la forma de trabajar las máquinas herramientas se pueden clasificar en tres tipos.

1.3.3.1.1 De desbaste o desbastadoras

Dan forma a la pieza por arranque de viruta.

1.3.3.1.2 Prensas

Dan forma a las piezas mediante el corte, el prensado o el estirado.

1.3.3.1.3 Especiales

Dan forma a la pieza mediante técnicas diferentes, láser, electroerosión, ultrasonidos, plasma.[3]

1.3.3.2 Principales Máquinas Herramientas

1.3.3.2.1 Torno

Trabaja mediante el arranque de material basado en una herramienta cortante y brocas. La pieza a trabajar gira y mediante un carro en el que se sitúa la herramienta de corte se va desgastando, obteniendo partes cilíndricas y cónicas. Si se coloca una broca, se pueden realizar agujeros.[4]



Figura 1.3 Torno CNC

1.3.3.2.2 Taladros

Usados para perforación. El útil es el que gira y la pieza permanece fija. El útil suele ser normalmente una broca que realiza el agujero. También se pueden realizar otras operaciones con diferentes útiles, como avellanar y escariar. Sus velocidades de giro suelen ser muy elevadas. [4]



Figura 1.4 Taladro CNC

1.3.3.2.3 *Fresadora*

Busca obtener superficies de una forma concreta, en las fresadoras el útil gira y la pieza permanece fija a una bancada móvil. El útil utilizado es la fresa, que suele ser redonda con diferentes filos, su forma coincide con la forma de la pieza a trabajar. La pieza se coloca sólidamente fijada a un carro que le acerca a la fresa en las tres direcciones, esto es en los ejes X, Y y Z. [4]



Figura 1.5 Fresadora CNC

1.3.3.2.4 *Pulidora*

Trabaja con un disco abrasivo que va comiendo el material de la pieza a trabajar. Se suele utilizar para los acabados de precisión. Normalmente no se ejerce presión mecánica sobre la pieza. [4]



Figura 1.6 Pulidora CNC

1.3.3.3 Elementos De Una Máquina Herramienta De Control Numérico

1.3.3.3.1 Programa de instrucciones

Consta de una serie de sentencias ejecutadas paso a paso que dirigen el equipo de procesado. El código se escribe en un lenguaje de programación.

1.3.3.3.2 Control numérico (CN)

Es la unidad que debe interpretar las instrucciones contenidas en el programa, convertirlas en señales que accionen los dispositivos de las máquinas y comprobar su resultado.

1.3.3.3.3 Equipo de procesado

Es el componente que realiza el trabajo útil; formado por la mesa de trabajo, las máquinas herramienta así como los motores y controles para moverlas. [5]

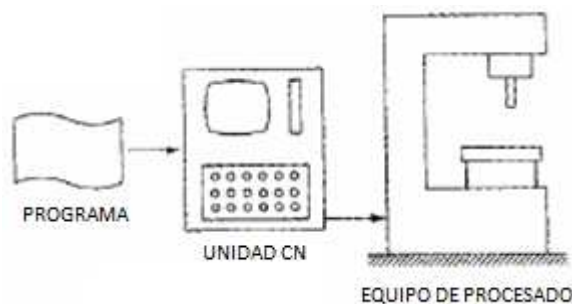


Figura 1.7 Elementos de una máquina herramienta de Control Numérico

1.3.3.4 Principios De Mecanizado En Control Numérico

El mecanizado es un proceso de fabricación que comprende un conjunto de operaciones para la conformación de piezas, en este caso consta de dos etapas: en la primera, el metal se calienta a alta temperatura (900°C) con la llama producida por el oxígeno y un gas combustible; en la segunda, una corriente de oxígeno corta el metal y elimina los óxidos de hierro producidos.

En este proceso se utiliza un gas combustible cualquiera (acetileno, hidrógeno, propano, hulla, tetreno o crileno), cuyo efecto es producir una llama para calentar el material, mientras que como gas comburente siempre ha de utilizarse oxígeno, a fin de causar la oxidación necesaria para el proceso de corte.

1.3.3.5 Ejes De Coordenadas En Control Numérico

Una máquina CNC es capaz de realizar diferentes tipos de movimiento, dependiendo de la aplicación que se desee, entre ellos se tiene máquinas desde dos ejes que permiten movimiento en dos dimensiones, hasta máquinas de nueve ejes, como es el caso de la máquina curvadora de tubos, que permite dar complejas formas geométricas al material, debido a sus múltiples planos, y radios; a continuación se detallan las máquinas CNC más usadas, en lo referente al número de ejes.

1.3.3.5.1 Máquinas CNC dos ejes

Los ejes son los planos cartesianos X y Y, cada uno con dos grados de libertad, se puede obtener un movimiento en dos dimensiones, las piezas obtenidas al usar este tipo de máquina se limitan a un plano. [6]



Figura 1.8 Máquina CNC dos ejes

1.3.3.5.2 Máquinas CNC tres ejes

Los ejes son los planos cartesianos X , Y, y Z cada uno con dos grados de libertad, se puede obtener un movimiento en tres dimensiones, las piezas obtenidas al usar este tipo de máquina tienen tres planos. Un ejemplo es la fresadora que permite mecanizar piezas. [6]



Figura 1.9 Piezas realizadas por una fresadora

1.4 UTILIZACIÓN DE MATLAB

MATLAB es un software matemático que ofrece un entorno de desarrollo integrado con un lenguaje de programación propio (lenguaje M). Permite la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos hardware. [7]

Para realizar el presente proyecto, en lo que se refiere a programación, se ha usado archivos M-File y Guide, que permiten escribir una secuencia de códigos, que al ejecutarse dan lugar al procesamiento de una imagen para finalmente obtener un código que permita el movimiento en dos dimensiones del soplete de una máquina oxicorte.

1.4.1 PROCESAMIENTO DE IMÁGENES

La primera idea es obtener el contorno de cualquier imagen, para mediante el mismo, procesar datos y llegar a obtener un código capaz de generar movimiento del soplete en una máquina oxicorte.



Figura 1.10 Imagen sin procesar, e imagen procesada

La figura procesada es una matriz de $n \times m$ elementos, donde los puntos blancos son 1L y los negros son 0L, cada punto tiene su ubicación representada por un valor (x,y) dentro de la matriz. Si se obtiene cada coordenada donde se encuentran ubicados los números 1L, se obtiene un vector que representa todo el perfil del dibujo.

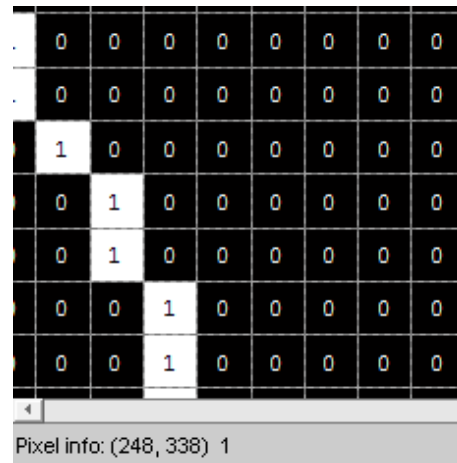


Figura 1.11 Matriz que representa una imagen procesada

El problema de obtener contornos con imágenes no prediseñadas o con sombras, es no tener un perfil definido y continuo, por lo cual resulta muy complejo obtener un vector que represente toda la figura.

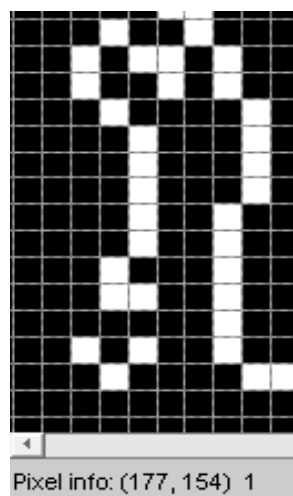
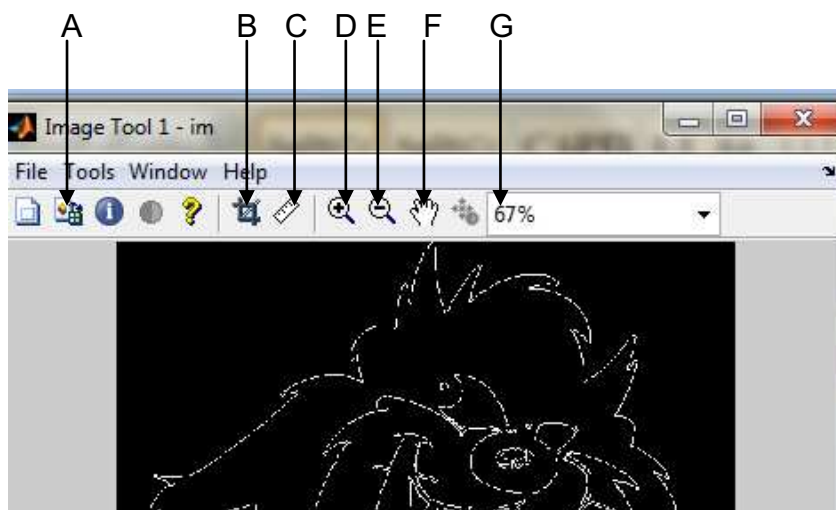


Figura 1.12 Discontinuidad de perfil en una imagen con sombras

Debido al inconveniente mencionado se utilizan figuras prediseñadas evitando así la discontinuidad que ofrecen las figuras con sombras y no definidas.

1.4.1.1 Herramienta Imtool

MATLAB tiene la capacidad de leer cualquier imagen y transformarla a una matriz; la herramienta *imtool* permite ver en detalle los parámetros referentes a dicha matriz. A continuación se detallan los usos principales de *imtool*.



1.4.1.1.1 A: Pixel Region.

Abre una ventana que permite desplazarse por la matriz que representa una imagen, observar el valor numérico de sus elementos, la posición que ocupan, su color, y alejarse o acercarse según se desee.

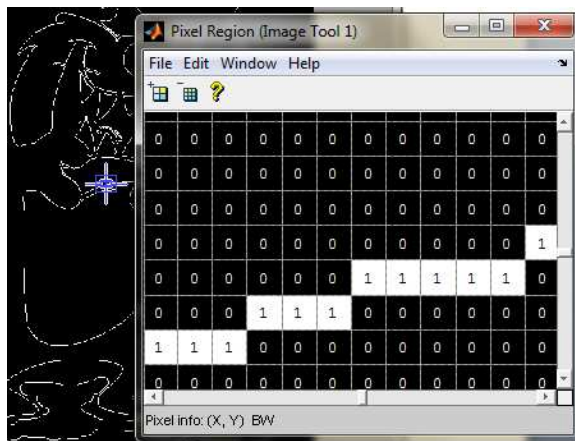


Figura 1.13 Pixel Region

1.4.1.1.2 B: Crop Image

Permite observar una parte exacta de la matriz, eliminando lo que no haya sido seleccionado.

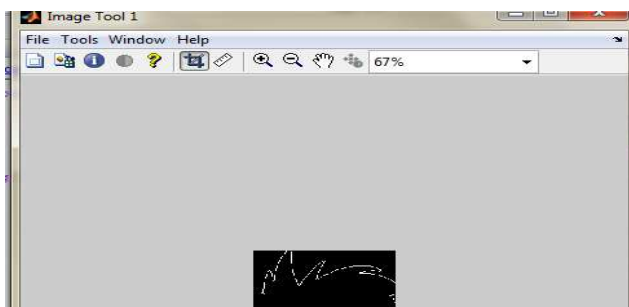


Figura 1.14 Crop Image

1.4.1.1.3 C: Measure Distance.

Mediante el uso de esta herramienta se obtiene la distancia entre dos puntos seleccionados, es decir el número de píxeles que existe entre dichos puntos.

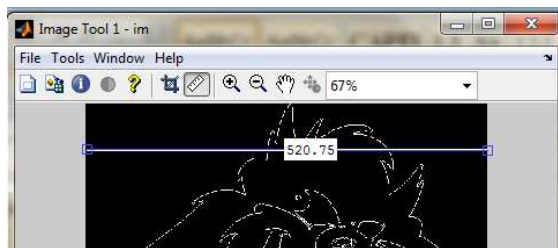


Figura 1.15 Measure Distance

1.4.1.1.4 D: Zoom In

Acerca la matriz imagen.

1.4.1.1.5 E: Zoom Out

Aleja la matriz imagen.

1.4.1.1.6 F: Pan

Permite sostener la matriz imagen y realizar un movimiento en un solo plano, sin alejar o acercar la misma.

1.4.1.1.7 G: Porcentaje

Permite ver el porcentaje de visualización de la matriz imagen.

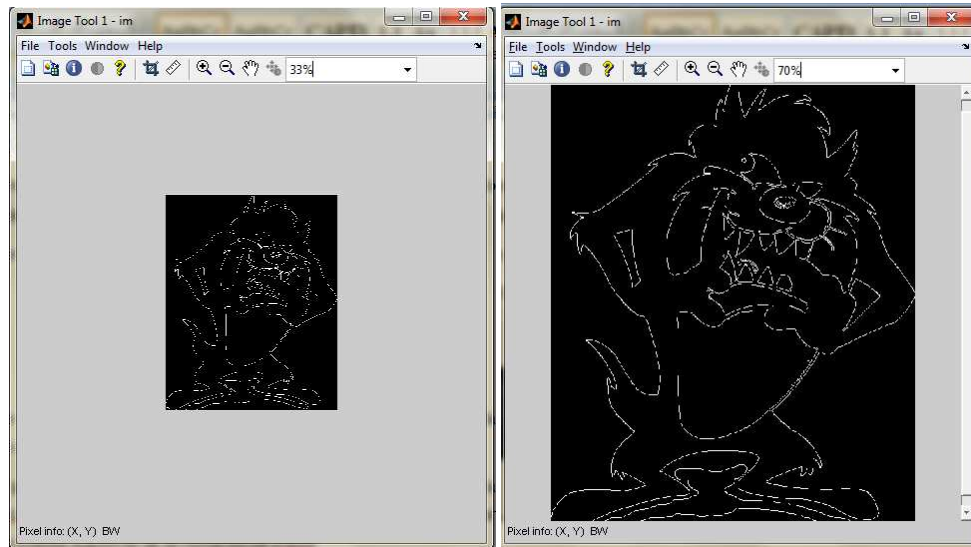


Figura 1.16 Porcentaje de visualización en la matriz imagen

El número de elementos de una matriz que proviene de una imagen procesada es igual al número de píxeles de la imagen.

Al colocar el puntero del mouse sobre un elemento de la imagen, se obtiene la coordenada (columna, fila) del elemento en mención; este valor se observa en la esquina inferior izquierda de la herramienta *imtool*.

1.4.2 UTILIZACIÓN DE DAQ

1.4.2.1 Características Generales

El USB-6501 de National Instruments es un dispositivo de E/S digital portátil, que brinda adquisición de datos y control confiables. Con conectividad USB plug-and-play, el NI USB-6501 es suficientemente robusto y versátil para aplicaciones de laboratorio e industriales. Posee una interfaz de bus USB 2.0 de alta velocidad (12 Mb/s). [9]



Figura 1.17 DAQ NI 6501 USB

A continuación se mencionan algunas de las principales características:

- Producto USB-6501
- Familia de Productos E/S Digital
- Formato Físico USB
- Sistema Operativo/Objetivo Windows , Linux , Mac OS , Pocket PC
- Tipos de Medida Digital
- Canales Bidireccionales 24
- Canales de Entrada Únicamente 0
- Canales de Salida Únicamente 0
- Número de Canales 24
- Niveles Lógicos TTL
- Capacidad de Corriente Simple 8.5 mA
- Capacidad de Corriente Total 65 mA [9]

1.4.2.2 Compatibilidad De Daq 6501 Con Pc

Se instala el driver (NI-DAQmx 9.0.2) que incluye el programa Measurement & Automation, dicho programa permite verificar la correcta compatibilidad de la DAQ con el computador y el correcto funcionamiento de sus puertos.

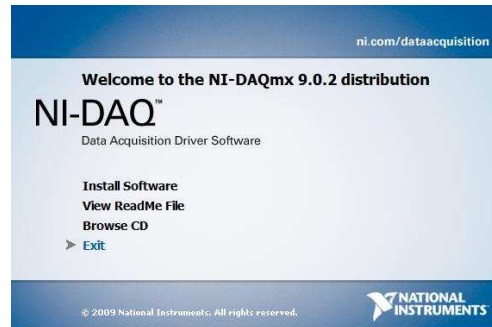


Figura 1.18 Driver NI-DAQmx 9.0.2

A continuación se procede a verificar los puertos de la DAQ con la herramienta *Test Panels* presente en el software Measurement & Automation. Se pueden declarar los pines de un puerto como entrada o salida. Si la declaración es una salida entonces es posible enviar un valor 1L ó 0L por un pin con sólo poner el interruptor mostrado en la interfaz (Figura 1.20) en estado alto o bajo. Si el pin es declarado como entrada permanece en estado alto (1L) hasta que se realice una conexión entre la tierra y el pin de un mismo puerto, en cuyo caso el estado del pin se torna bajo (0L), lo que indica la detección de una señal de entrada. Si se ha declarado un pin como entrada o salida se debe presionar el botón START para que el cambio surta efecto.



Figura 1.19 Measurement & Automation Explorer

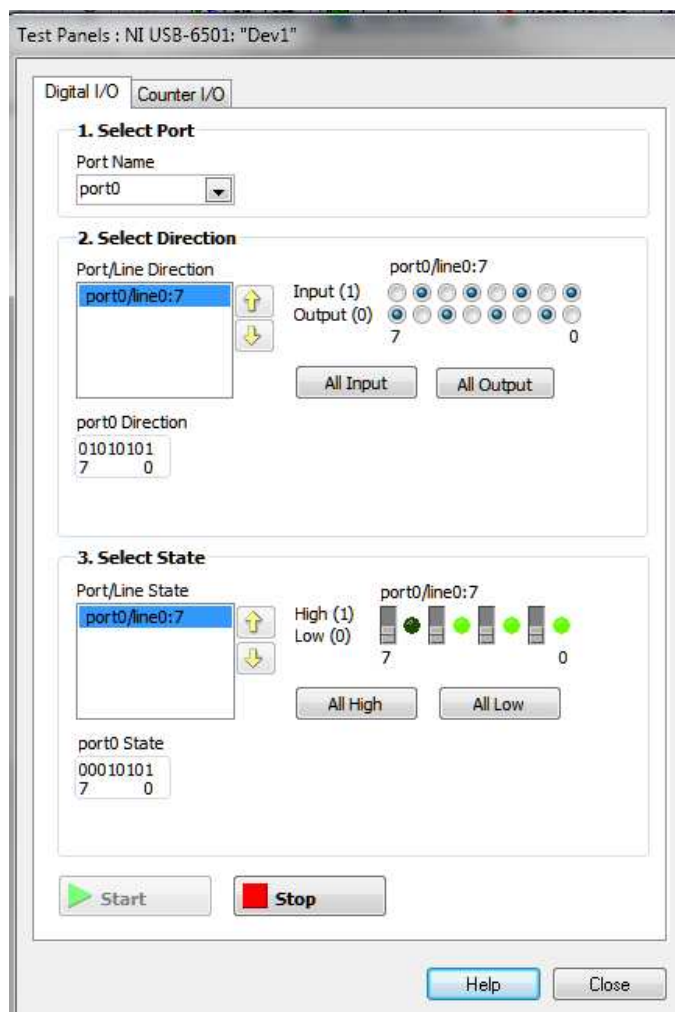


Figura 1.20 Test Panel DAQ 6501

1.4.2.3 Compatibilidad De Daq 6501 Con Matlab

Una vez verificados los puertos y la correcta compatibilidad de la DAQ con el computador se procede a generar el código necesario para usar la DAQ 6501 con el entorno MATLAB.

MATLAB posee una librería llamada Data Acquisition Toolbox, la cual permite dar un nombre a la DAQ y a sus puertos para poder enviar u obtener un dato según se requiera.

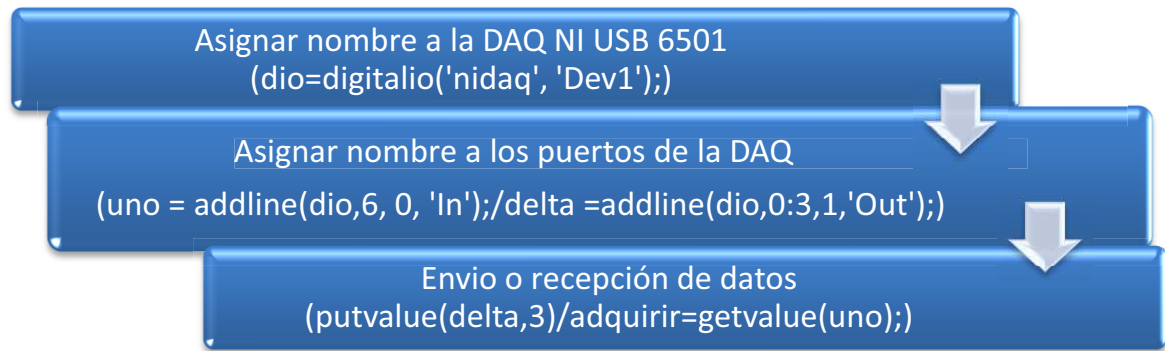


Figura 1.21 Proceso de compatibilidad de la DAQ 6501 con MATLAB

Explicación:

- En el primer cuadro se nombra a la DAQ 6501 “dio”.
- En el segundo cuadro se nombra “unos” y se define como línea de entrada (‘In’) al pin 6 de puerto 0 que pertenecen DAQ llamada “dio”. Adicionalmente se nombra “delta” y se define como líneas de salida (‘Out’) a los pines 0,1,2,3 de puerto 1 que pertenecen a la DAQ llamada “dio”.
- En el tercer cuadro se envía por la línea de salida llamada “delta” el valor 3 (binario: 0011), en la siguiente instrucción se recibe en la variable “adquirir” el valor encontrado en la línea de entrada llamada “unos”.

Terminado este proceso, la tarjeta de adquisición de datos es el camino por el cual, un programa realizado en archivo.m, envía un código hasta un microcontrolador ATTINY2313 mediante comunicación en paralelo.

1.4.2.4 Datos Que Maneja La Daq

Para el funcionamiento del programa del presente proyecto, la DAQ permite enviar dos valores y recibir un valor.

Acción	Dato	Descripción
Envío	Código G	Usa cuatro bits (0:3), ubicado en el puerto 1, envía un dato para el movimiento de los motores, y para enviar el siguiente dato espera que una señal de los encoders indique que el movimiento fue realizado.
Recepción	Encoders	Usan dos bits (5:6), ubicados en el puerto 0, reciben una señal por parte de los encoders, indica que se ha ejecutado un dato enviado por la DAQ (código G), y permite que se envíe el siguiente dato.

Tabla 1.1 Descripción de datos en los puertos de la DAQ 6501



Figura 1.22 Manejo de datos DAQ

1.4.3 HERRAMIENTA GUIDE

Las interfaces gráficas de usuario (GUI: Graphical User Interface), son la forma en que el usuario interactúa con el programa o el sistema operativo de una computadora. Una GUI contiene diferentes elementos como botones, campos de texto, menús, gráficos, entre otros. [8]

MATLAB permite realizar GUIs de una manera muy sencilla usando GUIDE (Graphical User Interface Development Environment). La forma de implementar GUI

con MATLAB es creando los objetos y definiendo las acciones que cada uno va a realizar. Al usar GUIDE se obtienen los siguientes archivos:

Archivo FIG.- Contiene la descripción de los componentes que contiene la interfaz.

Archivo M.- Contiene las funciones y los controles del GUI, así como los *callback* de cada elemento de la interfaz.

Un *callback* se define como la acción que llevará a cabo un objeto de la GUI cuando el usuario lo active.

Para crear una GUI en MATLAB se escribe GUIDE en el Command Window o se busca la opción en File en el menú principal.

A continuación se presenta el siguiente cuadro de dialogo: [9]

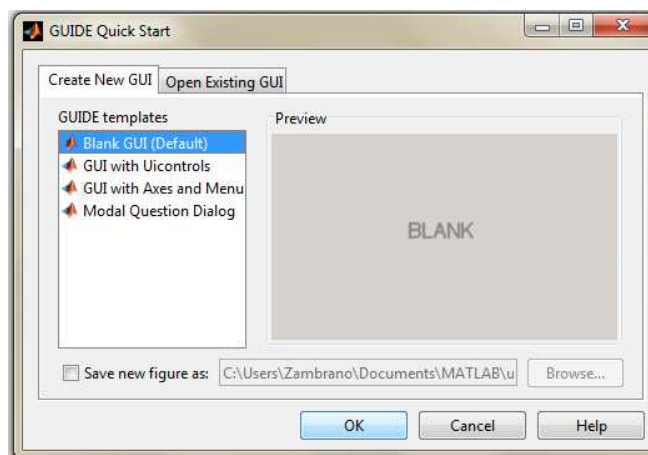


Figura 1.23 Cuadro mostrado al iniciar GUIDE

a) Blank GUI (Default)

La opción de interfaz gráfica de usuario en blanco (viene predeterminada), presenta un formulario nuevo, en el cual se puede diseñar un programa.

b) GUI with Uicontrols

Esta opción presenta un ejemplo en el cual se calcula la masa, dada la densidad y el volumen, en alguno de los dos sistemas de unidades. Se puede ejecutar este ejemplo y obtener resultados.

c) GUI with Axes and Menu

Esta opción es otro ejemplo el cual contiene el menú File con las opciones Open, Print y Close. En el formulario tiene un *popup menu*, un *push button* y un objeto *axes*, puede ejecutarse el programa eligiendo alguna de las seis opciones que se encuentran en el menú despegable y haciendo click en el botón de comando.

d) Modal Question Dialog

Con esta opción, se muestra en la pantalla un cuadro de diálogo común, el cual consta de una pequeña imagen, una etiqueta y dos botones Yes y No, dependiendo del botón que se presione, el GUI retorna el texto seleccionado (la cadena de caracteres 'Yes' o 'No').

Al elegir la opción Blank GUI, se tiene:

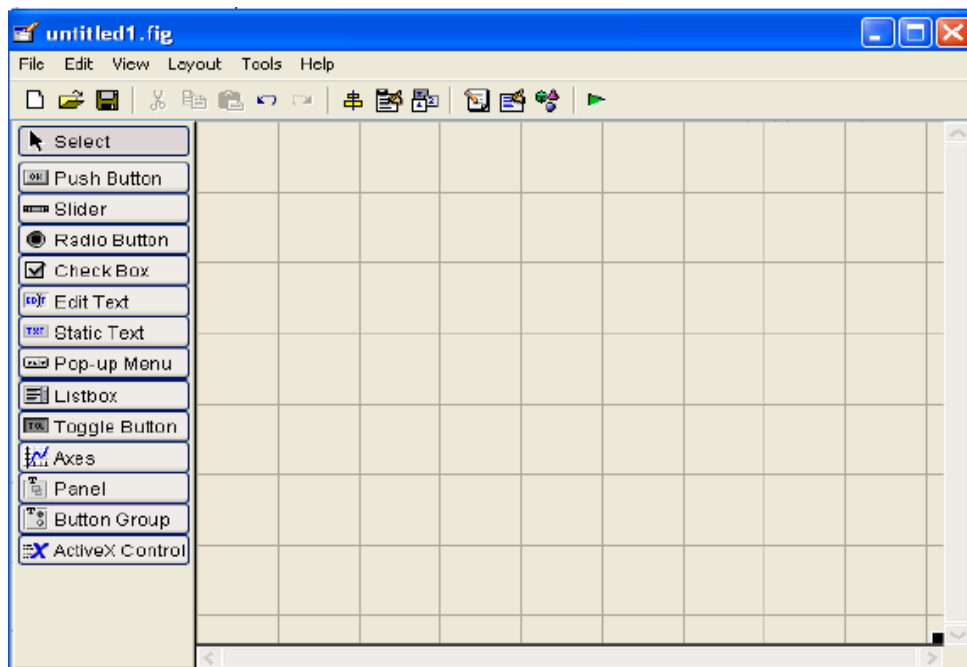


Figura 1.24 Blank GUI

1.4.3.1 Descripción De Los Principales Componentes De Una Ventana Gui

CONTROL	DESCRIPCIÓN
Check box	Indica el estado de una opción o atributo
Editable Text	Caja para editar texto
Pop-up menú	Provee una lista de opciones
List Box	Muestra una lista deslizable
Push Button	Invoca un evento inmediatamente
Radio Button	Indica una opción que puede ser seleccionada
Toggle Button	Sólo dos estados, "on" o "off"
Slider	Usado para representar un rango de valores
Static Text	Muestra un string de texto en una caja
Panel button	Agrupar botones como un grupo
Button Group	Permite exclusividad de selección con los radio button
Axes	Permite mostrar gráficos.

Tabla 1.2 Principales componentes de una ventana GUIDE [9]

Cada elemento GUI tiene un conjunto de opciones a las cuales se puede acceder. Entre las principales está la opción *Property Inspector* que permite personalizar cada elemento. El *tag* del elemento es el identificador con el cual el archivo .m reconoce al elemento, su *string* indica los símbolos que llevará el elemento en la

interfaz gráfica, adicionalmente se puede cambiar el color, tipo de letra y rangos en algunos casos.



Figura 1.25 Property Inspector GUI

En el presente proyecto se utilizan los siguientes elementos:

- Push button
- Axes
- Static text
- List box

Como se puede observar a continuación:

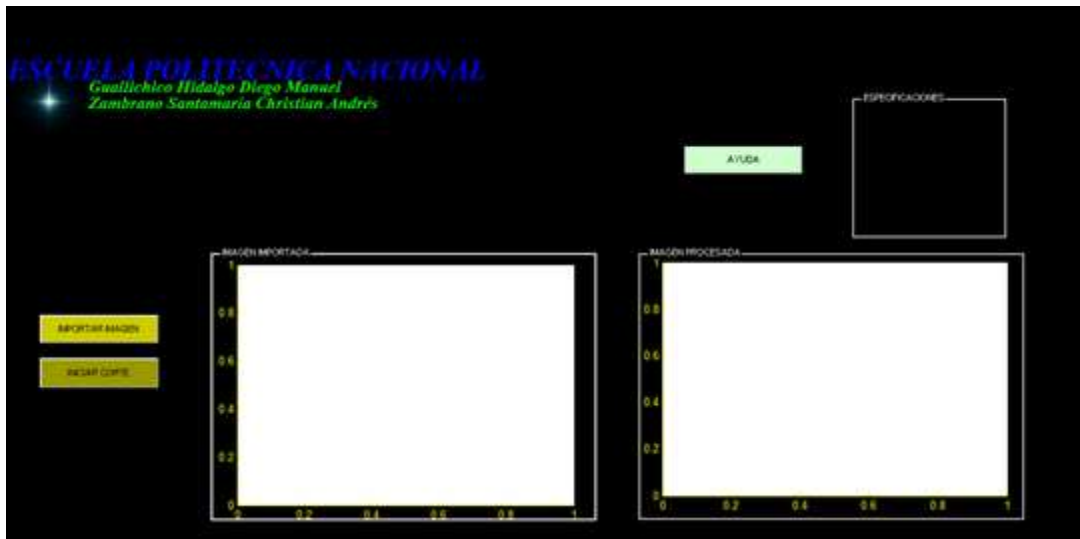


Figura 1.26 GUI Ventana Principal - Máquina oxicorte

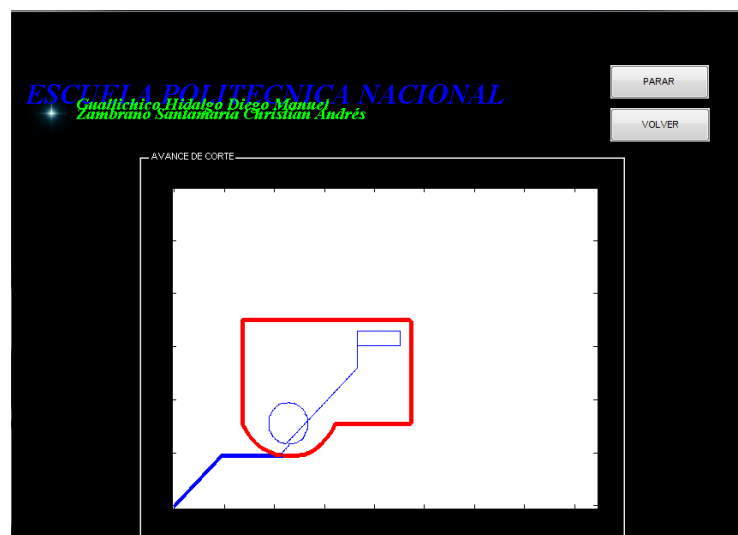


Figura 1.27 GUI Animación de corte

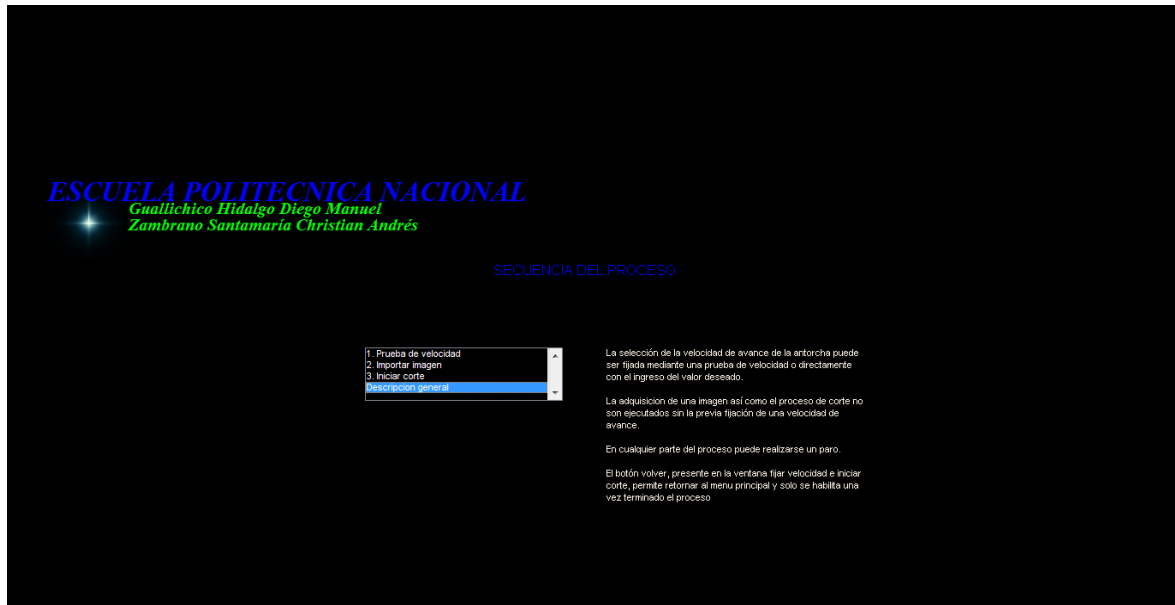


Figura 1.28 Ventana de ayuda

CAPÍTULO 2

DESCRIPCIÓN DEL ALGORITMO Y MÉTODO DE CONTROL

2.1 DESCRIPCIÓN GENERAL

El algoritmo de control consta de varias etapas que van desde la importación de la imagen a cortar, el tratamiento de la matriz obtenida para la identificación de los contornos existentes en la misma, hasta la generación del código necesario para dar una orden de movimiento a los motores mediante el llamado código G.

Una de las etapas más importantes es la interconexión del hardware a utilizar, en este caso una tarjeta de adquisición de datos, la cual enviará las órdenes de movimiento a los motores junto con un microcontrolador, que es el encargado de emitir una señal de confirmación de posicionamiento mediante el uso de encoders incrementales.

El sistema de control cuenta con una interfaz de comunicación creada con la herramienta GUI de MATLAB. Esta interfaz consta de varias etapas de acuerdo al procedimiento necesario para realizar cortes mediante el uso de un oxicorte, tal es el caso de un calentamiento previo de la antorcha antes de empezar el corte en sí. Este proceso se representa en la interfaz mediante una animación que implica el posicionamiento de la antorcha y el corte como tal.

Uno de los factores a tomar en cuenta es la posible eventualidad de una emergencia, es por esto que la interfaz cuenta con un botón de PARO que permite la opción de retorno al proceso de corte o la cancelación total del mismo.

2.2 IMPORTACIÓN DE IMÁGENES

Al tratarse de una imagen en escala de grises, sus elementos son representados en una matriz de $m \times n$ elementos, donde n representa el número de píxeles de ancho y m el número de píxeles de largo. Para leer imágenes contenidas en un archivo al ambiente de MATLAB se utiliza la función *imread* cuya sintaxis es: [9]

```
imread ('nombre del archivo')
```

Donde 'nombre del archivo' es una cadena de caracteres que contiene el nombre completo de la imagen con su respectiva extensión; MATLAB soporta archivos con extensiones mostradas en la Tabla 2.1. Para introducir una imagen guardada en un archivo con alguno de los formatos especificados en la tabla anterior solo se tiene que usar la función *imread* y asignar su resultado a una variable que representará a la imagen, por ejemplo: [9]

Para la imagen mostrada en la Figura 2.1,



Figura 2.1 Imagen a importar

```
var = imread('Laera.jpg');
```

Con esto, la imagen contenida en el archivo *imagen.jpg*, quedará contenida dentro de la variable *var*.

FORMATO	EXTENSIÓN
TIFF	.tif
JPEG	.jpg
GIF	.gif
BMP	.bmp
PNG	.png
XWD	.xwd

Tabla 2.1 Formatos soportados por MATLAB

Después de realizar un procesamiento con la imagen, es necesario desplegar el resultado obtenido, la función `imshow('variable')` permite desplegar la imagen en una ventana en el ambiente de trabajo de MATLAB. Por ejemplo al escribir en la línea de comandos:

```
imshow(var);
```

Se obtendrá:

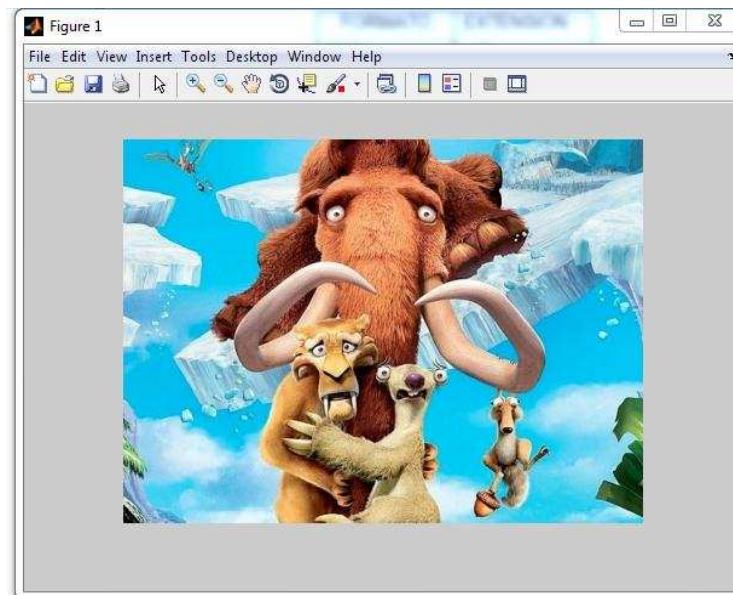


Figura 2.2 Imagen mostrada al usar la función `imshow`

Ahora la imagen está contenida dentro de una matriz, en este caso se trata de una imagen de color es decir una imagen RGB (Red, Green, Blue), que es representada por una matriz tridimensional de $m \times n \times p$, donde m y n tienen la

misma significación que para el caso de las imágenes de escala de grises, mientras que p representa el plano, que para RGB puede ser 1 para el rojo, 2 para el verde y 3 para el azul. [9]

2.2.1 CREACIÓN DE IMÁGENES PREDISEÑADAS

Las imágenes prediseñadas, por motivos del proceso de identificación de contornos deberán ser presentadas con una superficie rellena, de tal manera que se pueda identificar a simple vista los contornos de la misma, es decir, que se garanticen figuras cerradas al obtener una imagen binarizada. Es importante mencionar que los contornos a cortar deberán tener ángulos no menores a 20 grados, ya que por la naturaleza de la herramienta oxicorte dañaría el contorno. Con respecto al tamaño del formato A1, en el que se va a trabajar, los pixeles que lo determinan deberán ser de 841 x 594.

Para la creación de imágenes con una precisión aceptable es necesario emplear una plataforma CAD que permita realizar trazados y modificaciones. En este caso se utiliza AUTOCAD, en donde el área de trabajo debe ser de formato A1 que corresponde al número de pixeles requeridos.

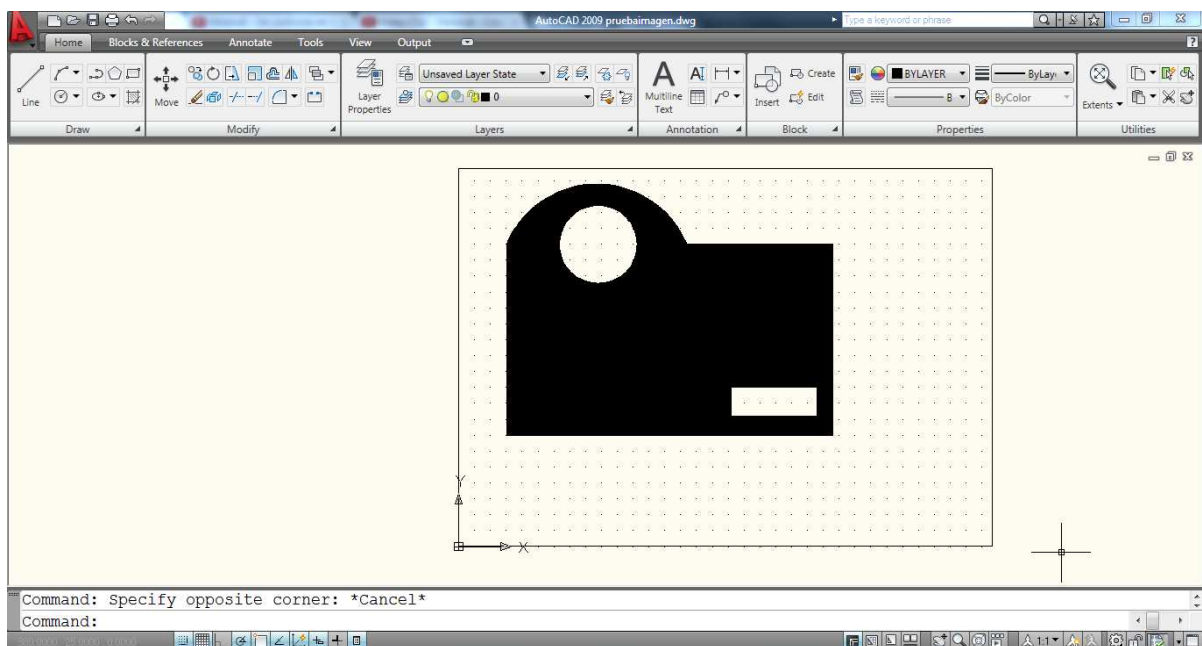


Figura 2.3 Diseño de imagen en AUTOCAD

La figura creada ahora debe ser convertida en un archivo en formato gráfico, para esto se emplea:

AutoCAD DWG to Image Converter 6.8.9



Figura 2.4 Icono de convertidor DWG a formato gráfico

Este software importa un archivo DWG y como archivo de salida se puede obtener diversos formatos gráficos, además permite fijar las dimensiones en pixeles de la imagen que se creará. [10]

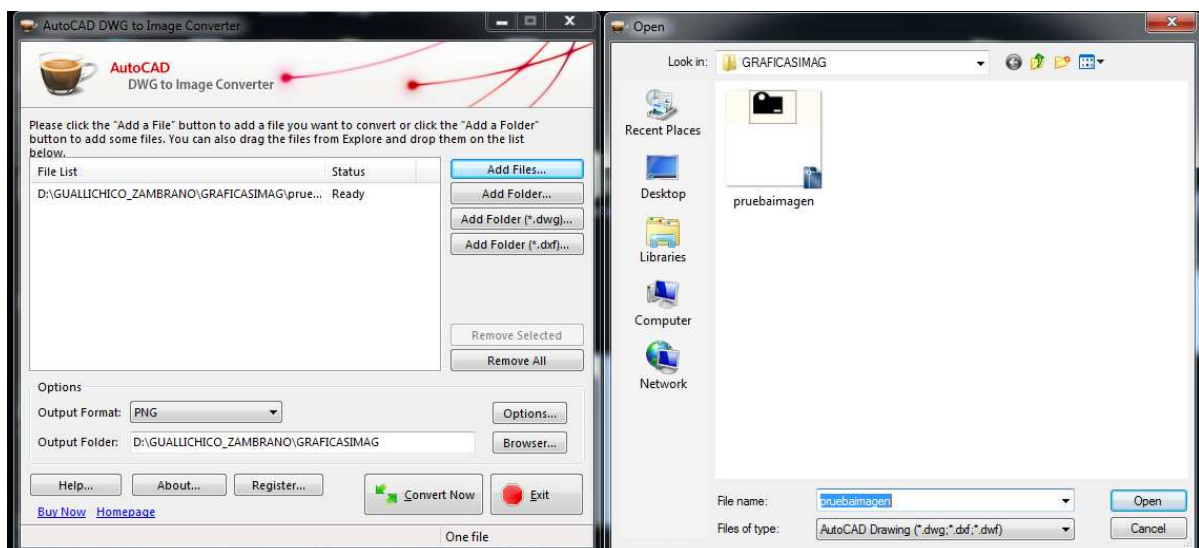


Figura 2.5 Interfaz de AutoCAD to Image Converter

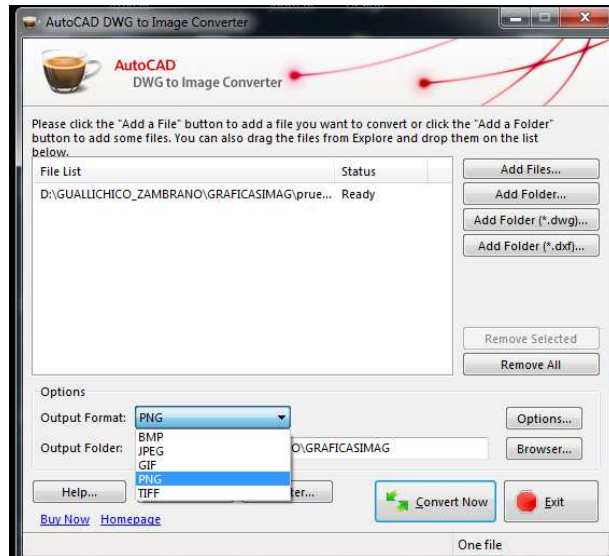


Figura 2.6 Selección de formato de imagen

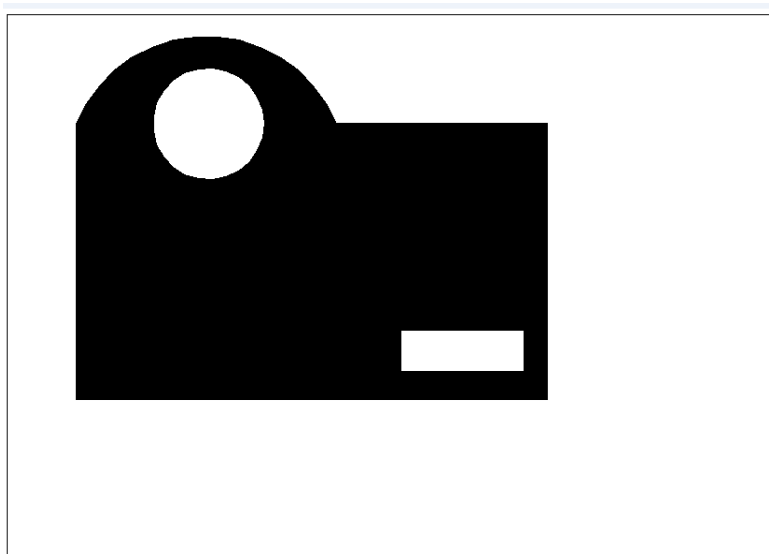


Figura 2.7 Imagen obtenida

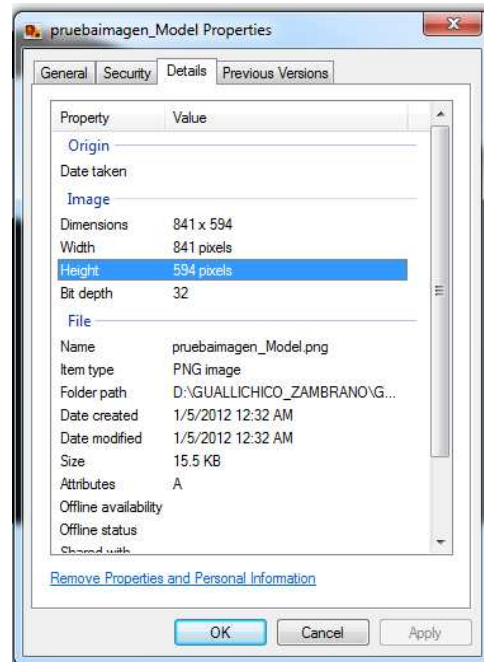


Figura 2.8 Propiedades de la imagen obtenida

Como se observa, la imagen obtenida cumple con los requerimientos en cuanto a las dimensiones de pixeles que representa a un área con formato A1.

2.2.1.1 Análisis De Imágenes Mediante La Herramienta Imtool

Para una mayor confiabilidad en cuanto a la continuidad de los contornos de la imagen creada se emplea a la herramienta *imtool*.

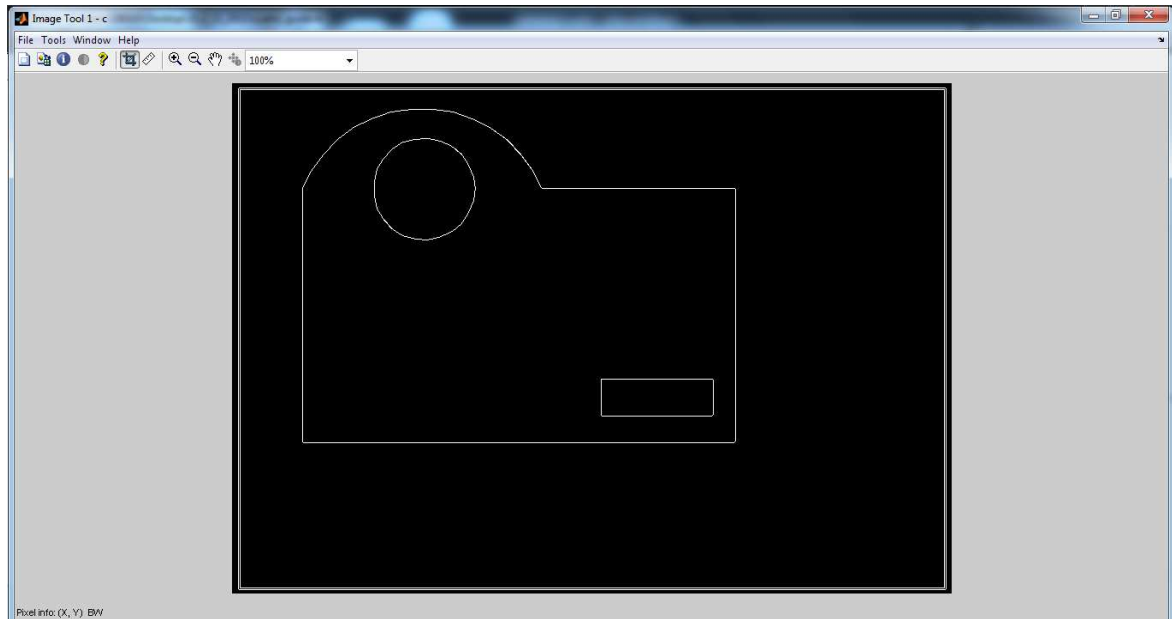


Figura 2.9 Utilización de *imtool* para análisis de discontinuidades en una imagen

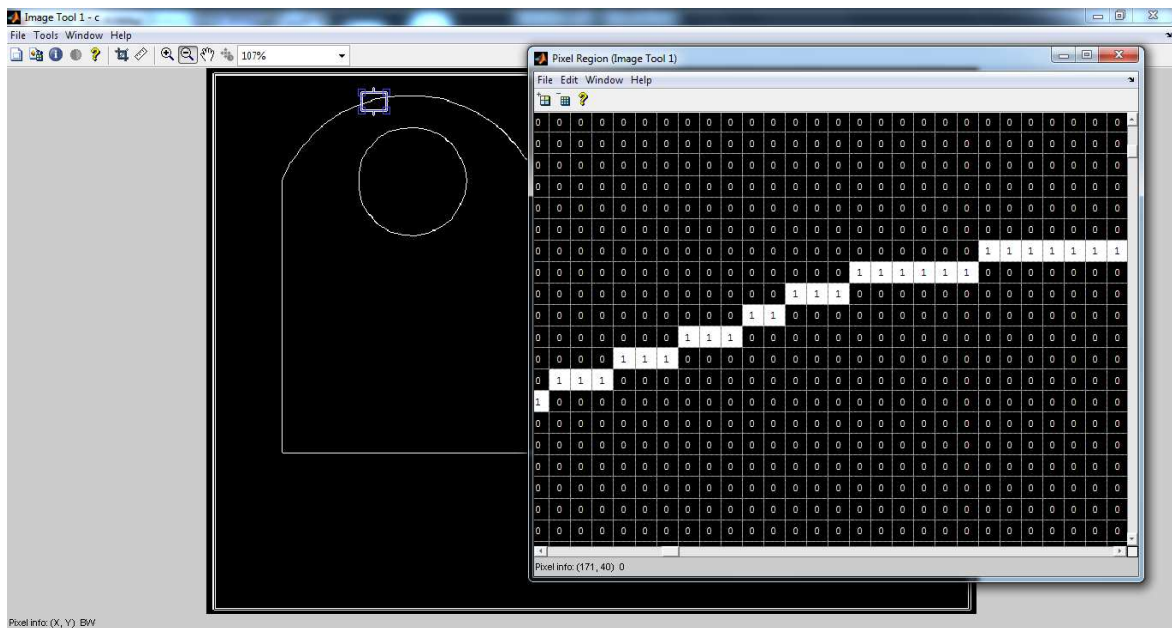


Figura 2.10 Verificación de continuidad en contornos

Como se observa, la imagen analizada cumple con los requerimientos de continuidad en su contorno.

2.2.2 IDENTIFICACIÓN DE CONTORNOS

Para el caso del procesamiento de la imagen a cortar en una lámina de metal, se requiere que la matriz con la que se trabajará sea de dos dimensiones, es por esto que se realiza una conversión de la matriz tridimensional en RGB a una matriz en dos dimensiones, lo que significa que se reducirá dicha imagen a escala de grises. El siguiente paso es reconocer los contornos de la imagen, para de esta manera obtener una matriz de ceros y unos, que representa a la imagen original, únicamente con sus contornos.

Por ejemplo, para obtener una imagen a escala de grises se emplea la sentencia:

im2bw(var), en este caso también se asigna esta matriz a una variable.

```
bin7 = im2bw(var);
```



Figura 2.11 Imagen en escala de grises

En este punto la imagen puede convertirse en una matriz que contenga únicamente los contornos de la figura empleando la función *edge*(var)

```
bord = edge(bin7);
```

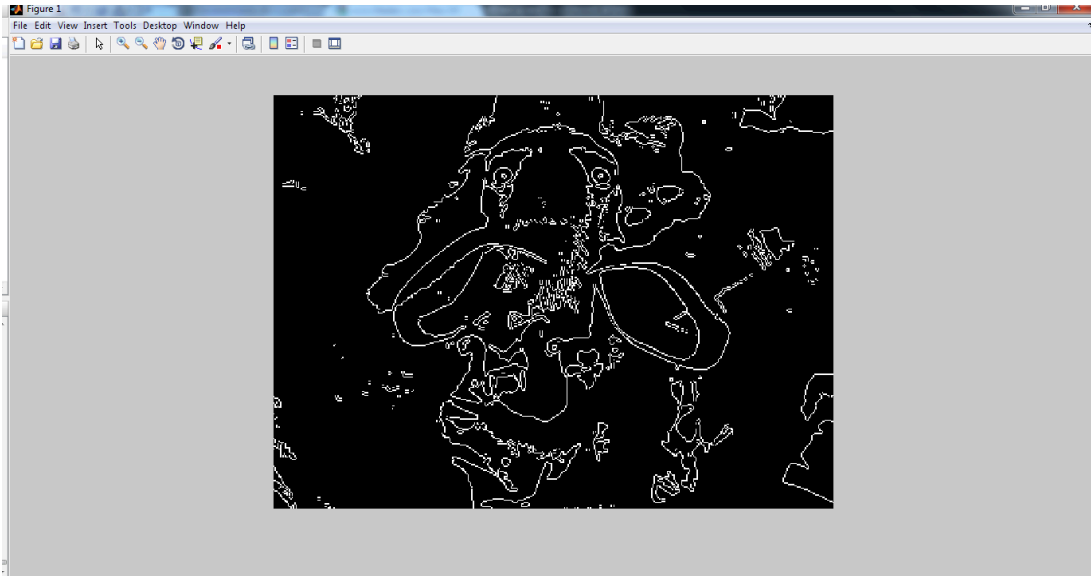


Figura 2.12 Identificación de contornos de una imagen

El objetivo es identificar los contornos existentes en una figura, en este caso se trata de figuras prediseñadas a ser cortadas en una plancha de metal, y con esta matriz obtenida ya es posible realizar un proceso secundario para la obtención de coordenadas de los elementos que componen dicha matriz.

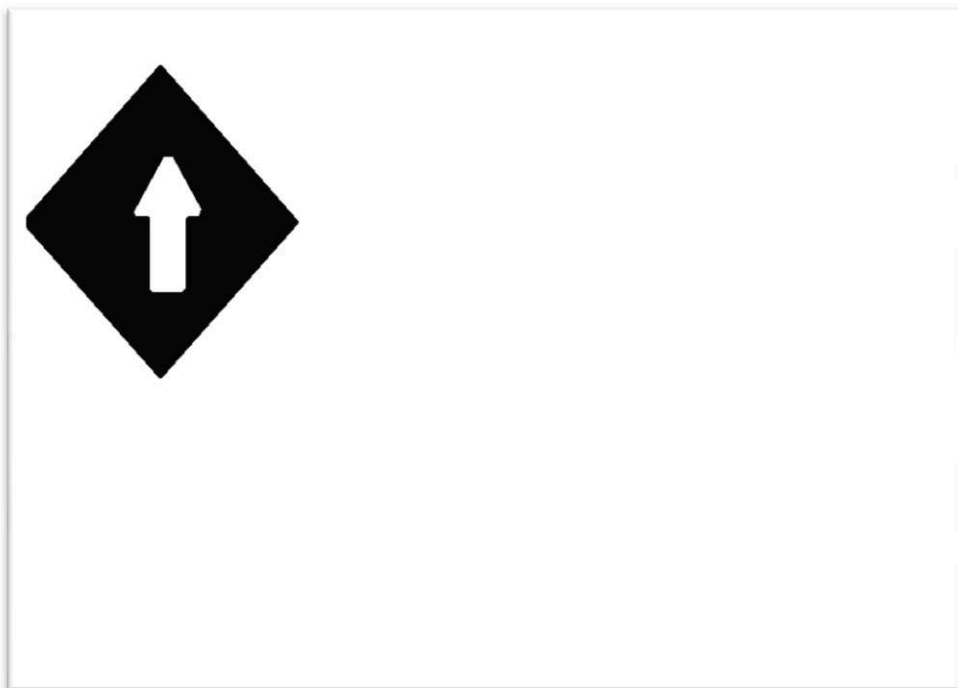
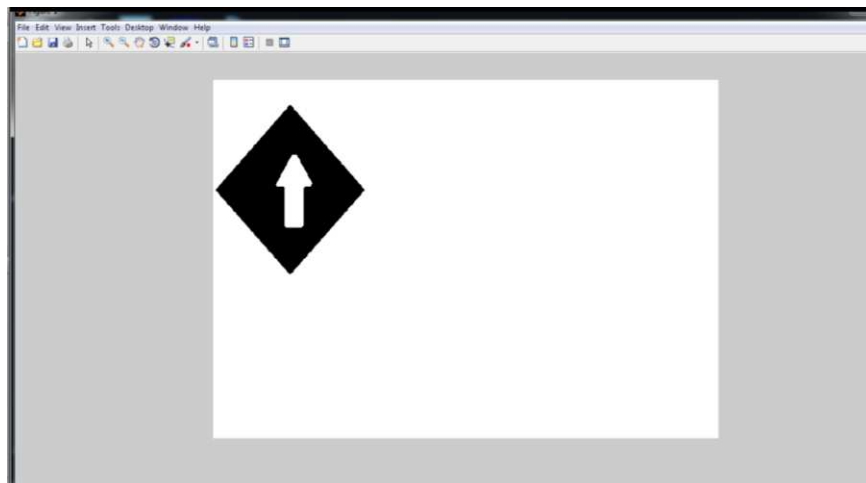


Figura 2.13 Modelo de dimensiones de imagen y de figura a cortar

El proceso de obtención de la matriz necesaria para definir los contornos de una imagen prediseñada es el mismo proceso descrito anteriormente, ahora debe realizarse un algoritmo para determinar el número de perímetros que existen en la imagen, así como también las coordenadas de cada valor que componen el perímetro al cual pertenece.

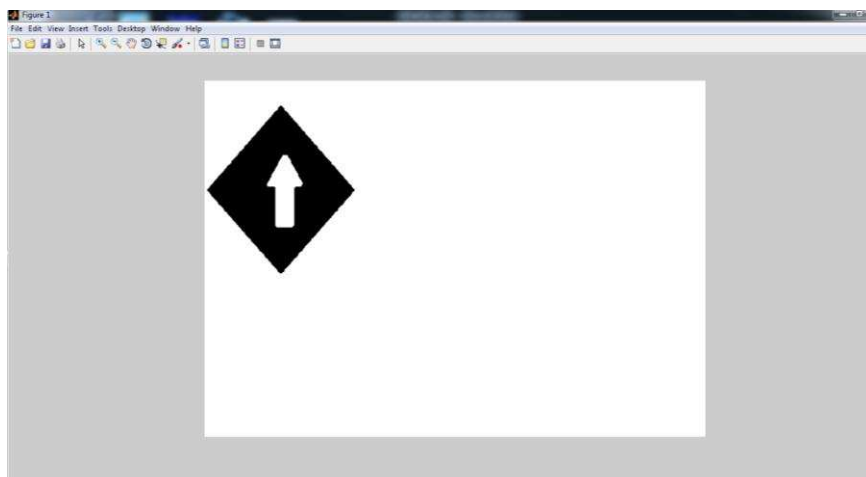
```
>> var = imread('Untitled.png');
```

```
>> imshow(var)
```



```
>> bin7 = im2bw(var);
```

```
>> imshow(bin7)
```



```
>> bord = edge(bin7);
```

```
>> imshow(bord)
```

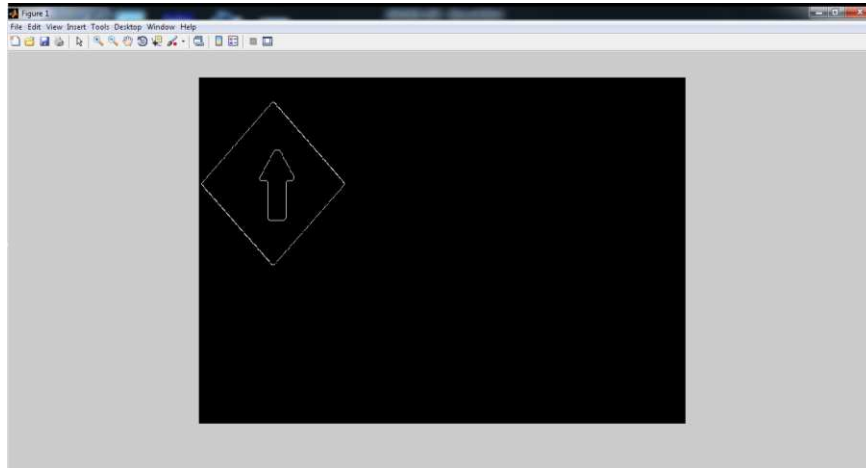


Figura 2.14 Proceso de identificación de contornos con imagen prediseñada

2.2.3 ALGORITMO DE CONTEO DE PERÍMETROS EXISTENTES

Es importante conocer el número de elementos obtenidos como 1L que conforman toda la imagen, para esto se emplea una rutina de conteo de los elementos mediante un barrido de filas con el empleo de un lazo FOR, a la vez se crea una matriz que contenga las coordenadas de la ubicación del mismo, la matriz obtenida contiene únicamente los contornos de la imagen original.

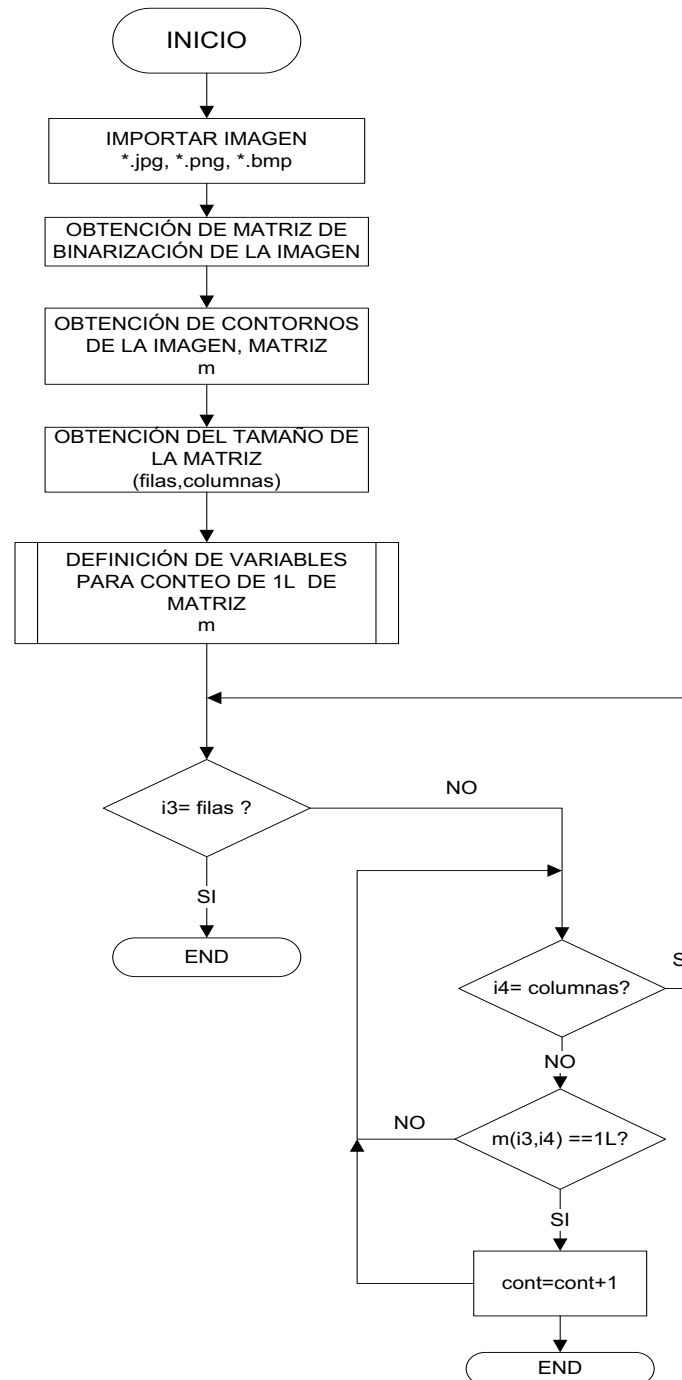


Figura 2.15 Importación de imagen, generación de matriz de 1L y conteo de 1L

Posteriormente se requiere conocer la ubicación del primer elemento no nulo de la matriz, para luego realizar el reconocimiento de las coordenadas de cada uno de ellos que conforman el primer contorno. Este algoritmo se basa en la utilización de un lazo condicional WHILE, de esta manera se realizarán las

repeticiones de lectura de los elementos de la matriz binarizada por medio de un barrido de filas.

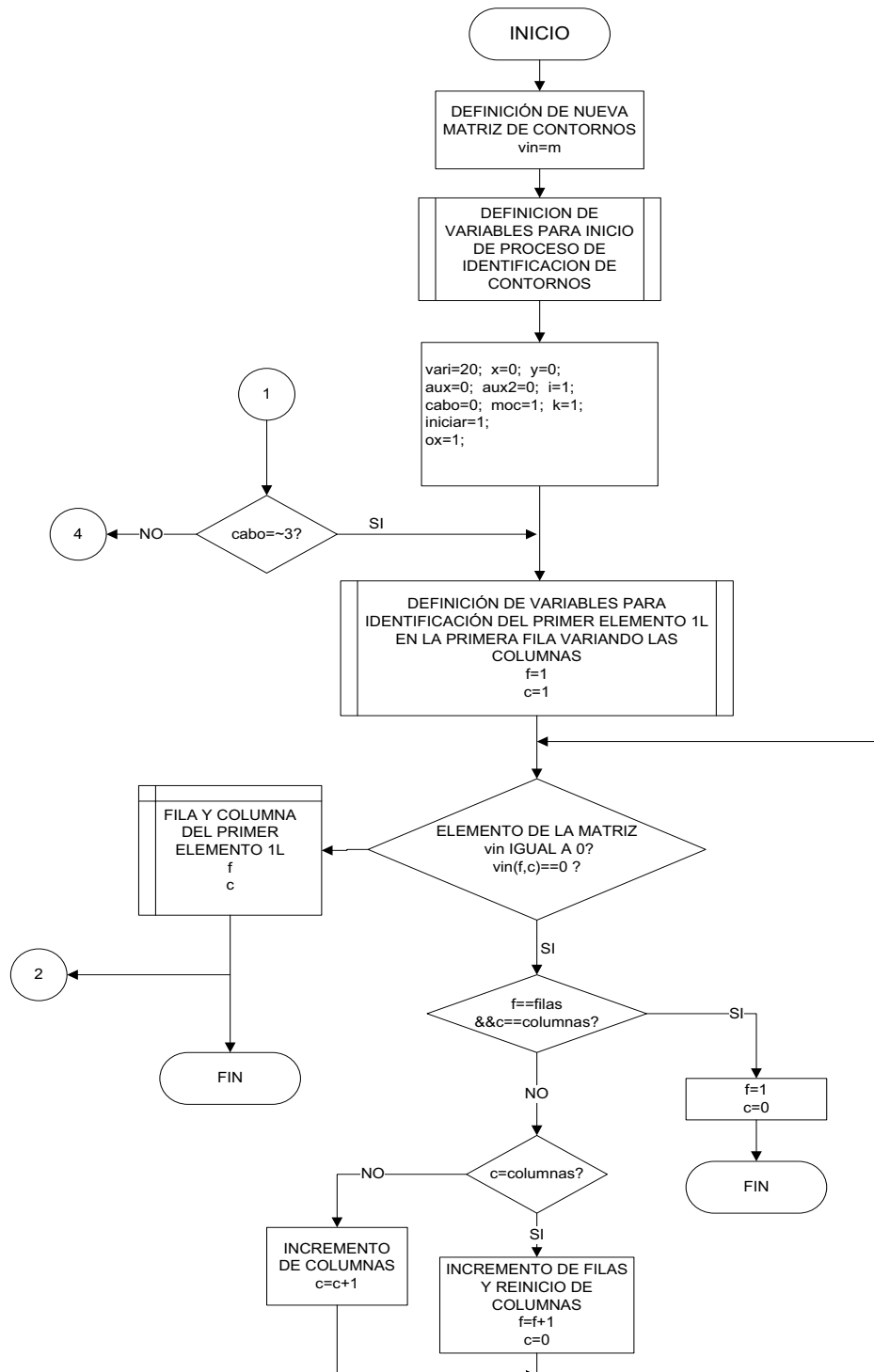


Figura 2.16 Obtención del primer elemento de la matriz de contornos existentes en la matriz de 1L

Se observa que mientras los elementos de la matriz sean igual a cero se realiza la lectura de los mismos mediante el incremento de la posición de las filas, hasta llegar a leer el valor de la última columna en su fila correspondiente, y cumplir que el valor de la fila y columna es igual al valor del tamaño de la matriz.

Conocer la posición del primer y último elemento no nulo de la matriz es importante para estimar las dimensiones que encierran a la figura que se va a cortar.

2.2.4 GENERACIÓN DE MATRICES DE PERÍMETROS

Una vez encontrado el primer elemento no nulo, se realiza un análisis punto a punto del contorno en sentido de las manecillas del reloj, empezando por una posición que corresponde a las 9hrs para de esta manera identificar la continuación del siguiente elemento no nulo con respecto al elemento actual, como se ilustra la Figura 2.9; para esto se crea una matriz “a” de n filas por 2 columnas mediante el uso de un lazo condicional que realice las repeticiones hasta culminar la lectura total de la matriz. Luego de realizar el análisis del elemento actual, se le asigna un valor de cero, de esta manera se elimina el contorno para luego empezar una nueva iteración de la matriz modificada y continuar con un segundo contorno, es decir que también se conoce el número total de contornos a cortar que posee la imagen. Es importante señalar que también se realiza un análisis de una posible discontinuidad del perímetro en cuestión en un radio de 2 píxeles, en este caso las posiciones a las que se refieren con respecto a la posición del elemento actual difieren con ± 2 píxeles.

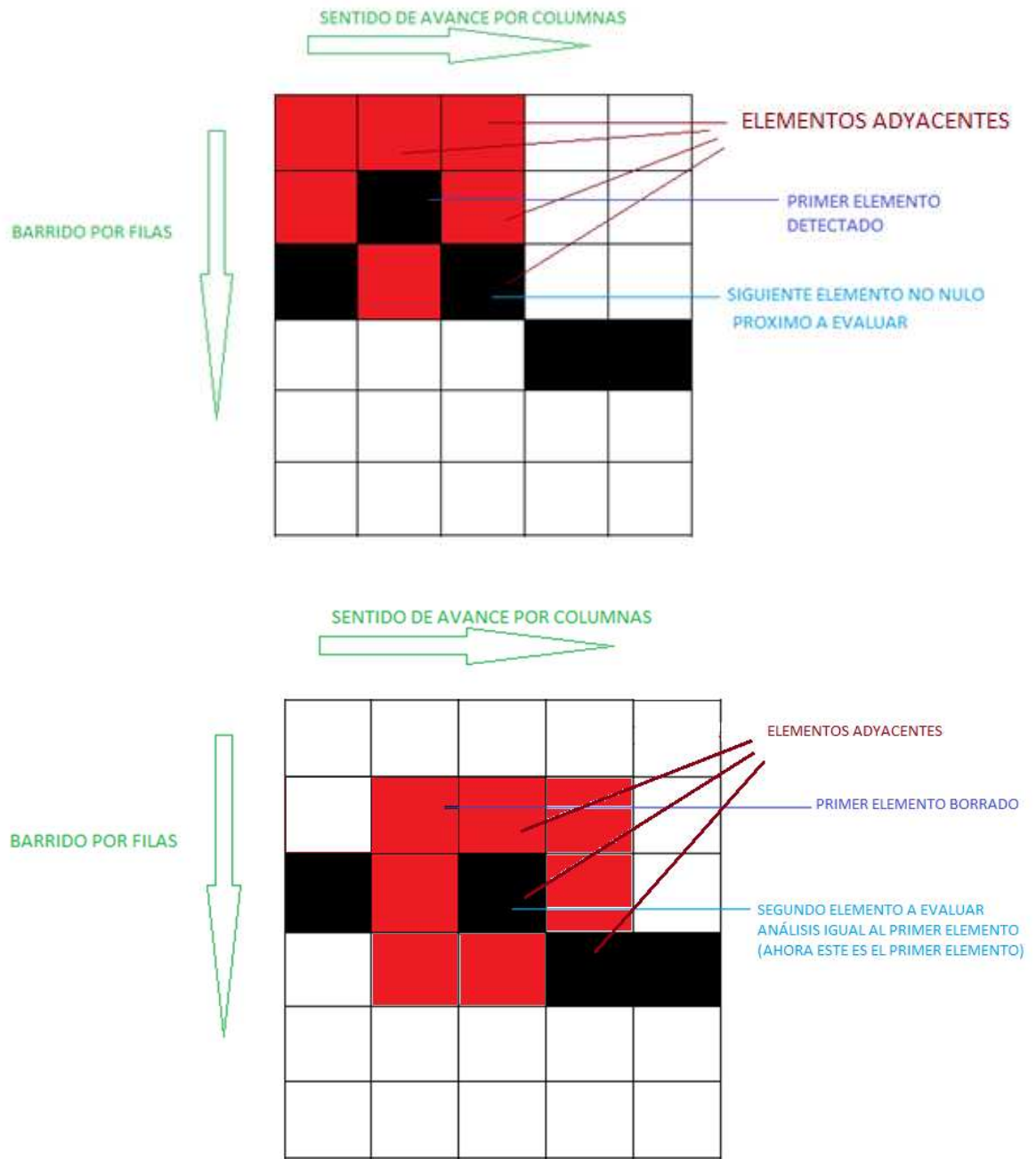


Figura 2.17 Análisis punto a punto del contorno con respecto al primer elemento no nulo identificado

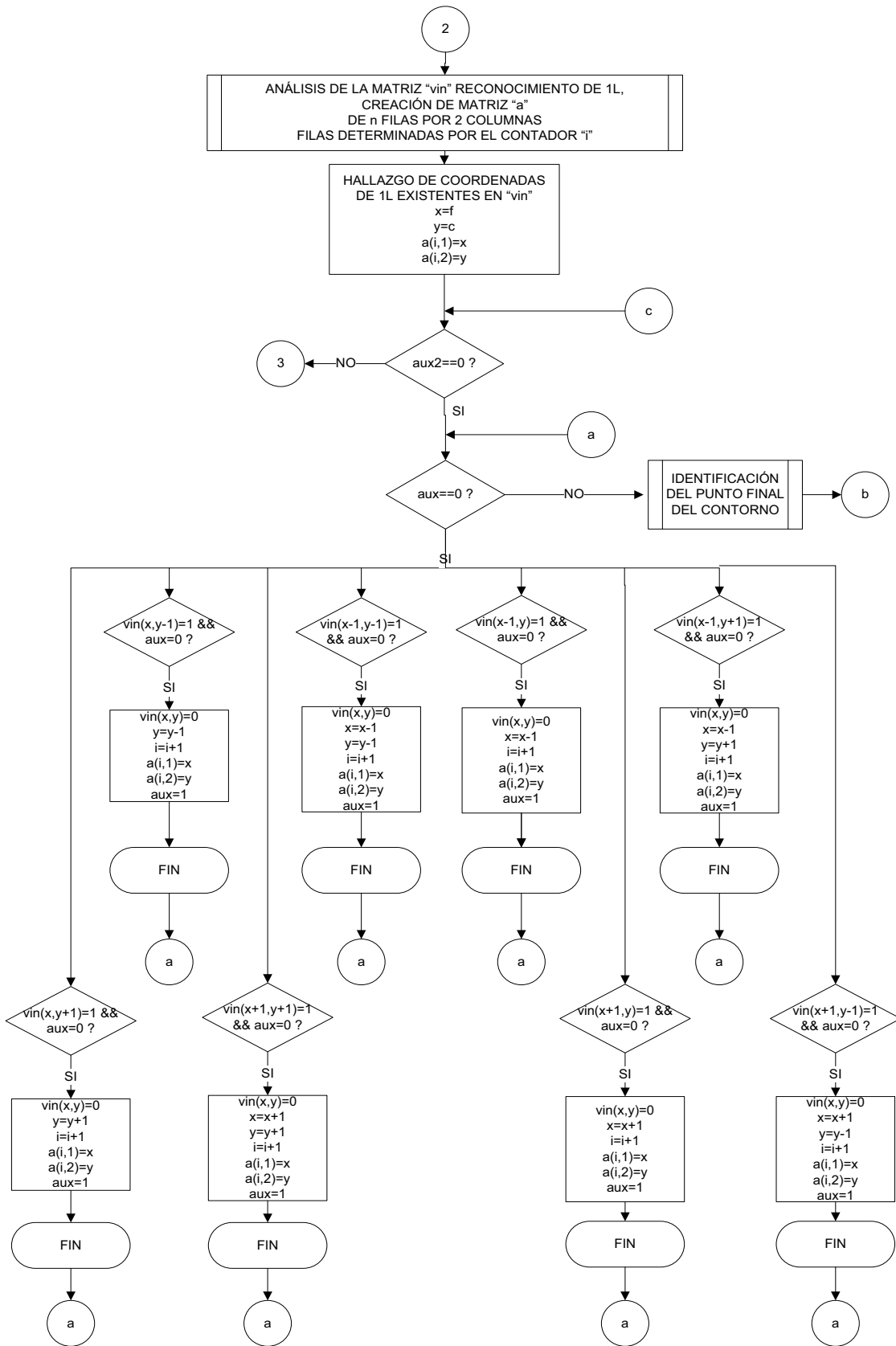


Figura 2.18 Creación de coordenadas de elementos no nulos y eliminación del contorno evaluado

El proceso de eliminación del contorno lleva a un punto en el que debe detectarse el último valor no nulo del contorno que se está tratando.

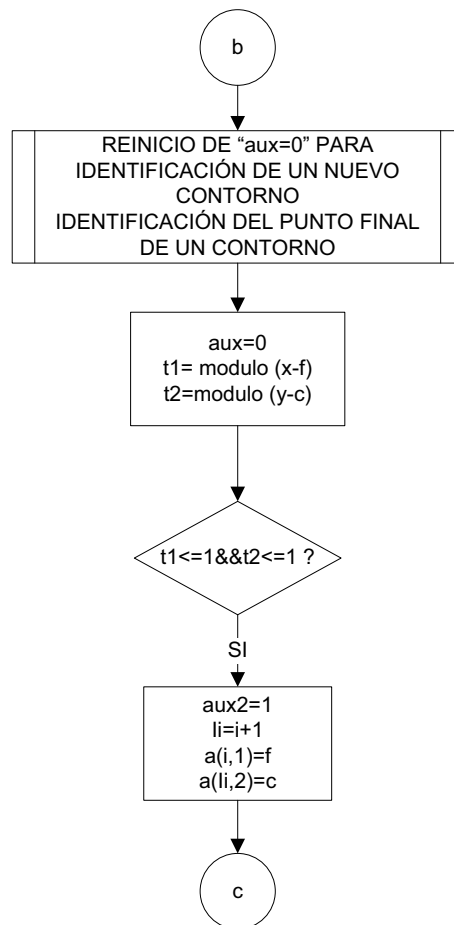


Figura 2.19 Identificación del último elemento de un contorno

2.2.5 GENERACIÓN DE VECTORES INDIVIDUALES DE CADA CONTORNO EXISTENTE EN LA FIGURA

Cada contorno equivale a un vector, por tanto deben crearse los vectores que corresponden a las coordenadas en el plano X-Y, en este caso se generan vectores de acuerdo a un estimado de contornos internos que puede contener una figura con respecto a las necesidades de este proyecto, en este caso se estima un número máximo de veinte contornos. Estos vectores también son necesarios para graficar cada uno de los contornos que conforman la figura.

El algoritmo depende de una variable ox que indica el número de veces que se está generando un contorno y de acuerdo a su valor se generan los vectores k , para la creación de la gráfica de cada contorno, por ejemplo al tratarse de 3 contornos se tiene:

- $ox = 3$;
- 6 vectores generados para la posición en el eje X y eje Y, respectivamente
 - o k_1, k_2
 - o k_3, k_4
 - o k_5, k_6

El siguiente paso es comprobar si existen más contornos en la matriz resultante, después de haber reemplazado los elementos no nulos por ceros, como se explicó en la Figura 2.17.

En la Figura 2.21 se ilustra la asignación de las variables dependiendo del número de contornos, por ejemplo para dos contornos se tiene:

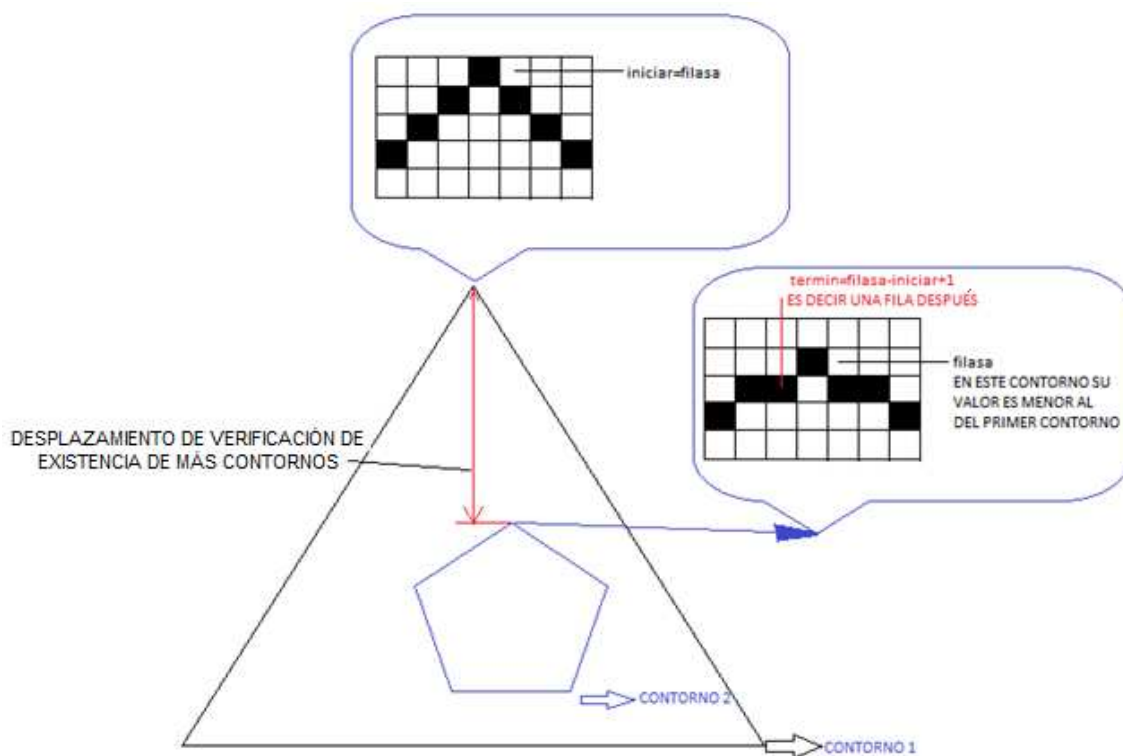


Figura 2.20 Generación de vectores de cada contorno para ploteo en plano X Y

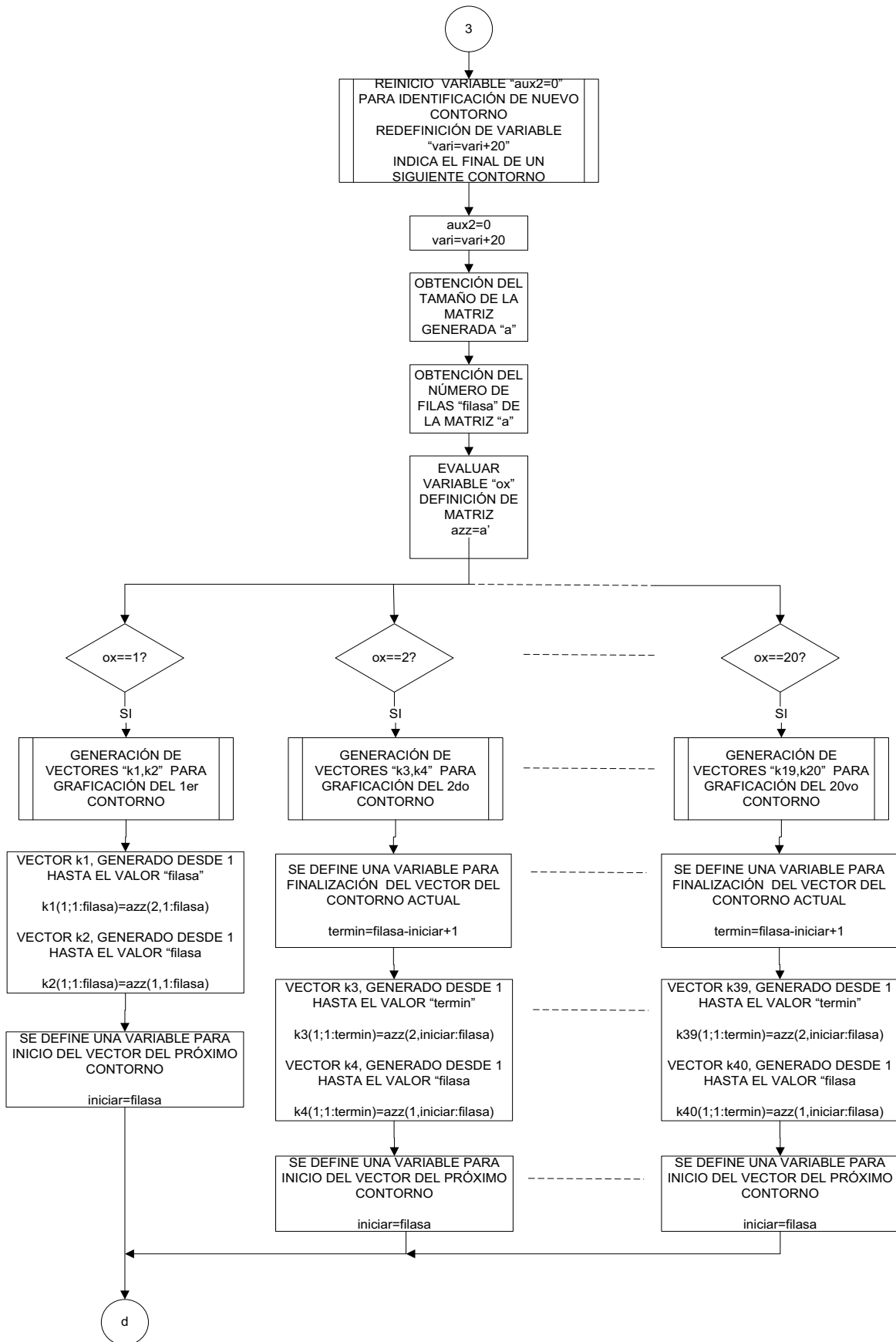


Figura 2.21 Generación de vectores individuales de cada contorno PARTE 1

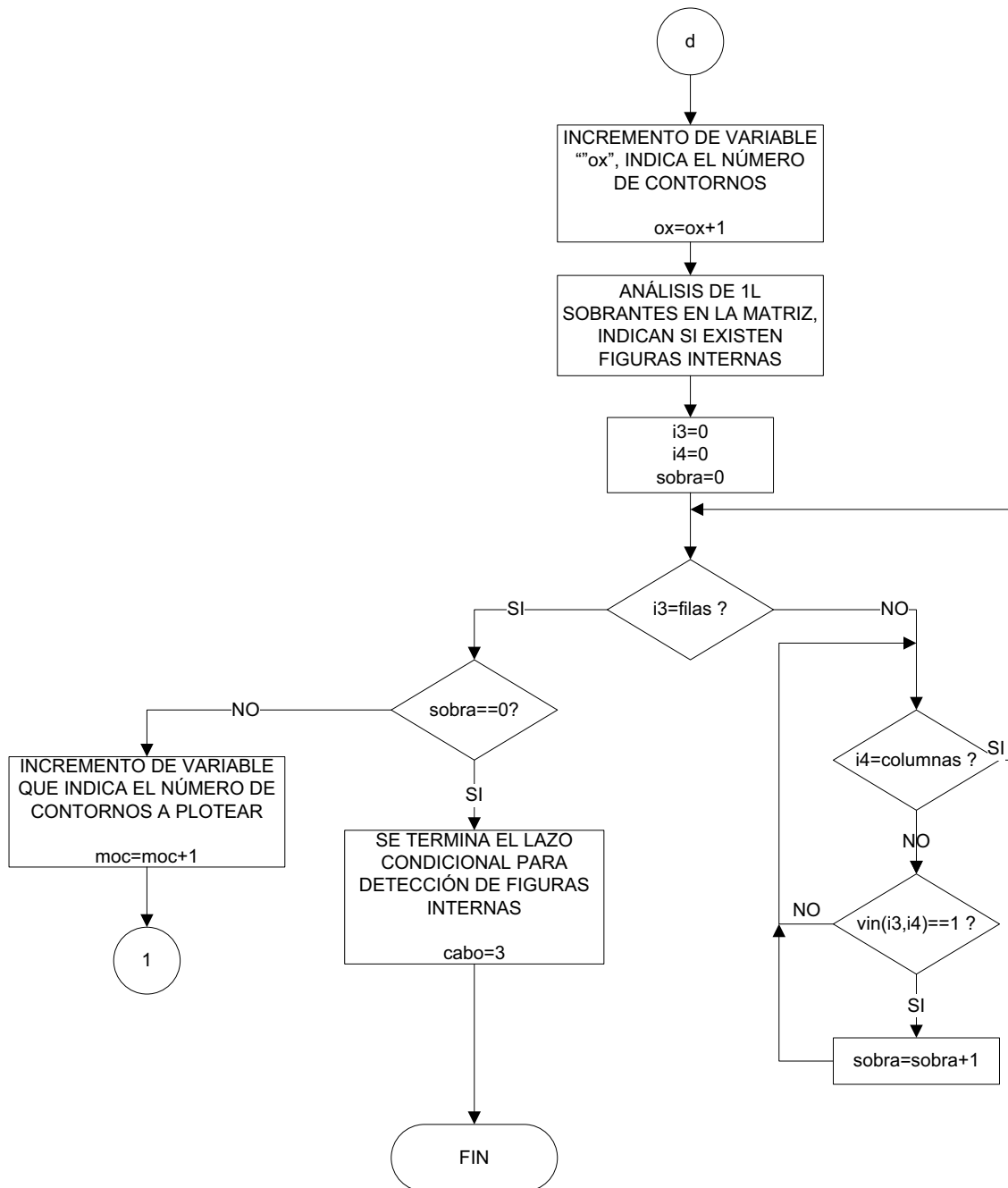


Figura 2.22 Generación de vectores individuales de cada contorno PARTE 2

2.2.6 GRÁFICA DE PERÍMETROS

Para graficar una función o un conjunto de valores en el plano X-Y, el entorno de MATLAB necesita de dos vectores, es decir, uno para cada eje. En este caso al

tratarse de varios contornos existentes en la figura, se realizan 20 casos de ploteo dependiendo del valor de la variable *ox*, MATLAB permite graficar un conjunto de valores para coordenadas en los ejes X-Y con la sentencia:

$$\text{plot}(k_1, k_2, \dots, k_{11}, k_{12}, \dots, \dots)$$

Es decir grafica un conjunto de vectores k_1 vs. k_2 y k_{11} vs. k_{12} .

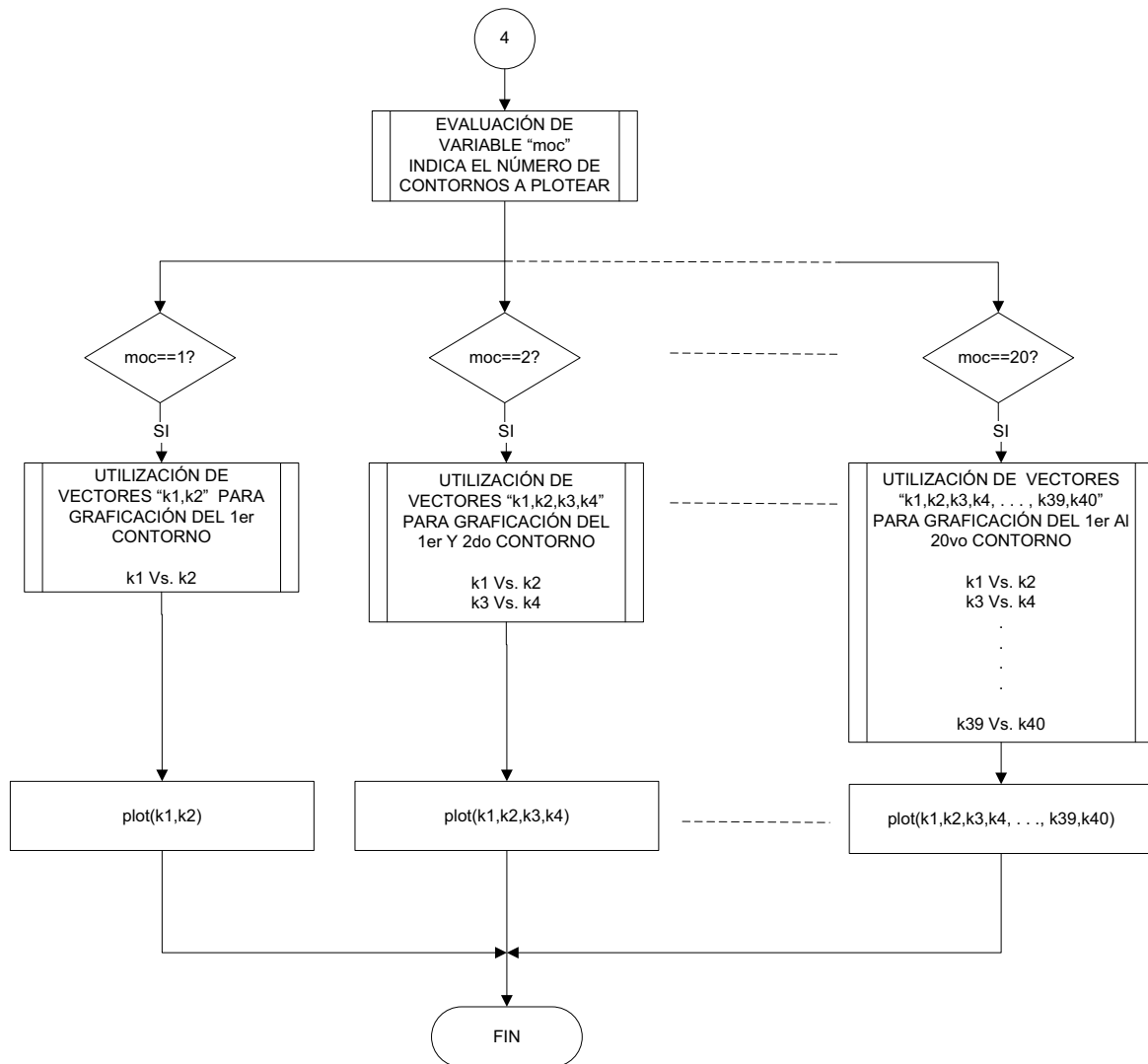


Figura 2.23 Gráfica de cada contorno con los vectores "k" generados

El proceso para graficar el número de contornos existentes en la figura como se observa depende de la variable "*moc*", con los vectores generados en la etapa

anterior se obtiene una gráfica que contiene a cada contorno independiente, como se muestra en la Figura 2.24 cada contorno se grafica de color diferente.

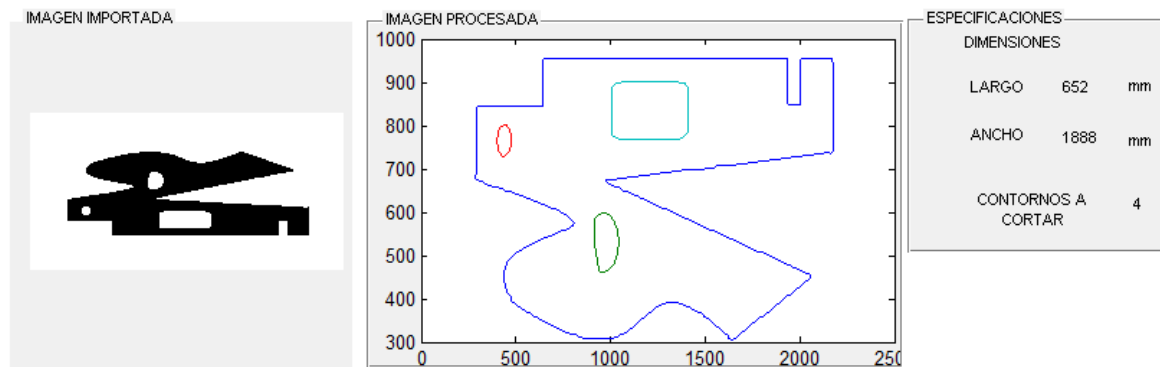


Figura 2.24 Ejemplo del proceso de una imagen y ploteo de contornos independientes

2.3 GENERACIÓN DE CÓDIGO G

El código G determina el tipo de operación realizada en la máquina. Los códigos G son los más importantes en la programación CNC, ya que controlan el modo en que la máquina va a realizar un trazado, o el modo en que va a desplazarse sobre la superficie de la pieza que está trabajando. Seleccionan además, el tipo de movimiento, el modo de velocidad o de avance, etc. En este caso el código necesario para controlar el avance y dirección del oxicorte se basa en el control de dos motores por medio de un puente H, por lo que se necesitan 4 bits de control, los cuales indican la selección del motor a utilizar y el sentido de giro de cada uno.

El método de obtención del código se basa en la Tabla 2.2, donde el elemento a analizar o elemento actual de la matriz se encuentra en el centro de la tabla correspondiente al casillero E. Suponiendo una posición del elemento en la fila 113 y columna 329, al estar el elemento actual en el centro de la tabla la posición siguiente puede ser cualquiera de las posiciones adyacentes al elemento.

	328	329	330
112	A -1,-1	B -1,0	C -1,1
113	D 0,-1	E X	F 0,1
114	G 1,-1	H 1,0	I 1,1

Tabla 2.2 Método de generación de código G

Como se observa, los valores que pueden obtenerse desde el casillero E hacia algún casillero adyacente pueden ser +1,-1 ó 0 en cualquiera de los ejes X o Y, con estos valores se pueden comandar los motores que trabajarán en los dos ejes, análogamente como desplazarse entre filas y columnas, siendo el primer valor el que corresponde al motor que se mueve en el eje Y, y el segundo valor el que comanda el motor del eje X, es decir el binomio representa:

(vertical, horizontal); por ejemplo, si el siguiente punto es el casillero C:

- Se tienen los valores -1,1, es decir:
 - Eje Y: Hay un decremento de -1
 - Eje X: Hay un incremento de 1

Lo que significa que el signo del binomio en cada casillero indica el sentido de giro que el motor debe realizar. Para esto se realiza la siguiente convención de control de motores M1 y M2:

M1	M1	M2	M2
arriba	abajo	izquierda	derecha

Tabla 2.3 Convención de bits de control de motores M1 y M2

Por tanto, se tienen 4 bits de control, que traduciéndolos en valores binarios y decimales se obtiene la siguiente tabla.

CASILLA ADYACENTE	BINOMIO (vertical, horizontal)	EQUIVALENTE BINARIO (M1 M1 M2 M2)	EQUIVALENTE DECIMAL
A	-1,-1	0110	6
B	-1,0	0100	4
C	-1,1	0101	5
D	0,-1	0010	2
F	0,1	0001	1
G	1,-1	1010	10
H	1,0	1000	8
I	1,1	1001	9

Tabla 2.4 Equivalencias decimales de casilleros adyacentes

El código G generado en este caso comanda el posicionamiento y trayectoria del corte, a continuación se presenta un ejemplo.

La matriz de coordenadas en los ejes X-Y que representa la Figura 2.25 está dada mediante una matriz "a= [y1 x1;y2 x2; . . . , yn xn]"

a = [26 34;26 35;26 36;26 37;27 38;28 39;29 40;30 40;31 40;32 40;33 39;34 38;34 37;34 36;34 35;34 34;34 33;33 32;32 31;31 31;30 31;29 31;28 32;27 33;29 37;30 37;31 37;32 37; 32 36;32 35;32 34;31 35;30 36];

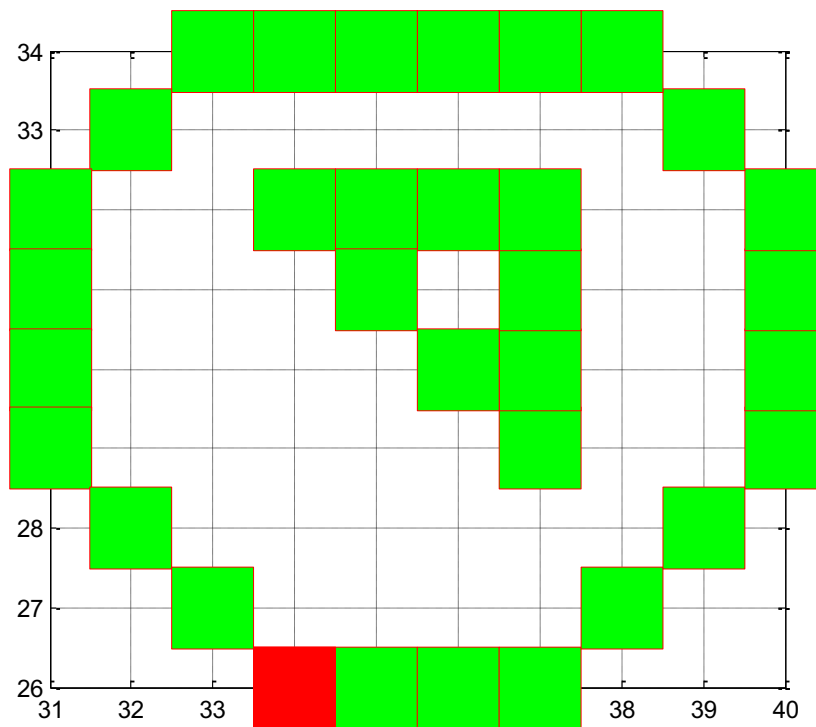


Figura 2.25 Ejemplo de trayectoria con código G

Cada coordenada indica la posición del elemento no nulo, como se observa se generan coordenadas para cada contorno, en este caso existen dos contornos. Con el algoritmo de identificación de la posición del primer elemento, mediante un barrido de filas y variación de columnas, se determina que el primer elemento se encuentra en la posición (26,34); el siguiente paso es determinar la diferencia entre el punto actual y el punto anterior de cada eje, por ejemplo:

Posición	EJE Y	EJE X	DIFERENCIA EJE Y Posición n - Posición n-1	DIFERENCIA EJE X Posición n - Posición n-1
1	31	40		
2	32	40	1	0
3	33	39	1	-1
4	34	38	1	-1

Tabla 2.5 Diferencia entre coordenada actual y coordenada anterior de cada eje

Como se observa, la diferencia entre coordenadas puede ser -1, 0 ó 1, estos valores indican la dirección que se debe tomar para seguir el contorno, es el mismo principio indicado en la Tabla 2.4.

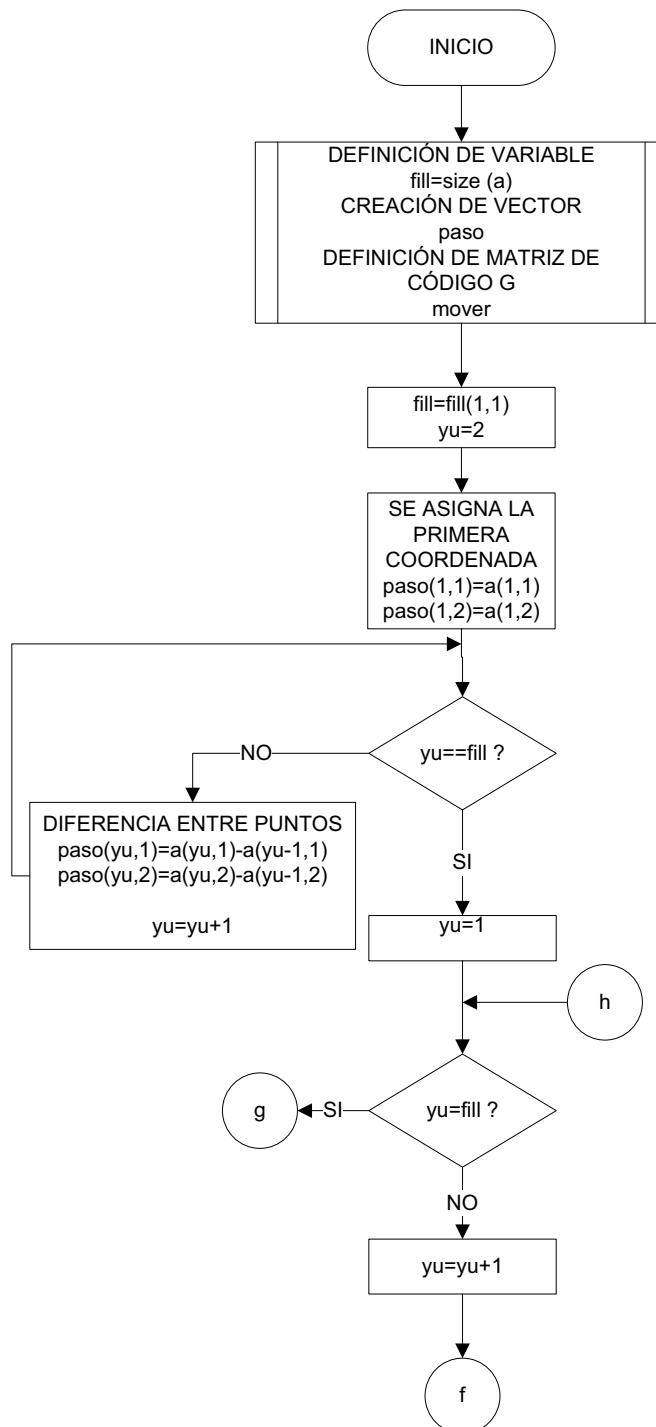


Figura 2.26 Creación de matriz de diferencia entre puntos adyacentes

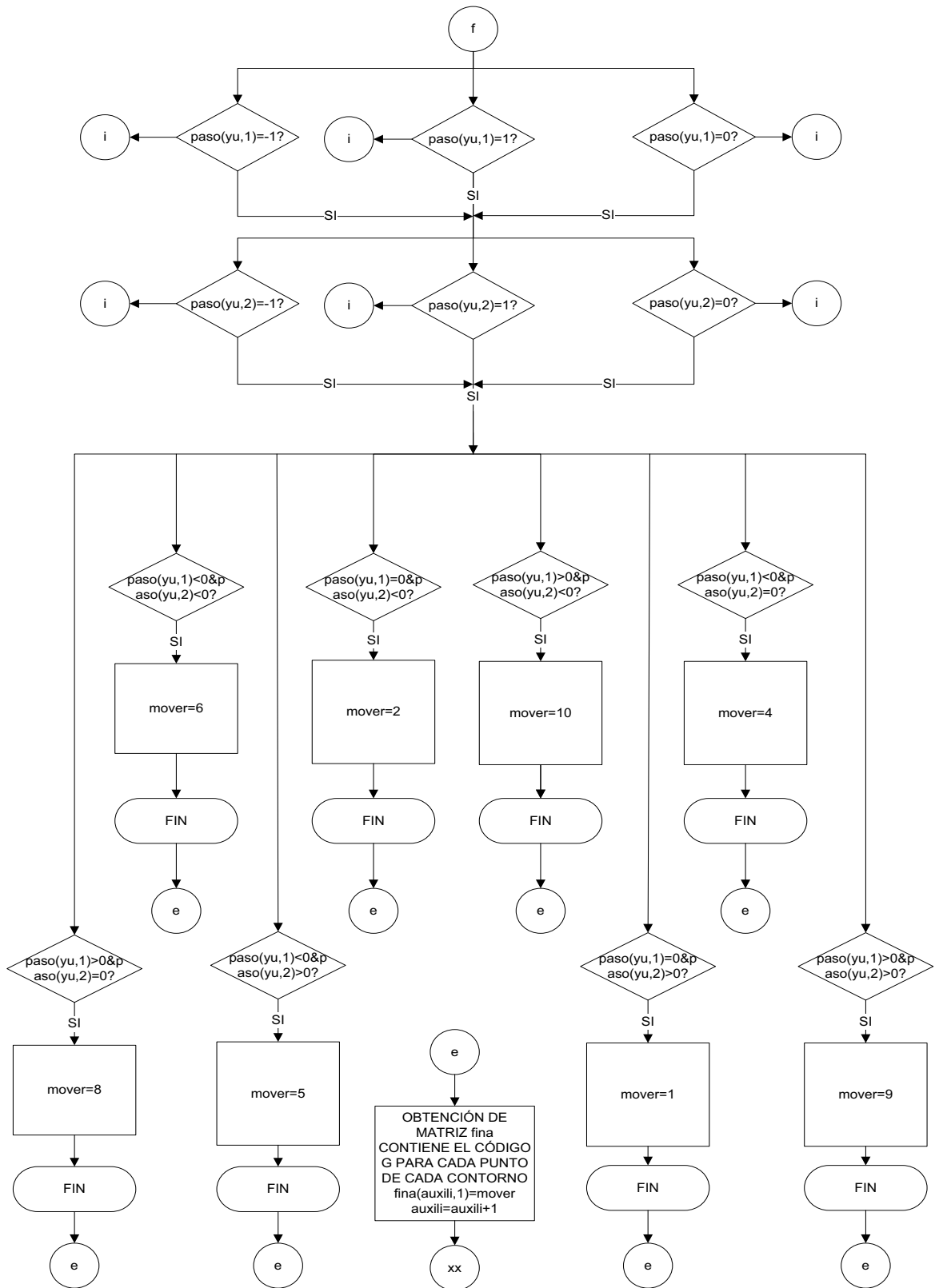


Figura 2.27 Creación de la matriz que contiene el código G, ver Tabla 2.4

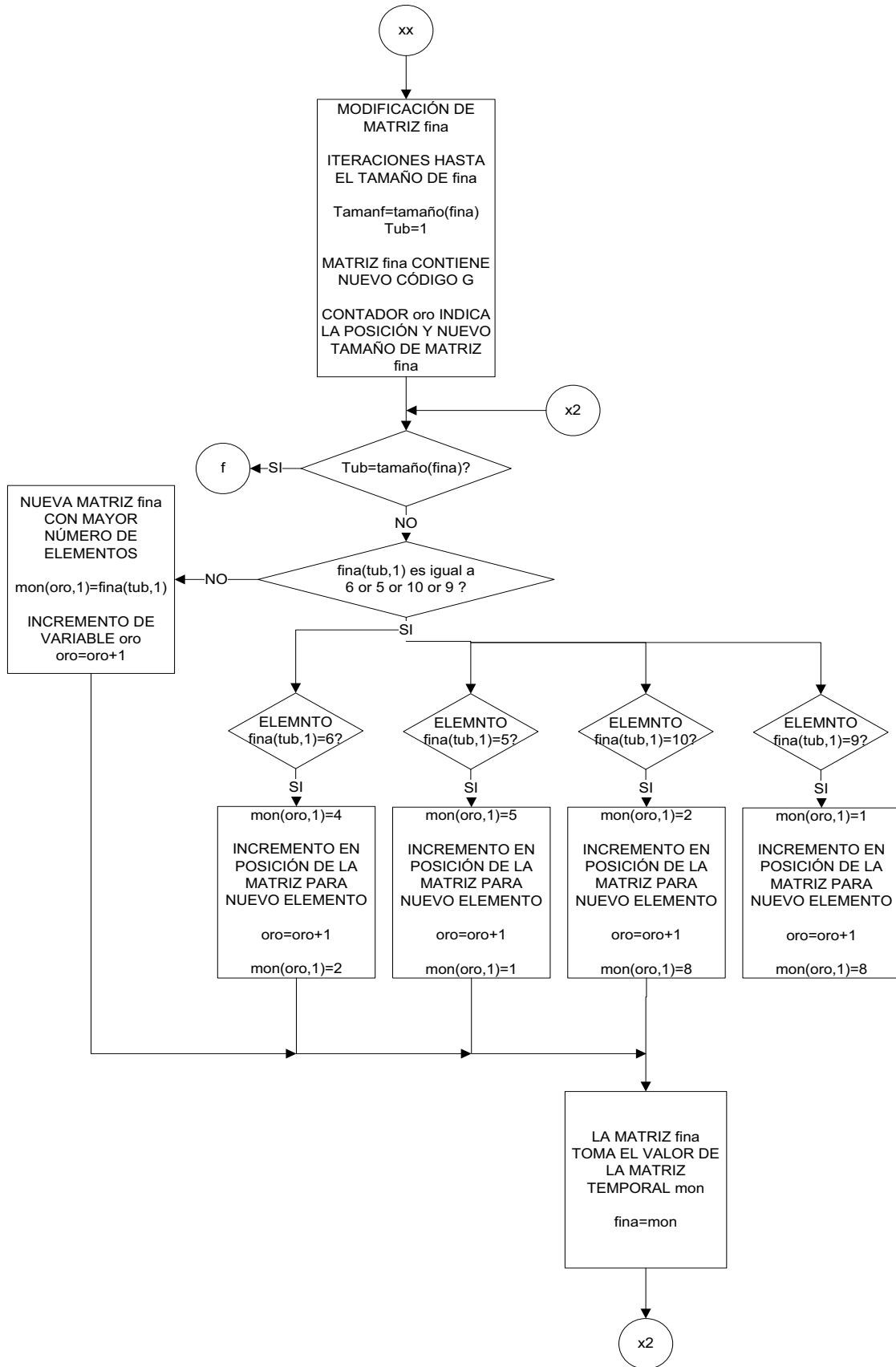


Figura 2.28 Nueva matriz de avance en escalones



Figura 2.29 Creación de código G que indica el posicionamiento hacia un nuevo contorno

En este caso, al ser la diferencia entre los puntos de la matriz de 1L y 0L, distinta de 1, -1 ó 0, se debe dar una orden de paro de motores para parar el soplete de la oxicorte y posicionarse sobre el siguiente contorno.

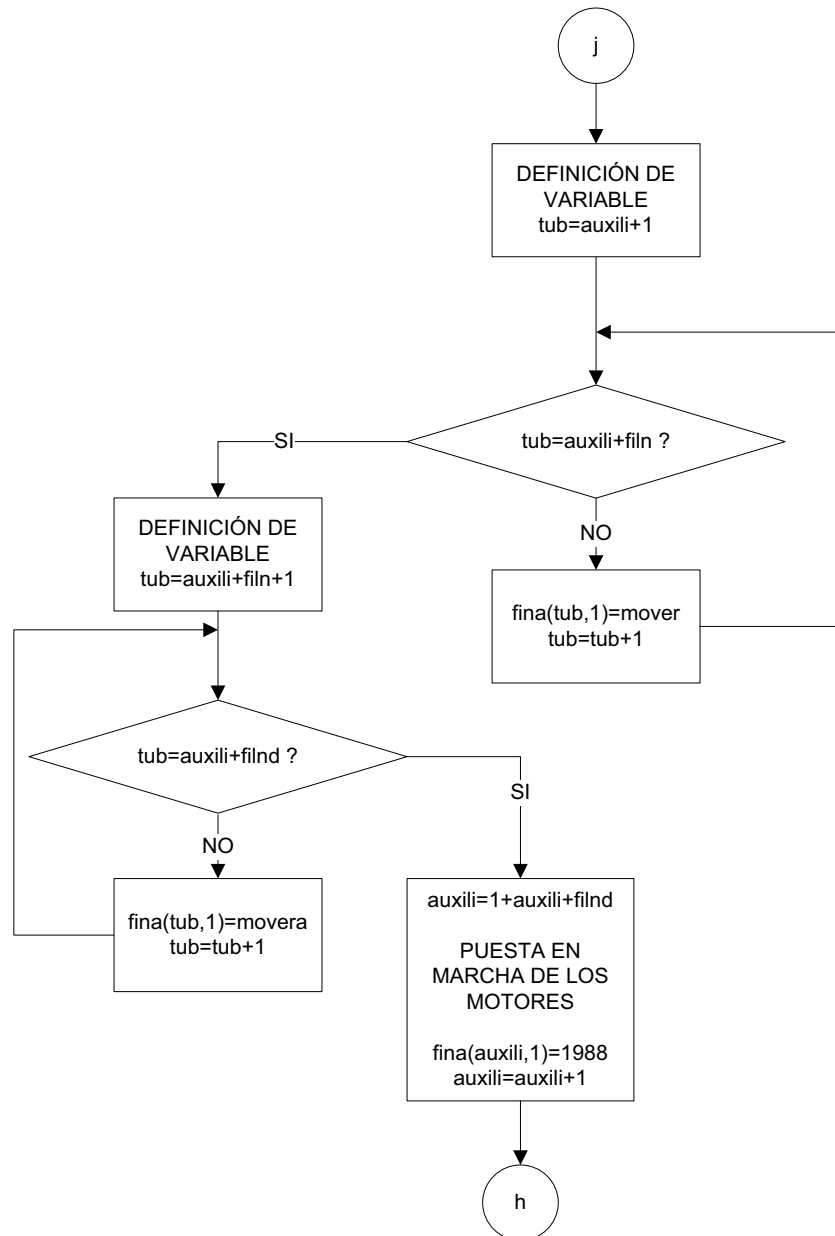


Figura 2.30 Generación de código para posicionamiento en un nuevo contorno y marcha de motores

Una vez culminado el corte del primer contorno y apagados los motores, se genera el código que determina la trayectoria para el posicionamiento en el

segundo contorno, como se explica en la Figura 2.30. El proceso siguiente es enviar los datos generados como código G con 4 bits. Otro factor que se debe tomar en cuenta, es que la matriz generada que contiene el código G, puede tener un contenido con un número de columnas impar, esto se confirma mediante el uso de la función *mod* que indica el residuo de una división, como en este caso, se realiza con el número de columnas. Si el caso fuese tener un número de columnas impar, se aumenta una columna con un valor de cero. En la Figura 2.31 se explica el proceso.

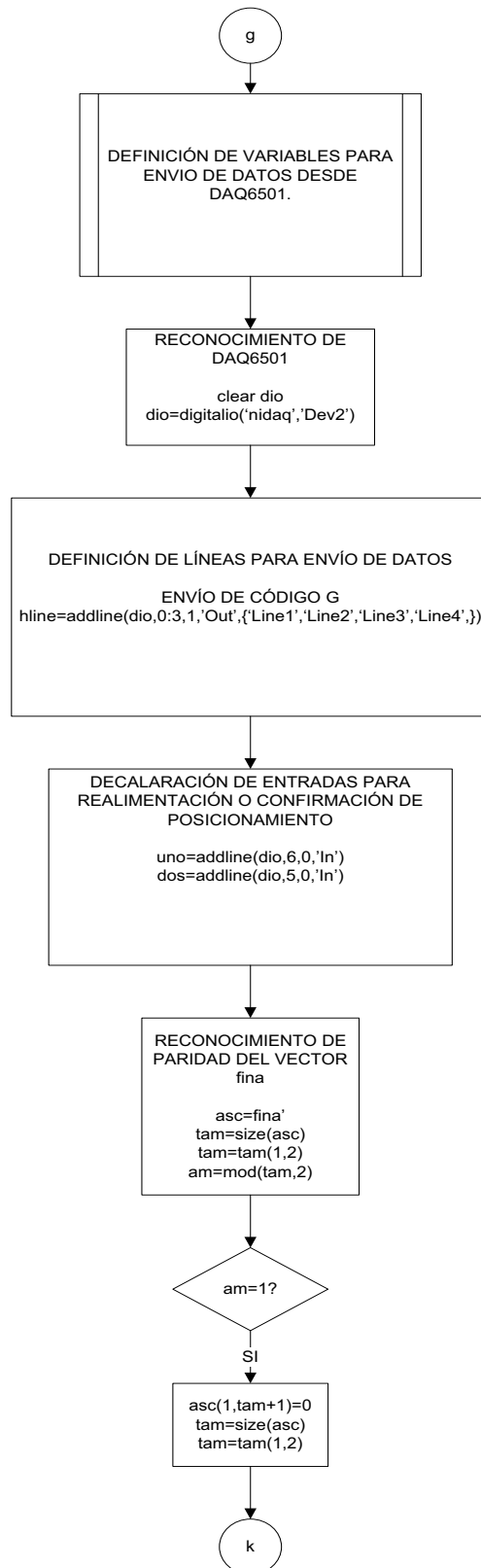


Figura 2.31 Declaración de DAQ y verificación de matriz con número de columnas par

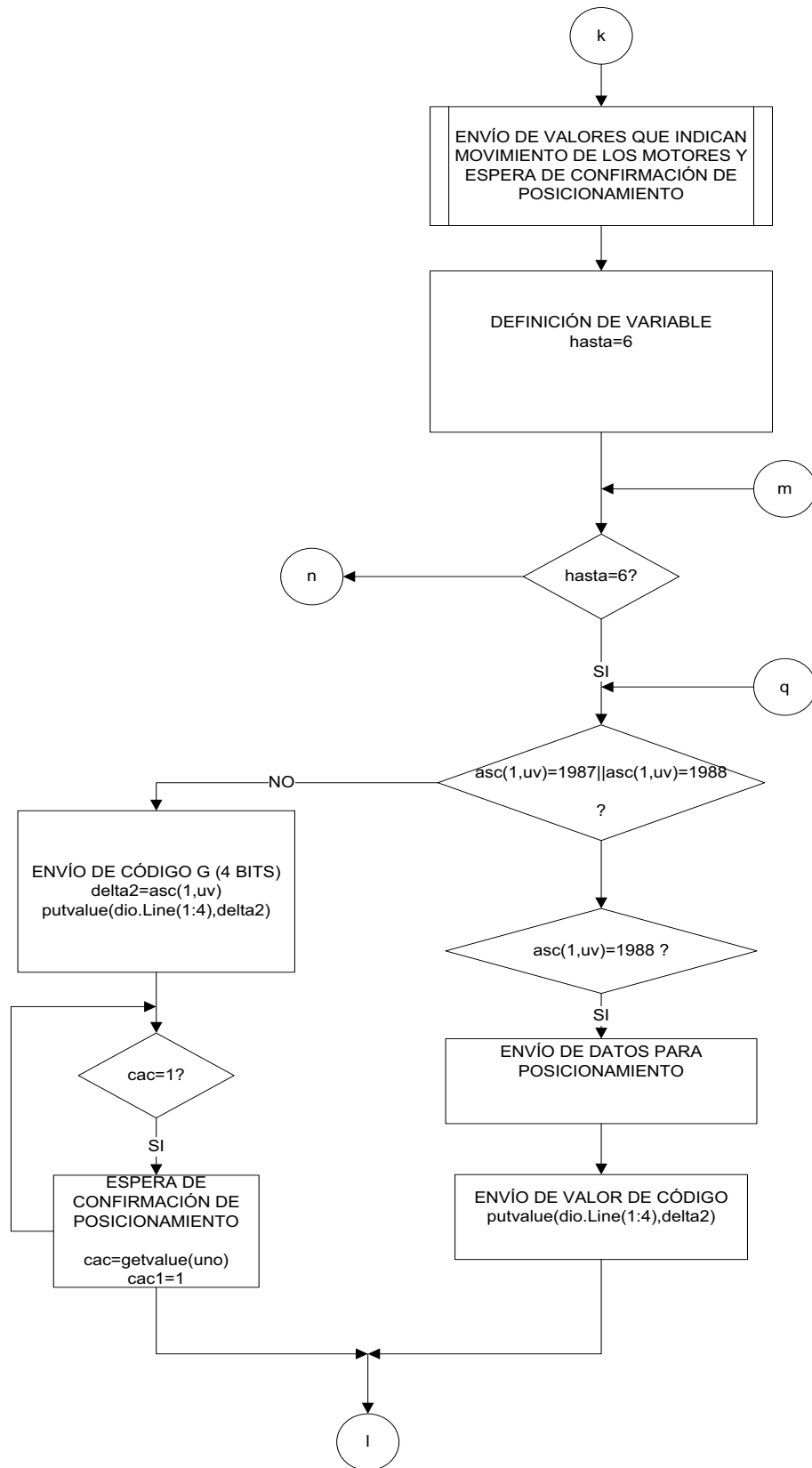


Figura 2.32 Envío de datos

El lazo presentado en la Figura 2.33 indica la espera de la confirmación del posicionamiento para luego enviar un nuevo dato, este proceso se repite hasta terminar el envío de datos que contiene la matriz que contiene el código G.

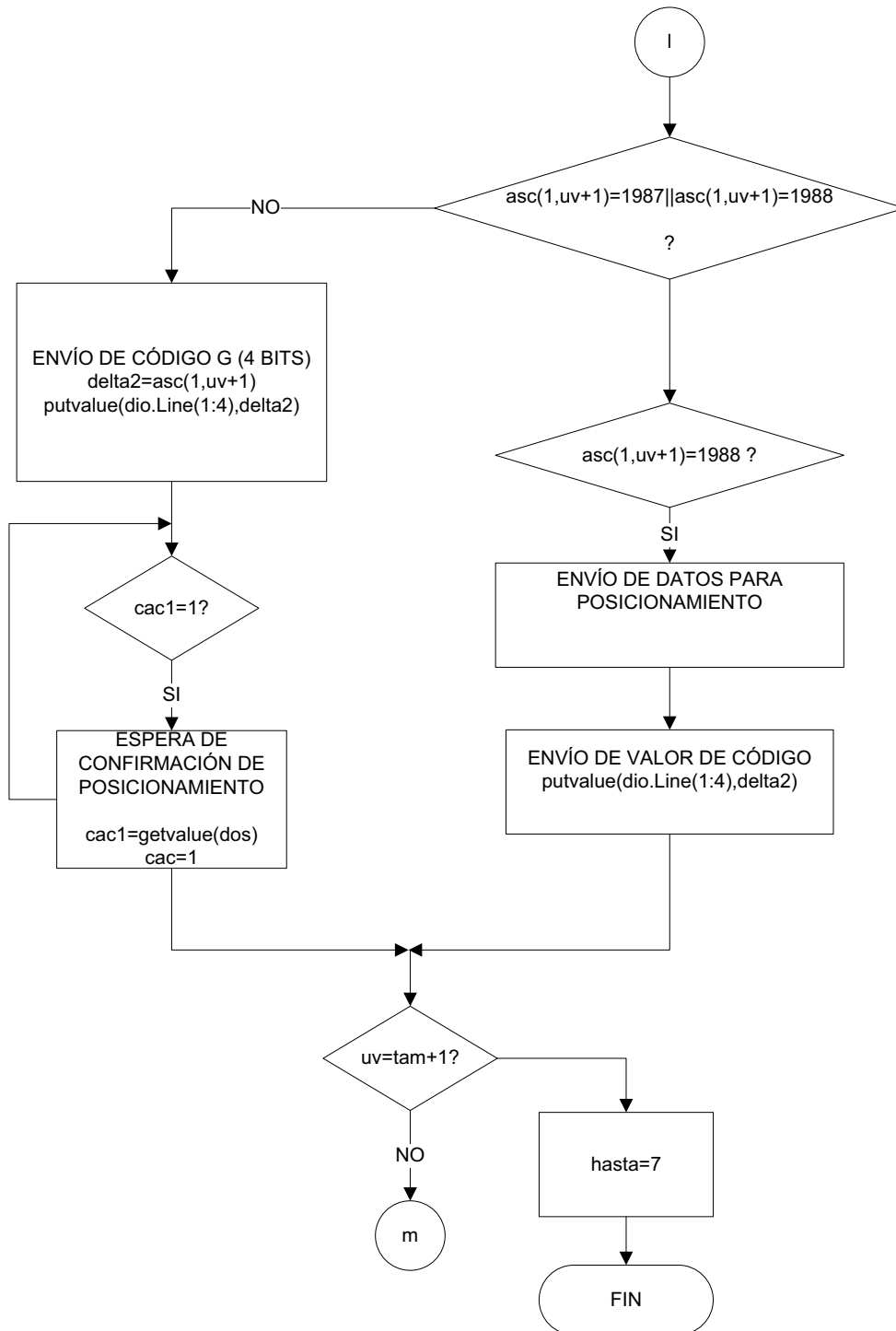


Figura 2.33 Realimentación, se espera la confirmación del posicionamiento

Al haber terminado el corte de todos los contornos, se debe conocer el punto final en el que se encuentra, para luego realizar una rutina de retorno al punto de origen, como se muestra en la Figura 2.34.

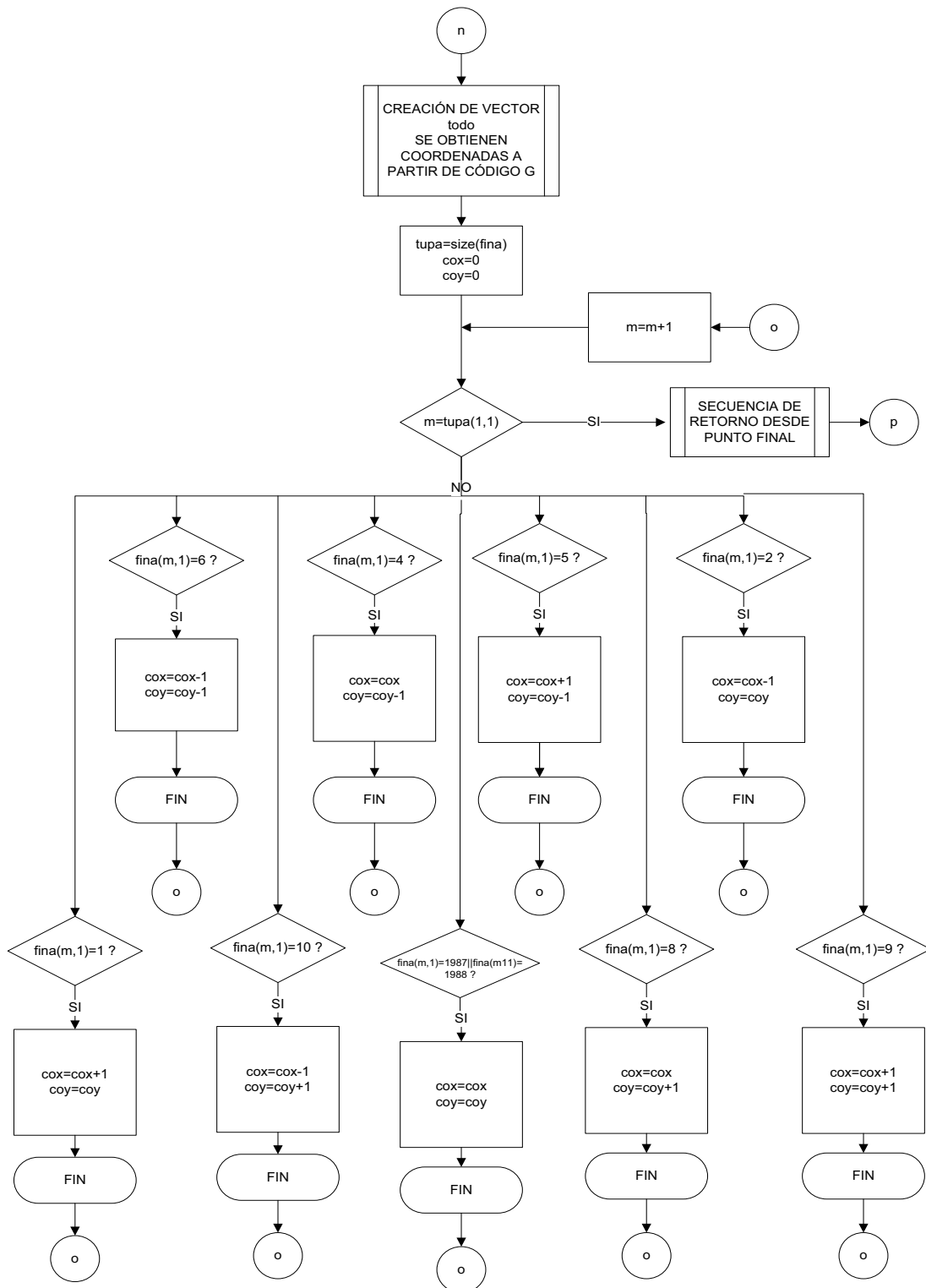


Figura 2.34 Generación de código para retorno

Por último se envía el código necesario para retornar el soplete al punto de partida.

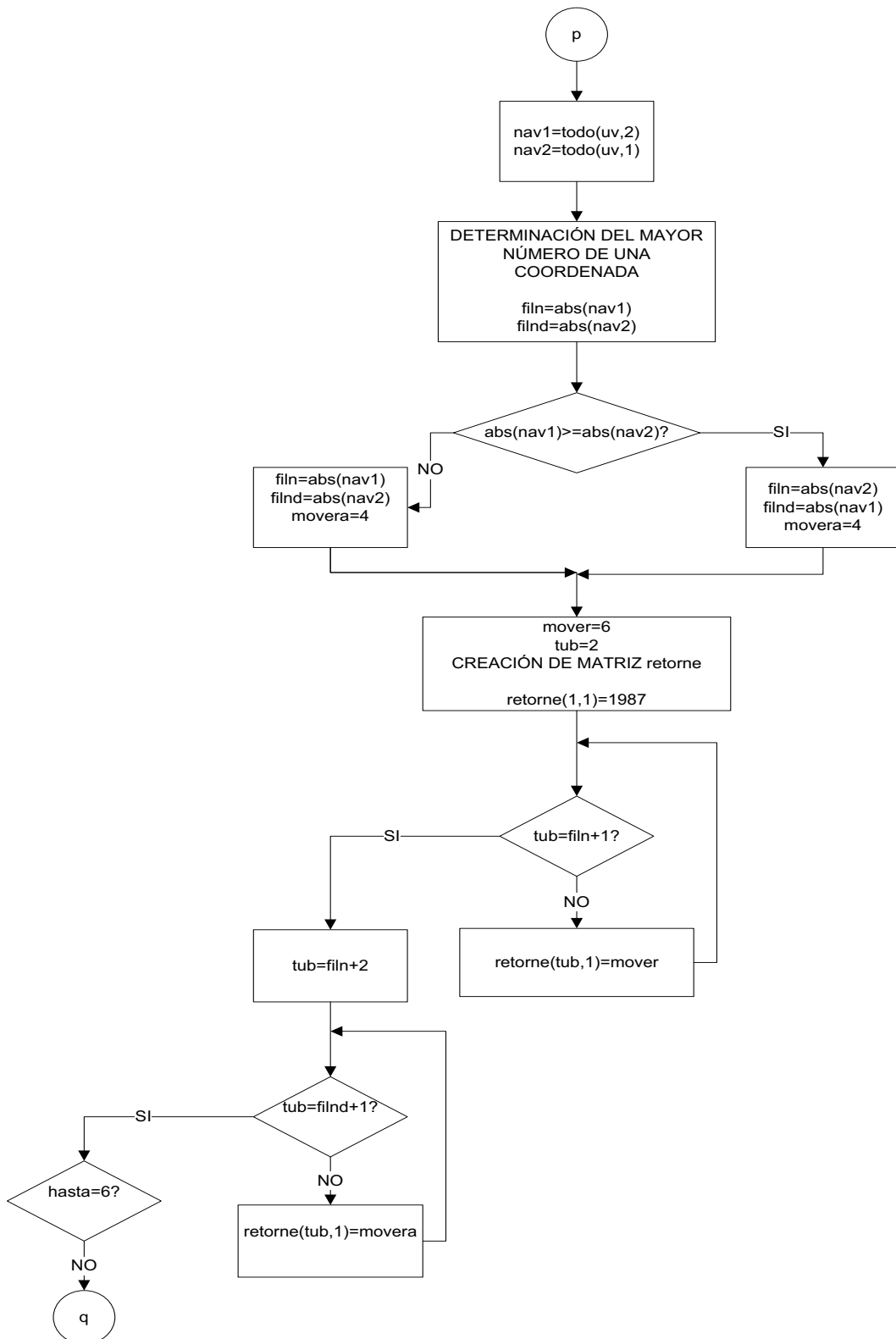


Figura 2.35 Retorno al punto inicial

El objetivo de este proceso es realizar cortes en materiales de espesor entre los 6 mm y 20 mm, por lo que se puede emplear una velocidad de corte fija con regulación de la presión de oxígeno y propano. Otro aspecto a considerar es la definición con la que se realiza el corte, es por esto que se convierten los avances en diagonal a una combinación de avances en escalones, es decir los desplazamientos diagonales son una combinación de dos de los cuatro movimientos básicos.



Figura 2.36 Combinación de desplazamientos para lograr un posicionamiento diagonal al punto actual

2.4 CONTROL DE MOTORES

El desplazamiento entre puntos generados de la imagen a cortar es de una tolerancia de 1 mm. Para esto se convierte el movimiento circular del tornillo sin fin a emplear en movimiento rectilíneo; este desplazamiento es sensado mediante encoders incrementales giratorios. La resolución de estos encoders es de 250 pulsos por revolución, es decir, se obtendría un desplazamiento de 2.54 mm por revolución debido a que el paso del tornillo sin fin es de 10 espiras por pulgada. Para obtener una resolución mejorada deben incrementarse los pulsos por revolución y determinar el número de pulsos que representen un desplazamiento de 1 mm. Los motores deben ser controlados con respecto al número de pulsos que determinan el posicionamiento o desplazamiento requerido, para esto se emplea un puente H de denominación L298N. Otro factor importante es conocer el movimiento que cada motor debe realizar mediante el código G generado, para

esto se emplea un microcontrolador ATTINY 2313 que es el encargado de identificar el código y enviar las señales hacia el puente H.

Uno de los problemas a solucionar es el tiempo que se requiere para enviar una señal de confirmación de posicionamiento hacia la DAQ6501 para que envíe el siguiente valor de código G. Se determina que el valor de esta señal debe ser de 10 ms, y a la vez debe sensarse el número de pulsos generados por los encoders, que son de una frecuencia mucho mayor a la de la señal de confirmación de posicionamiento, es decir, que este microcontrolador no tiene la capacidad de contar el número de pulsos y mantener una señal en alto durante 10 ms, ya que en este tiempo se pierden cuentas de los encoders, por tanto se requiere de un microcontrolador que se encargue del conteo de pulsos y genere una señal de confirmación hacia otro microcontrolador que se encarga de enviar una señal de confirmación por dos pines, en forma alternada, hacia la DAQ6501.

2.4.1 DESCRIPCIÓN DEL HARDWARE DE INTERFAZ

El envío y recepción de señales desde la DAQ6501 hacia los microcontroladores no se realiza directamente, debido a una pérdida del nivel de voltaje de estas señales, por esto se emplea una etapa de acoplamiento mediante opto acopladores de la denominación 4N35, así como también para el control del puente H, debido al ruido generado por los motores.

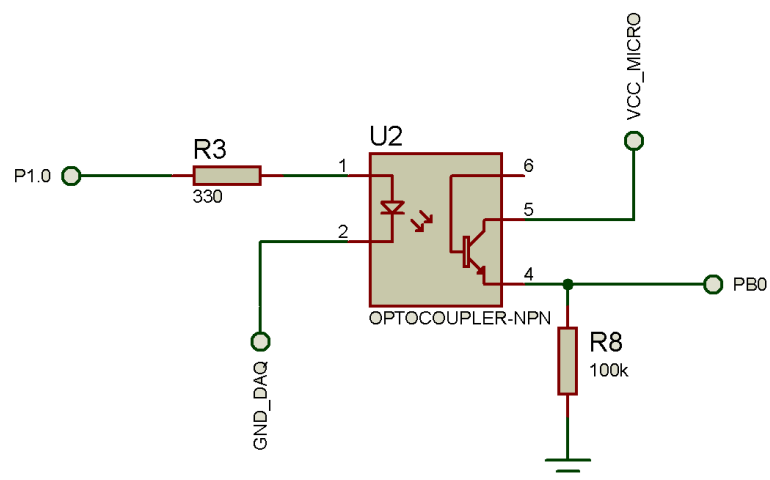


Figura 2.37 Circuito de acoplamiento entre DAQ6501 y ATTINY2313

A continuación se muestra el esquema general del circuito de control que implica el acoplamiento de los microcontroladores con la DAQ6501 y con el puente L298N.

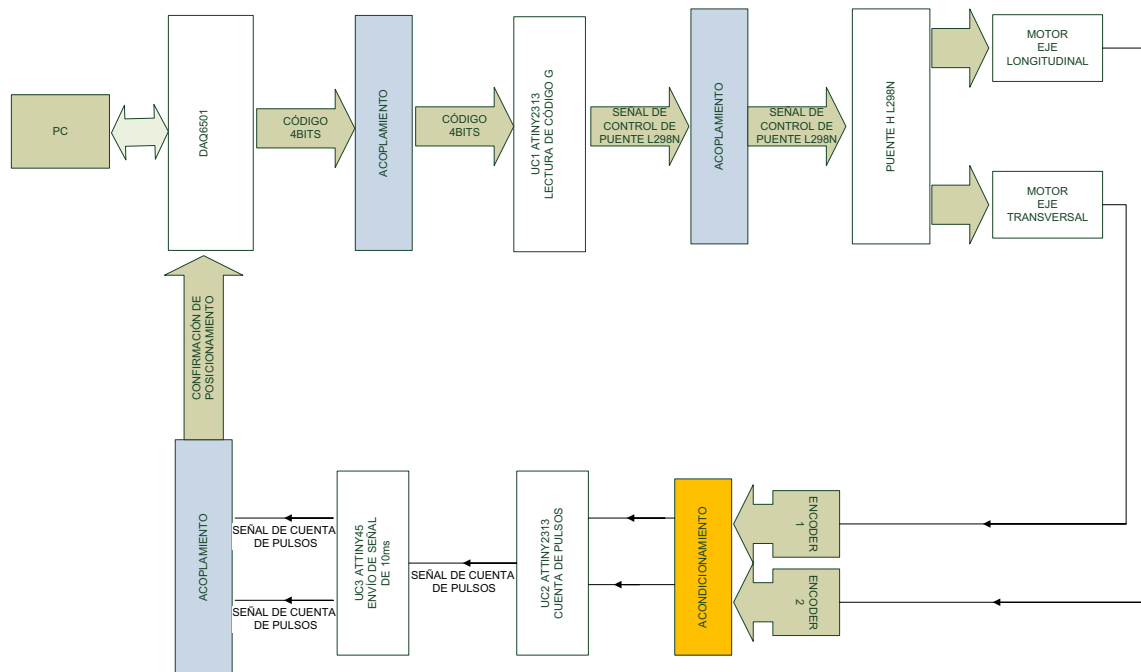


Figura 2.38 Diagrama de bloques de hardware del sistema

2.4.1.1 Acondicionamiento De Señal De Encoders

Los encoders incrementales utilizados constan de un emisor y un receptor, la descripción de los terminales de alimentación y señal, así como también su alimentación se presentan en la Tabla 2.6.

TERMINAL	ALIMENTACIÓN/SEÑAL	DESCRIPCIÓN
BLANCO-ROJO	VCC=12V	LED
BLANCO-AZUL	GND	LED
ROJO	VCC=12V	COLECTORES COMUNES
AMARILLO	OUT	EMISOR
TOMATE	OUT	EMISOR
VERDE	OUT	EMISOR

Tabla 2.6 Terminales de encoders incrementales

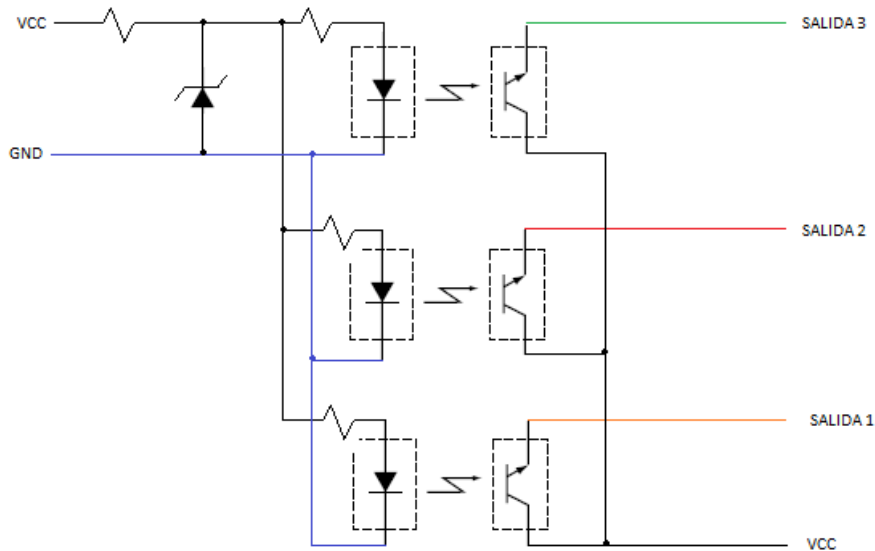


Figura 2.39 Configuración del encoder incremental

Las señales obtenidas en la salida 1 y salida 2 están desfasadas 90° , mientras que la salida 3 emite una señal en cada revolución completa. Cada señal varía dependiendo de la velocidad de giro del encoder, es decir, se obtiene una frecuencia de salida así como también un offset, por lo que es necesario acondicionar dichas salidas para un nivel de señal lógica de 0L y de 1L.

La polarización de los elementos del encoder requieren un nivel de voltaje de 12 Vdc, mientras que las señales que un microcontrolador puede aceptar están en un nivel lógico de 0V y 5V, es por esto que se emplean varias etapas de acondicionamiento, como son:

- Detección de inicio de ciclo
- Reducción a nivel de 0V- 5V y obtención de señal digital.

2.4.1.1.1 Detección De Inicio De Ciclo

La señal generada por los encoders llega hasta un punto máximo que es el voltaje de alimentación y a un nivel mínimo de voltaje que depende de la velocidad de giro del encoder, para detectar el inicio de un ciclo se requiere de una etapa

similar a un detector de cruces por cero, es decir, se emplea una etapa de comparación con el uso de un amplificador operacional LM324, con una referencia que se encuentre entre el valor medio de la señal de salida, de esta manera la señal de salida tendrá un aspecto de señal cuadrada con un nivel entre 0V y 12V.

Se necesitan 2 encoders: uno para el eje longitudinal y uno para el eje transversal, en cada caso se emplean las salidas denominadas salida1 mostradas en la Figura 2.39, que en el caso del circuito de acondicionamiento son las entradas denominadas SEÑAL1 y SEÑAL2. Estas entradas no poseen una resistencia de alto valor para lograr una caída de voltaje apreciable sobre la misma y asegurar una buena señal a la entrada del comparador.

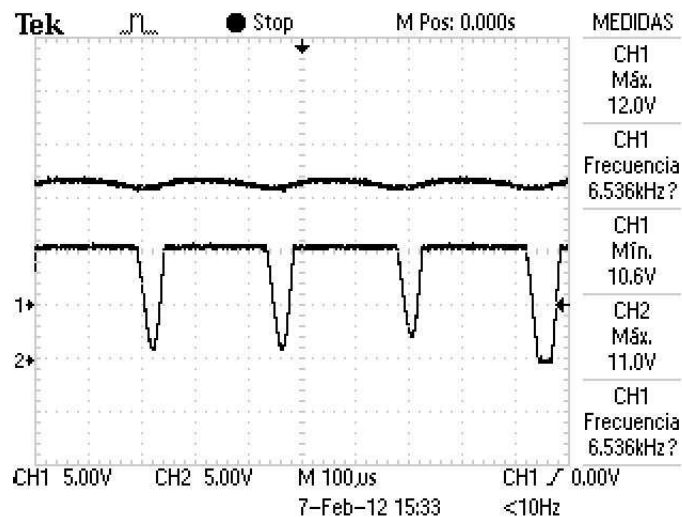


Figura 2.40 Señal de salida de un encoder, señal acondicionada a 12V con comparador

Como se observa en la Figura 2.40, debido a la velocidad de giro de los encoders, se obtienen los valores máximo y mínimo de 12V y 10.6V, respectivamente, es por esto que se realiza una etapa de comparación con un valor de voltaje de 11V, obteniendo una señal de salida entre 0V y el valor del voltaje de saturación del amplificador operacional.

La Figura 2.41 indica las salidas denominadas SEÑAL11 y SEÑAL22 que son señales de un nivel de voltaje entre 0V y 12V correspondientes al ciclo comprendido entre la señal del encoder y el voltaje de referencia que por características de cada encoder se requiere un voltaje de referencia para cada uno.

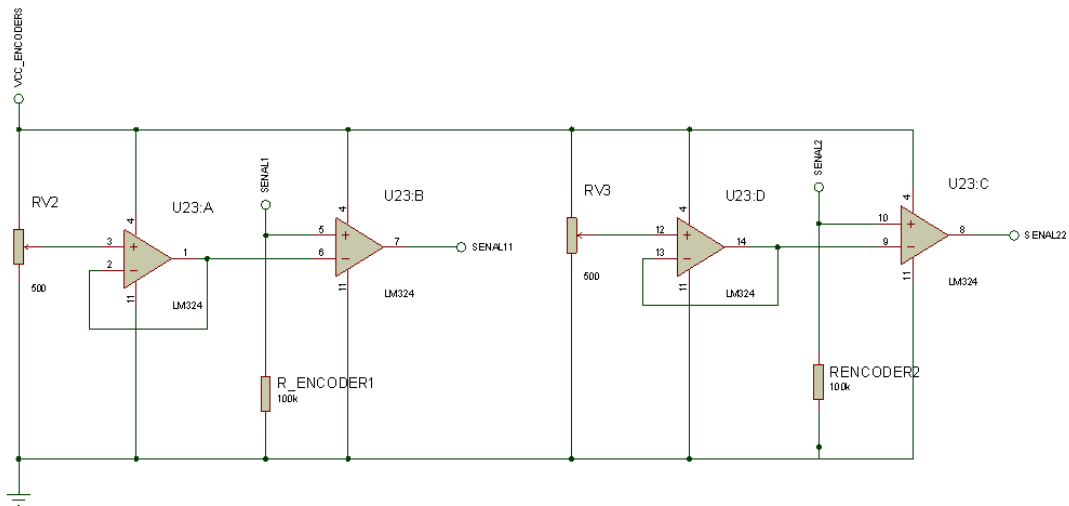


Figura 2.41 Detector de inicio de ciclo

2.4.1.1.2 Reducción a nivel de 0V- 5V y Obtención de señal digital.

El voltaje que se requiere debe ser obtenido mediante un comparador cuyo valor de saturación debe ser 5V, para luego pasar por un Schmitt trigger 74LS14, de lógica negativa, de esta manera la señal digital tendrá un tiempo en alto menor que el tiempo en bajo. Esta etapa requiere de un seguidor emisor.

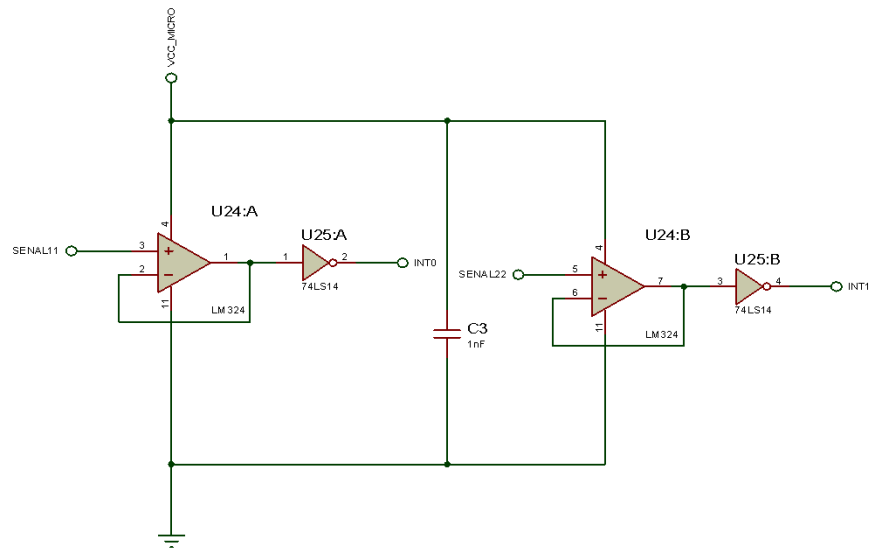


Figura 2.42 Circuito de reducción a 5V mediante un comparador

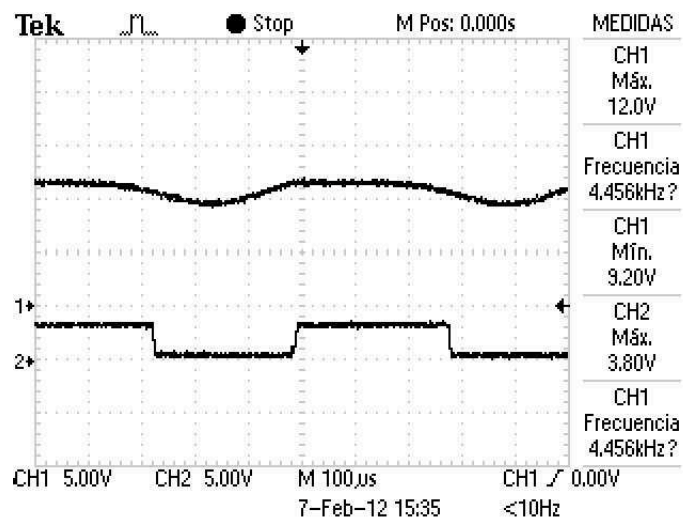


Figura 2.43 Señal de salida a niveles lógicos de voltaje

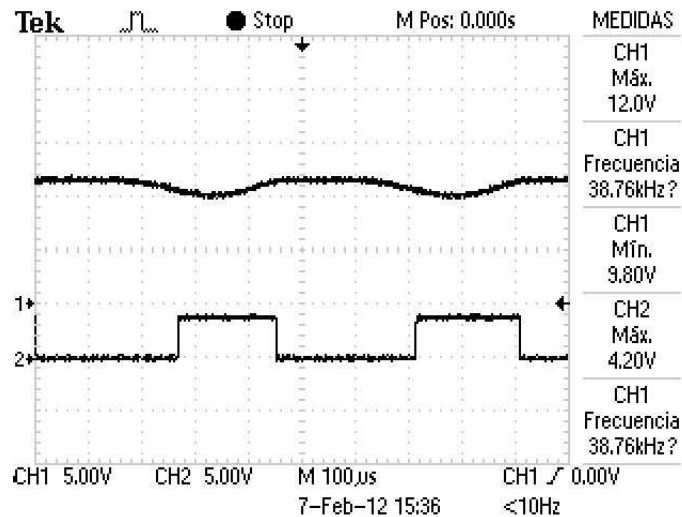


Figura 2.44 Señal digital obtenida después de acondicionamiento de encoders

Las salidas obtenidas INT0 e INT1 son dirigidas al microcontrolador encargado del conteo de pulsos equivalentes a un desplazamiento de un milímetro.

2.4.1.2 Realimentación Hacia Daq6501

Uno de los principales puntos a tomar en cuenta en esta etapa es el tiempo que se requiere mantener una señal en alto para que la DAQ6501 la reconozca y envíe un próximo valor al microcontrolador encargado del control del puente H, es por esto que se emplea un microcontrolador para el conteo de pulsos que indican un desplazamiento de 1 mm y otro que mantenga una señal en alto por un tiempo en el que la DAQ6501 lo pueda reconocer. A continuación se muestra el diagrama de tiempos establecidos para el envío de estados en alto para cada elemento.

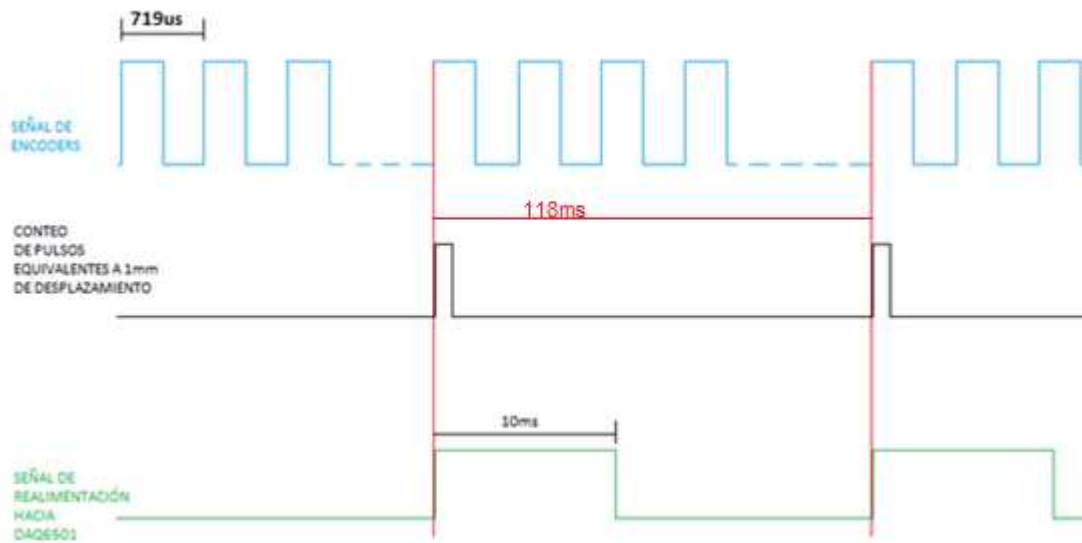


Figura 2.45 Diagrama de tiempos en alto para realimentación con la DAQ6501

Los encoders incrementales se encuentran acoplados a los tornillos mediante piñones que tienen 30 y 50 dientes, respectivamente, la velocidad de los motores cuando la relación de trabajo del PWM es 1, es de 200 rpm, por tanto la frecuencia de los pulsos de los encoders es encontrado por medio de las siguientes relaciones.

$$\omega_1 \times n_1 = \omega_2 \times n_2$$

$$200 \times 50 = \omega_2 \times 30$$

Donde:

ω : Velocidad [rpm]

n : Número de dientes de los piñones

$$\omega_2 = \frac{333.33 \text{ [rpm]}}{60} = 5.56 \text{ [rev/seg]}$$

Los encoders tienen una resolución de 250 pulsos por revolución por tanto:

$$f = \omega_2 \times N$$

Donde:

f : Frecuencia de pulso

N : Número de pulsos por revolución

$$f = 5.56 \times 250 = 1390 \text{ Hz}$$

Teniendo un periodo de:

$$T = \frac{1}{f} = 719.4 \mu\text{s}$$

Al hacer una relación entre los dos piñones por una revolución se tiene:

$$\omega_1 \times n_1 = \omega_2 \times n_2$$

$$1 \times 50 = \omega_2 \times 30$$

$$\omega_2 = \frac{5}{3} [\text{rev}]$$

Multiplicando este valor por la resolución del encoder, se obtiene el número de pulsos por cada revolución del tornillo sin fin, por tanto éste es el número de pulsos equivalente a un desplazamiento de 2.54 mm.

$$N = 250 \times \omega_2$$

$$N = 250 \times \frac{5}{3} \approx 417 \text{ pulsos/rev}$$

Es decir, se tienen 417 pulsos en un desplazamiento de 2.54 mm, por tanto el número de pulsos por cada milímetro es:

$$N_{1\text{mm}} = \frac{N}{2.54} = \frac{417}{2.54} \approx 164 \text{ pulsos}$$

Por lo que, el tiempo entre el envío y confirmación de un nuevo dato de la DAQ6501 es de:

$$t = T \times N_{1\text{mm}} = 719.4 \mu\text{s} \times 164 \approx 118 \text{ ms}$$

Con este tiempo de 118 ms, se asigna 10 ms de tiempo en alto para la realimentación hacia la DAQ.

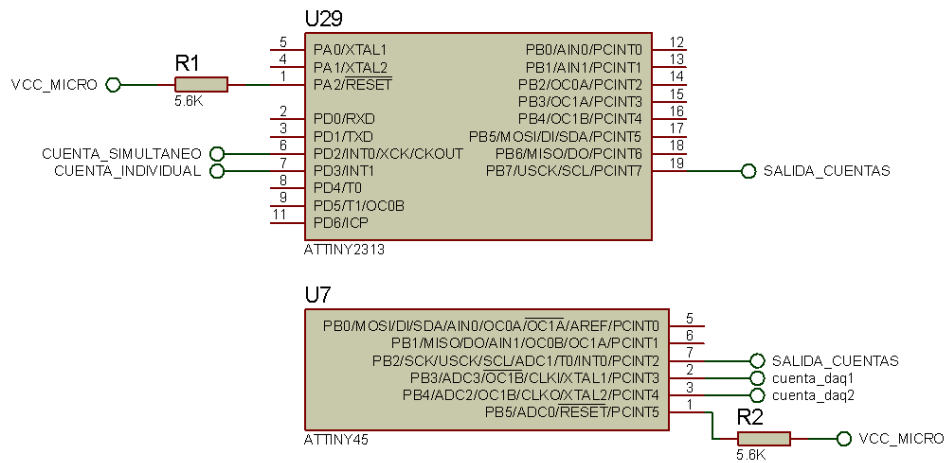


Figura 2.46 Hardware de realimentación hacia DAQ6501

Para este propósito se emplea un microcontrolador ATTINY2313 que es el encargado de realizar el conteo de pulsos requeridos para el desplazamiento de 1 mm. Esto se realiza mediante interrupciones INT0 e INT1 que corresponden a las salidas del Schmitt trigger, para luego enviar una señal SALIDA CUENTAS al microcontrolador ATTINY45, que tiene una sola interrupción. Este es el encargado de mantener una señal en alto durante 10 ms *cuenta_daq1* y *cuenta_daq2* hacia la DAQ6501 en forma alternada, para luego enviar un nuevo valor hacia el microcontrolador encargado del control de los motores.

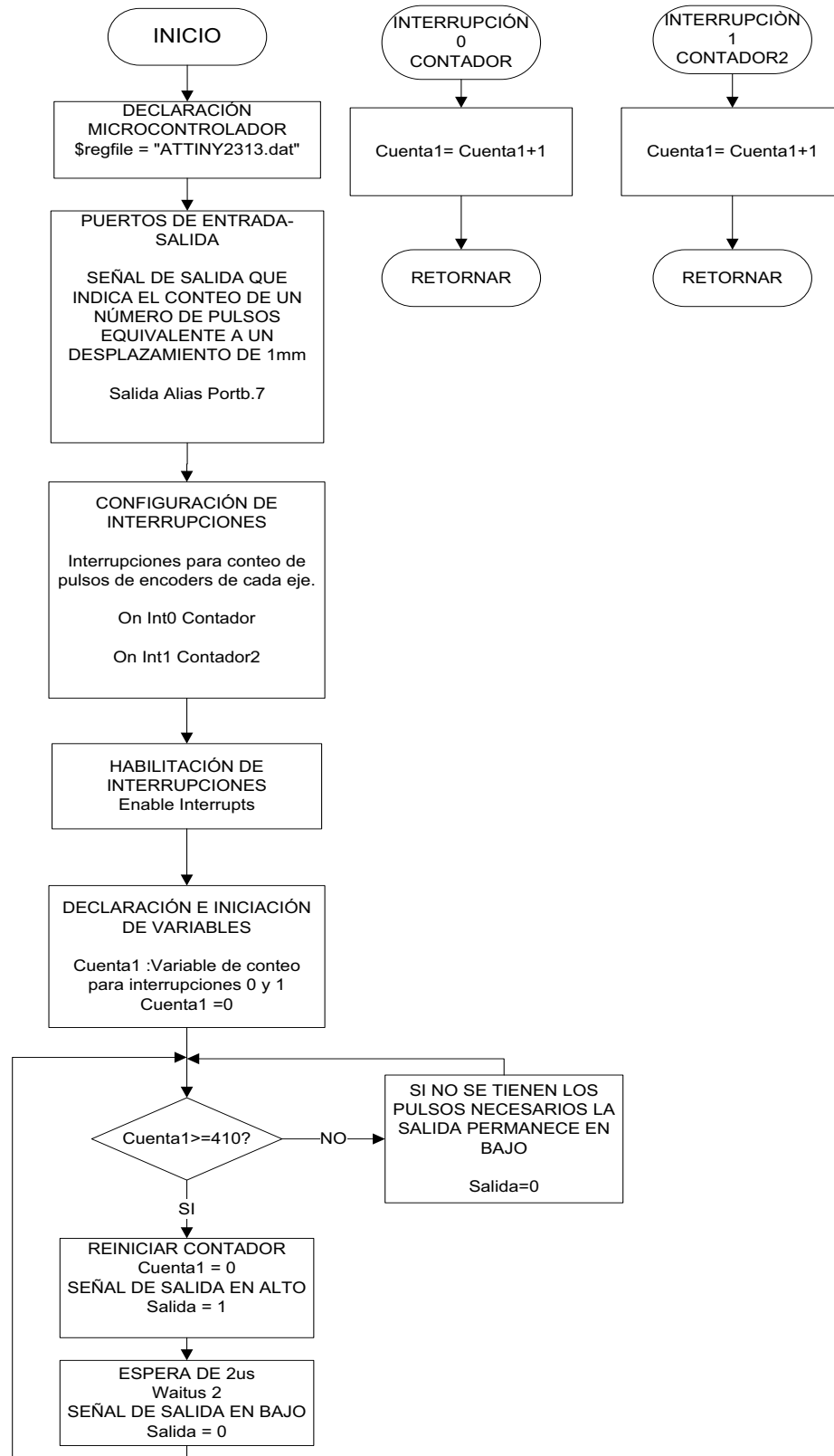


Figura 2.47 Conteo de pulsos equivalente a 1 mm de desplazamiento, uso de ATTINY2313

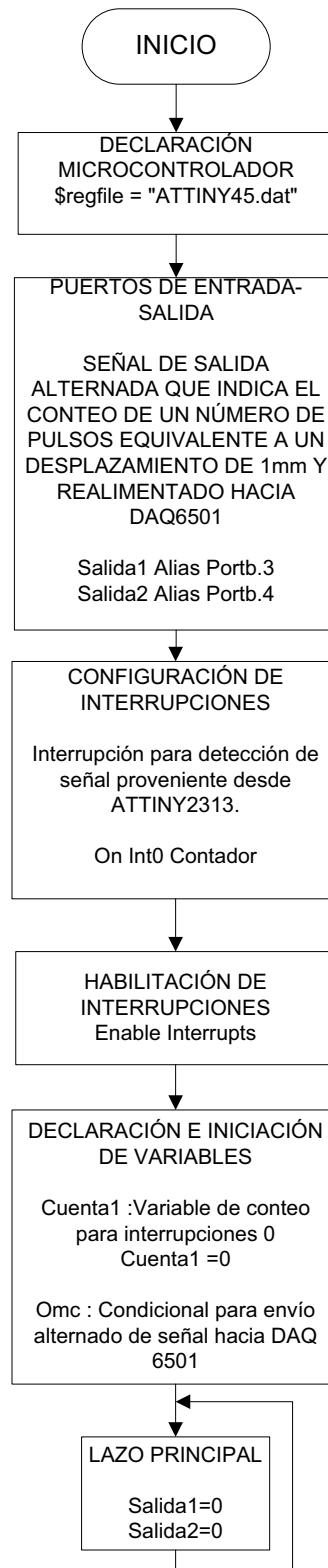


Figura 2.48 Lazo principal de ATTINY45 para envío de señal alternada.

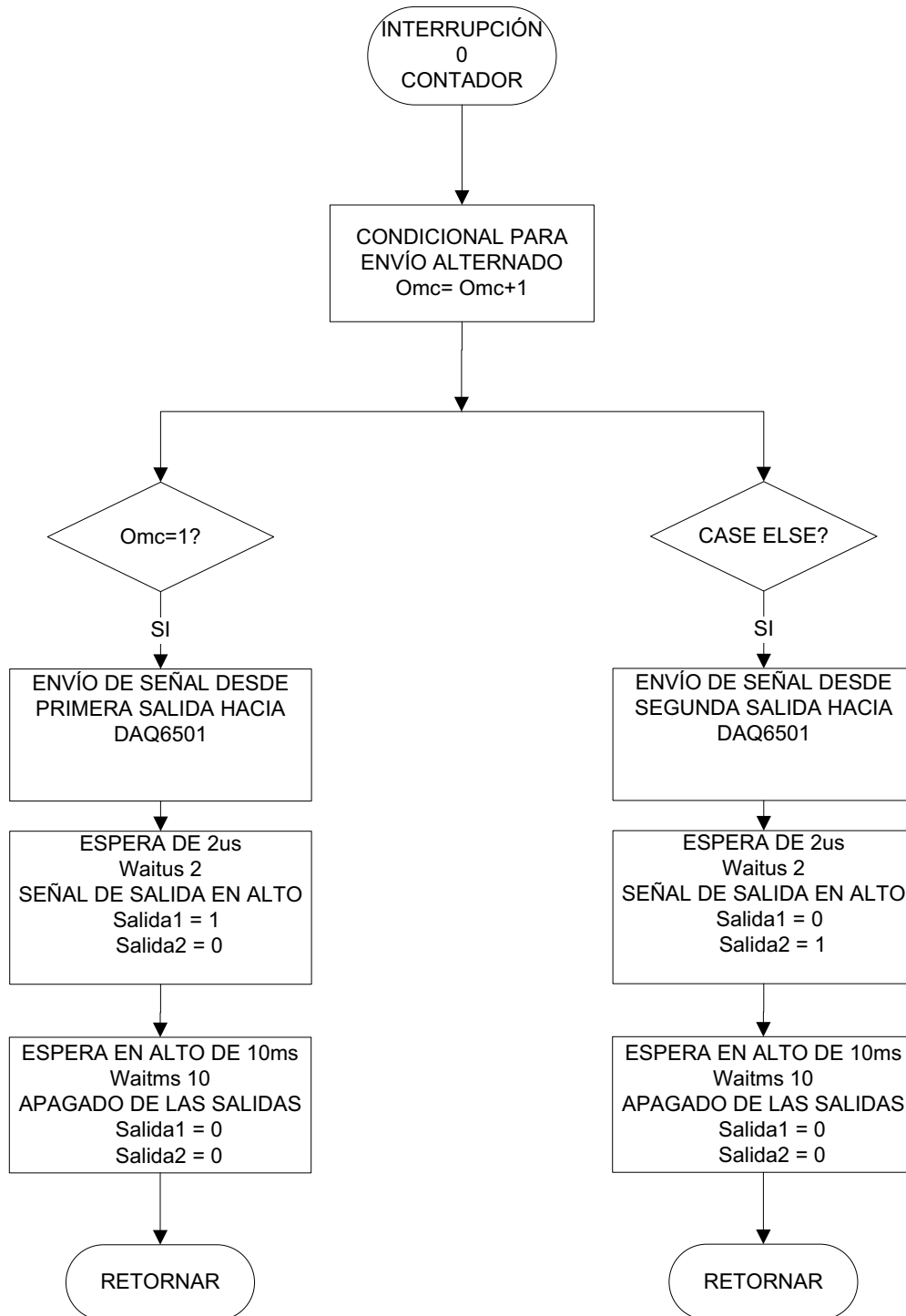


Figura 2.49 Uso de interrupción de ATTINY45 para envío de señal alternada con tiempo en alto de 10 ms hacia DAQ6501

2.4.2 ALGORITMO DE CONTROL DE MOTORES

El avance del corte de la antorcha se realiza en forma escalonada, de esta manera se tiene mayor definición en el contorno a cortar incluso en figuras circulares. El algoritmo implementado en un microcontrolador ATTINY2313 comanda los movimientos correspondientes al código enviado desde la DAQ6501, también cuenta con la detección de cuatro pulsadores que corresponden a los cuatro movimientos básicos, dichos pulsadores deben ser empleados únicamente para realizar un reposicionamiento de la antorcha en cualquier evento extraordinario, como por ejemplo un repentino fallo de la energía de la red, o realización de cortes rectos sin el uso del software y una PC.

Debe aclararse que el código generado por los pulsadores cuenta con la restricción ante el accionamiento de dos pulsadores que indiquen el avance en dos sentidos en un mismo eje, es decir se evita la contradicción de que un motor gire a la derecha y a la izquierda a la vez. Este código tiene mayor jerarquía sobre el enviado desde la DAQ6501, por tanto, mientras se realiza un corte deben evitarse pulsar cualquiera de estos botones; es por esto que como punto de seguridad estos pulsadores estarán bajo llaves.

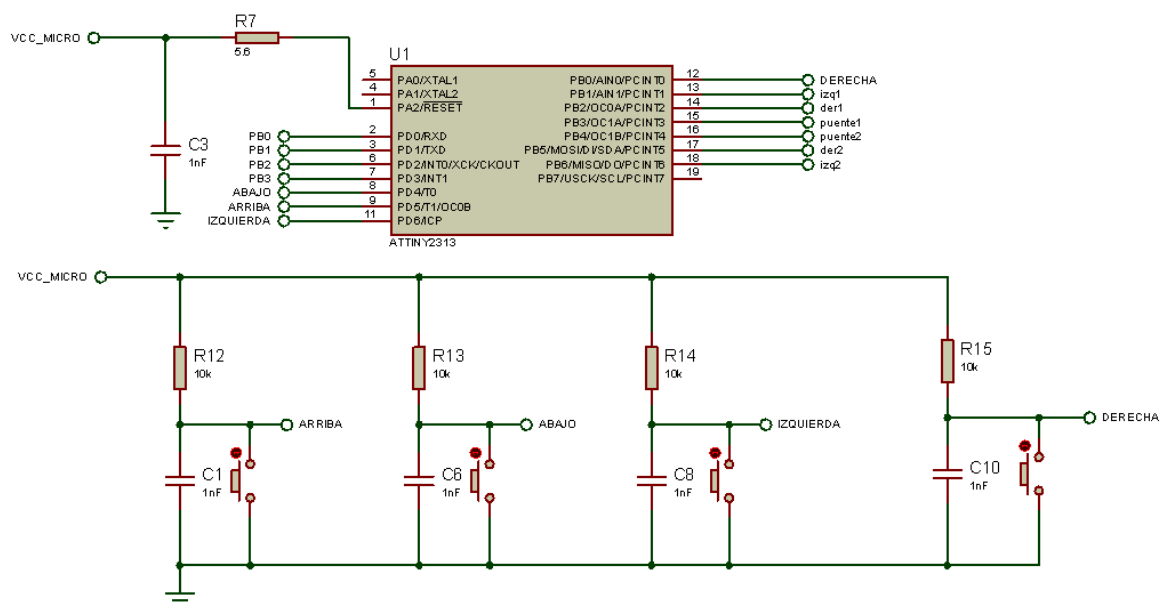


Figura 2.50 Hardware de control de motores, pulsadores para reposicionamiento manual

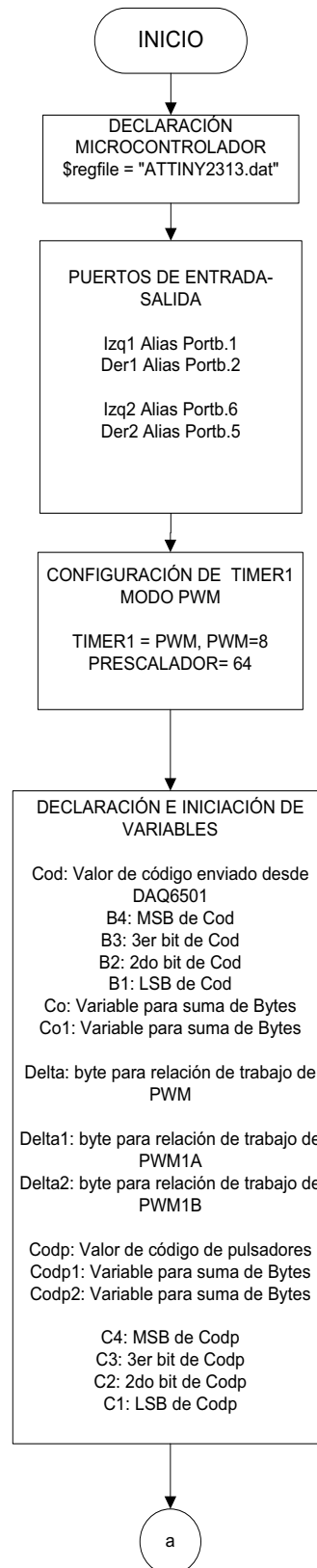


Figura 2.51 Configuración de PWM, configuración de puertos, declaración de variables

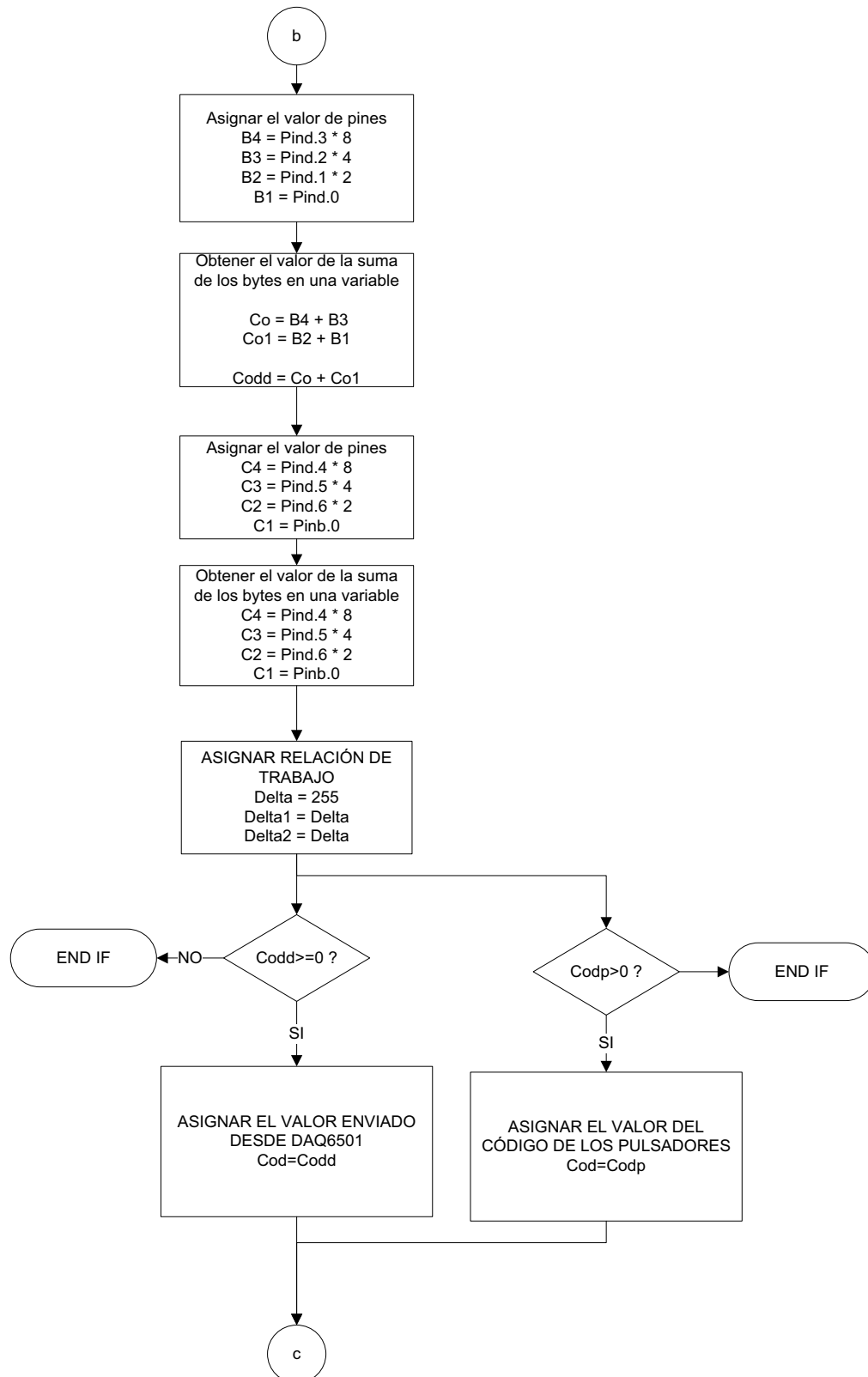


Figura 2.52 Lazo principal, lectura de código G y asignación de PWM a cada motor PARTE1

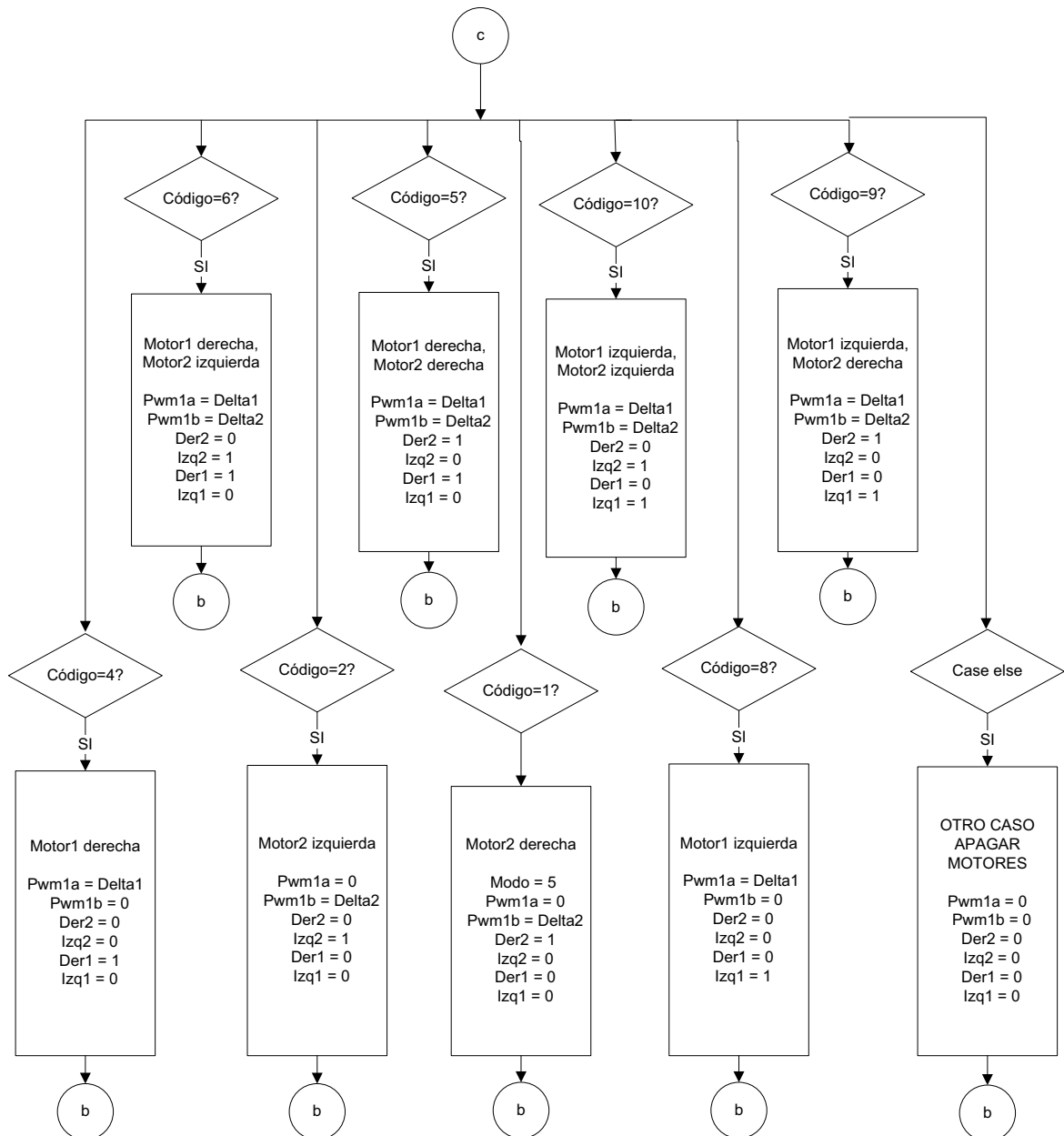


Figura 2.53 Lazo principal, lectura de código G y asignación de PWM a cada motor PARTE2

2.5 CREACIÓN DE ARCHIVO EJECUTABLE, ARCHIVO .EXE

MATLAB posee un compilador que convierte programas en archivos .exe, los cuales pueden ejecutarse sin necesidad de abrir MATLAB o funcionar en otros

ordenadores que no tengan instalado este software. Para ejecutar este compilador es necesario tener instalado SIMULINK. [9]

En la ventana de comandos se ejecuta el siguiente código.

```
>> mcc -m cuatro_guide.m cuatro_guide.fig
```

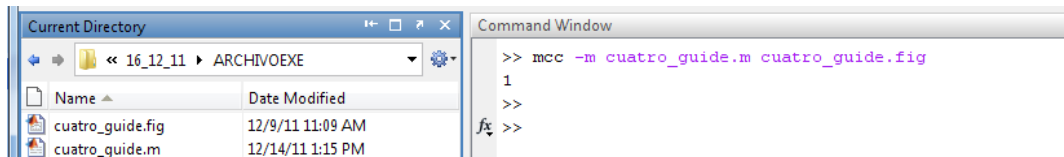


Figura 2.54 Sentencia para crear archivo ejecutable.

Donde *cuatro_guide* es el nombre de los archivos .m y .fig del programa que se requiere convertir en .exe; debe aclararse que este archivo es la ventana principal de la interfaz, basta con ejecutar éste para que las ventanas secundarias sean creadas.

El compilador crea los siguientes archivos, donde el primero es el ejecutable.

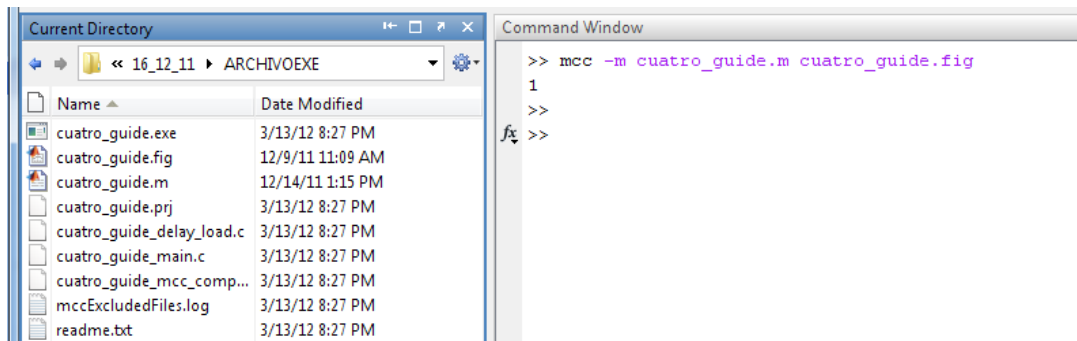


Figura 2.55 Archivos creados junto con archivo .exe

2.6 INSTALACIÓN DE ARCHIVO EJECUTABLE

Para instalar una aplicación que no posea MATLAB: [9]

1. Copiar el instalador del RunTime de un computador que cuente con MATLAB: <matlabroot>\toolbox\compiler\deploy\win32\MCRInstaller.exe



Figura 2.56 Instalador de RunTime de MATLAB

- Desempaquetar los instaladores, haciendo doble clic en MCRInstaller (ya en el computador sin MATLAB).

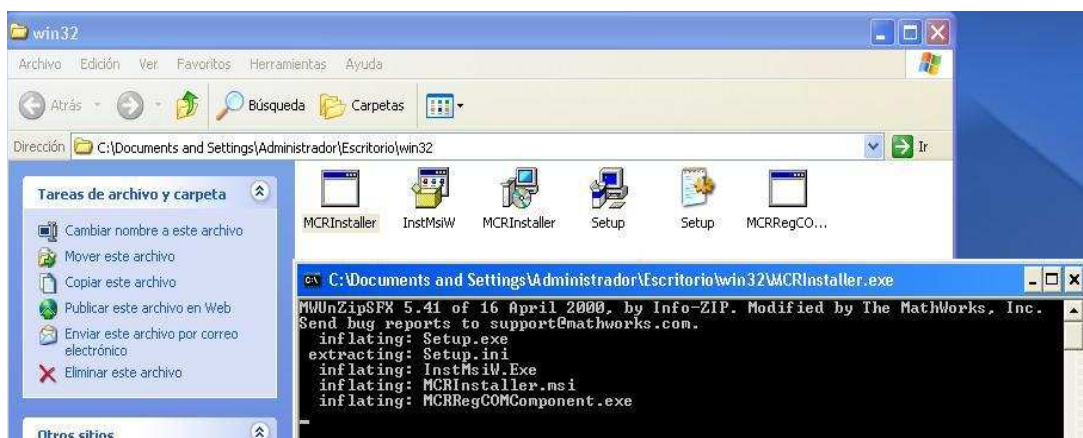
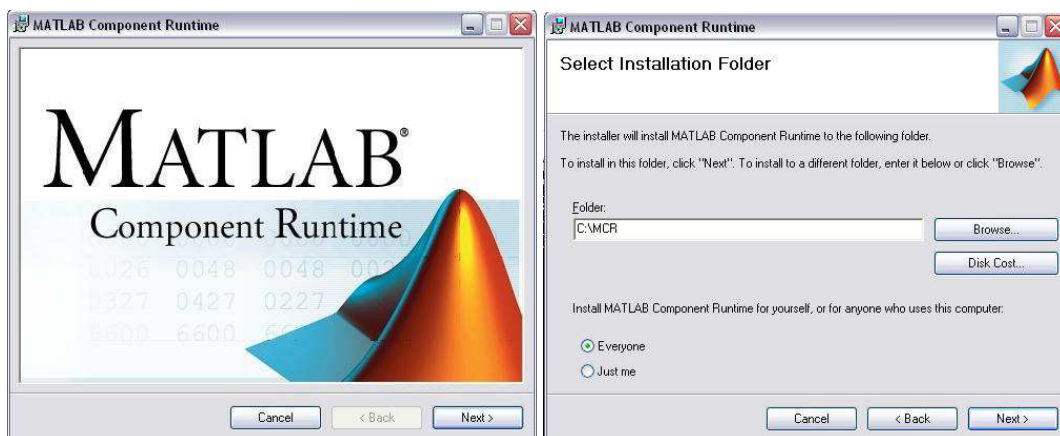


Figura 2.57 Extracción del paquete de instalación.

- En el ordenador de destino: Instalar MCRInstaller en C:\MCR (por ejemplo).



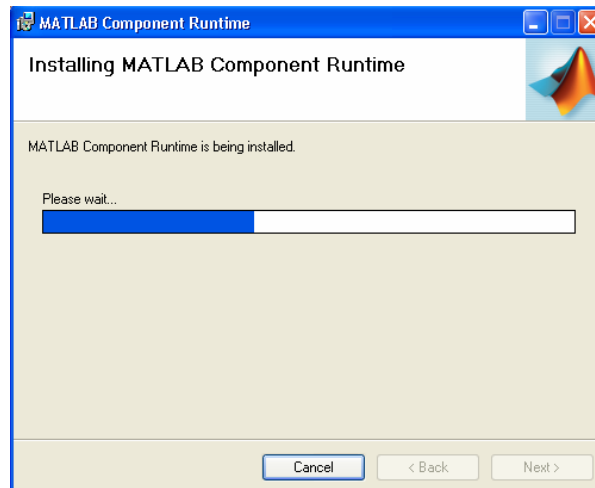


Figura 2.58 Instalación del componente RunTime en el computador de destino

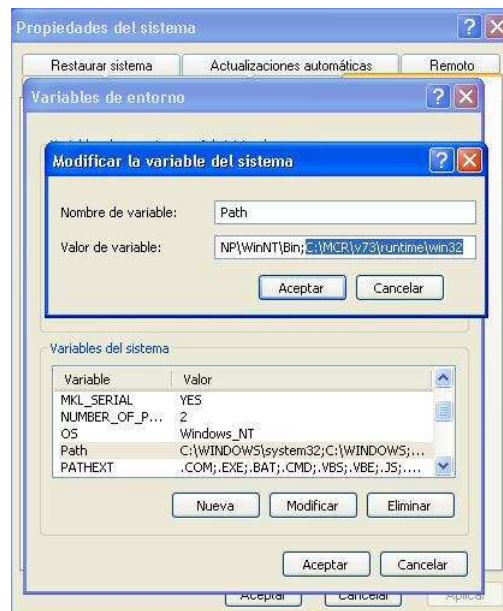
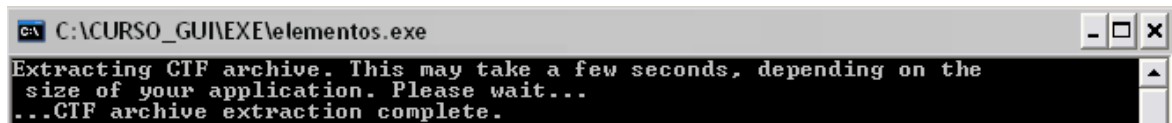


Figura 2.59 Path del Sistema Operativo

Luego de la instalación del RunTime, asegurarse de que `C:\MCR\v73\runtime\win32` esté en el PATH. Para esto, hacer click derecho en el ícono *Mi PC*, seleccionar *propiedades, opciones avanzadas, variables de entorno, variables de sistema*. [9]

En caso de no encontrarse, debe agregarse al path: `C:\MCR\v73\runtime\win32`

Copiar elementos.exe y elementos.ctf al directorio de la aplicación del CPU que no tiene MATLAB. Al abrir el archivo .exe, aparecerá la pantalla del DOS y luego el programa: [9]

A screenshot of a DOS command window. The title bar reads "C:\ACURSO_GUI\EXE\elementos.exe". The window contains the following text: "Extracting CTF archive. This may take a few seconds, depending on the size of your application. Please wait... ..CTF archive extraction complete." The window has standard DOS window controls (minimize, maximize, close) in the top right corner.

```
C:\ACURSO_GUI\EXE\elementos.exe
Extracting CTF archive. This may take a few seconds, depending on the
size of your application. Please wait...
...CTF archive extraction complete.
```

Figura 2.60 Extracción de los archivos CTF

Es importante indicar que en el directorio, en cual se encuentran los elementos .exe y .ftc, deben añadirse todos los archivos que se emplearon para la creación del archivo .fig, tales como archivos de formato gráfico que conforman el mismo, caso contrario la interfaz creada no se ejecutará como previamente se la haya evaluado.

CAPÍTULO 3

DISEÑO DEL SISTEMA Y MONTAJE

El objetivo principal del sistema mecánico es proporcionar un movimiento longitudinal y transversal a la antorcha para una máquina oxicorte, además de dar soporte a los elementos que actúan en el proceso.

3.1 DESCRIPCIÓN GENERAL DEL PROCESO

3.1.1 MECANISMO TORNILLO TUERCA

Para transformar el movimiento circular de un tornillo en movimiento lineal se emplea el mecanismo tornillo-tuerca, conocido también como husillo-tuerca, el cual es un mecanismo compuesto por una tuerca alojada en un eje roscado (tornillo), si el tornillo gira y se mantiene fija la orientación de la tuerca, la tuerca avanza con movimiento rectilíneo. [11]

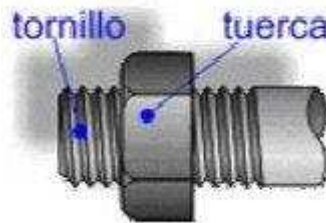


Figura 3.1 Transformación del movimiento circular a lineal

La velocidad de avance depende de dos factores:

- La velocidad de giro del elemento motriz.
- El paso de la rosca, es decir, la distancia que existe entre dos crestas de la rosca del tornillo. Cuando mayor sea el paso, mayor será la velocidad de avance.

3.1.2 TORNILLOS SIN FIN

Para conseguir el movimiento del soplete de la máquina oxicorte en cualquier punto del plano, usan dos tornillos sin fin de $\frac{3}{4}$ ", con piñones acoplados a sus extremos, además con tuercas que realizan el movimiento lineal como se explicó en el literal 3.1.1

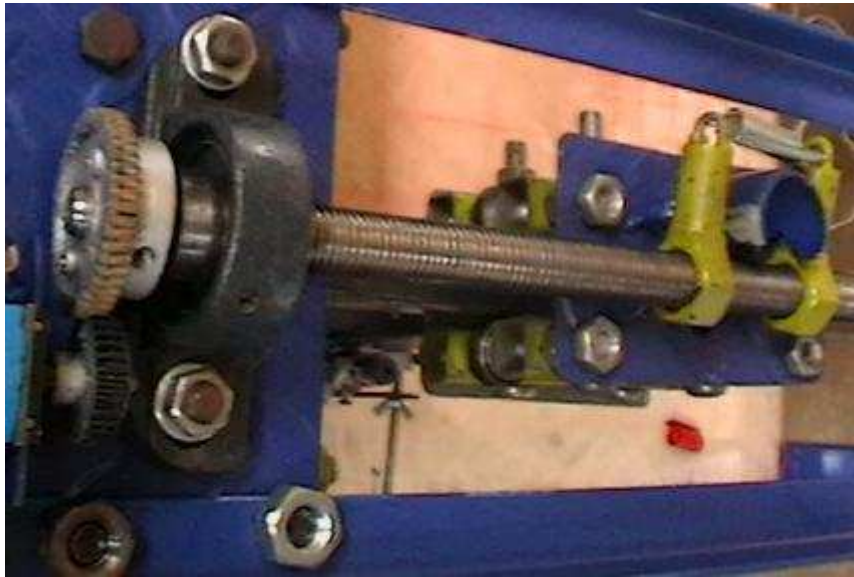


Figura 3.2 Acoplamiento de piñones en los extremos del tornillo sin fin

Como soporte de los tornillos sin fin se emplean chumaceras de $\frac{1}{2}$ " con el propósito de permitir un movimiento en su propio eje, cabe notar que el diámetro de los tornillos sin fin es $\frac{3}{4}$ " y el diámetro de las chumaceras es $\frac{1}{2}$ ", por lo cual es necesario devastar los extremos del tornillo hasta conseguir un diámetro de $\frac{1}{2}$ ". Se realiza este procedimiento con el objetivo de mejorar la sujeción de los tornillos en las chumaceras¹.

¹ Pieza con una muesca que permite descansar y girar un eje de maquinaria.



Figura 3.3 Chumacera que soporta tornillo sin fin

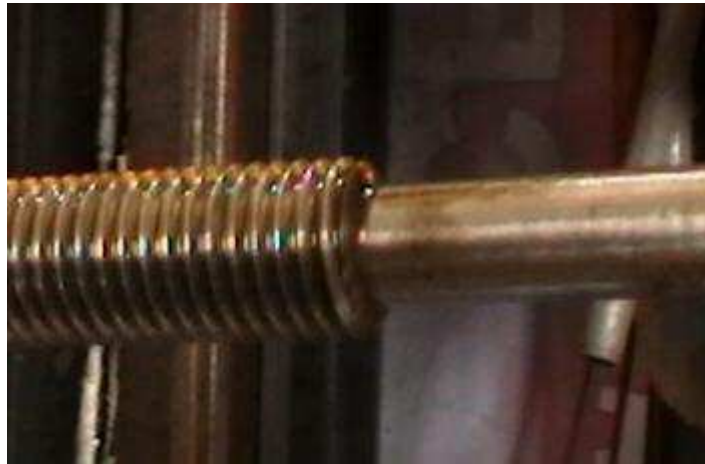


Figura 3.4 Tornillo devastado

3.1.3 ACOPLAMIENTO DE PIÑONES

Para transmitir el movimiento del motor hacia el tornillo sin fin se acopla un piñón al eje del motor y uno al tornillo sin fin, la relación de piñones es 12/30 por lo cual el motor debe dar 2.5 vueltas, para que el tornillo gire una vuelta.

Al cambiar la relación entre piñones se consigue mayor o menor velocidad en la rotación del tornillo sin fin.

El grosor del material tiene una relación inversa con la velocidad de avance del soplete, para los requerimientos de este caso sólo es necesaria una velocidad.

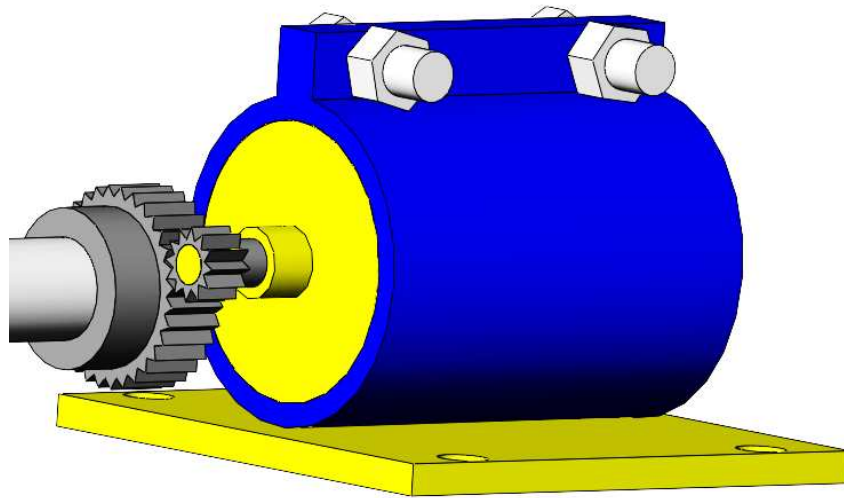


Figura 3.5 Acoplamiento de piñones entre motor y tornillo sin fin

A continuación se presenta la relación que existe entre el espesor de una plancha metálica y la velocidad de corte requerida para la misma.

Espesor (mm)	Diámetro boquilla (mm)	Presiones O ₂ en el soplete (bar)	Velocidad de corte (m/ft)
5	0,6	1,5	20
8	0,8	1,5	17
10	1	1,5	15
15	1	2	12
20	1	2,5	11,5
25	1,5	2,5	10
30	1,5	2,5	9,5
40	2	3	8,5
50	2	3,5	7
75	2	4	5,5
100	2,5	4	4,5

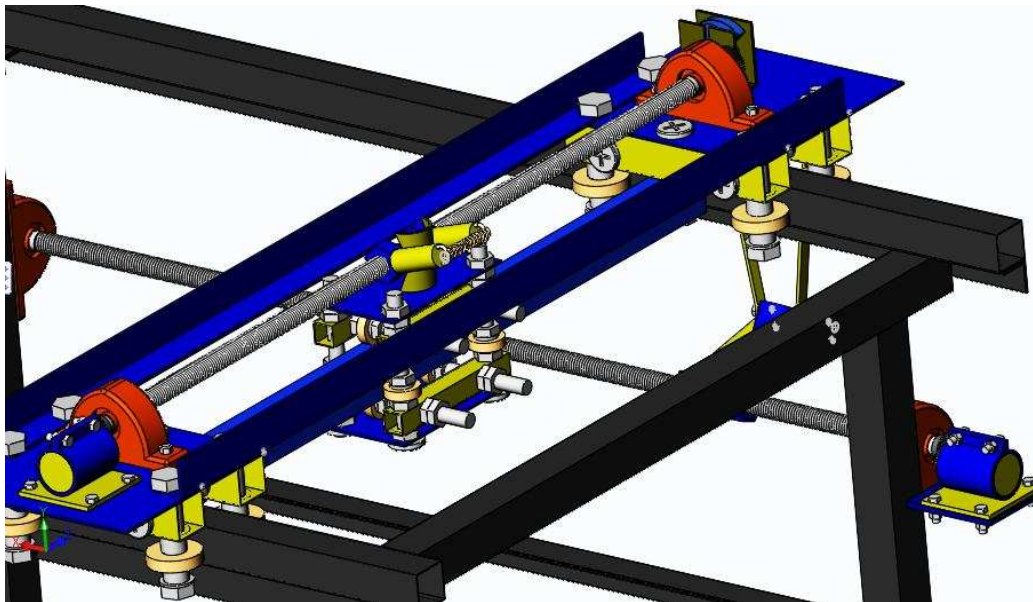
Tabla 1.- Valores recomendados para Uxcorte

Figura 3.6 Espesor de la plancha metálica vs. velocidad de corte

3.1.4 EJES DE MOVIMIENTO

El primer movimiento se realiza en el eje longitudinal, usando el mecanismo tornillo tuerca se consigue movimiento lineal, el cual es transmitido a un sistema compuesto por dos carros.

El segundo movimiento se realiza en el eje transversal, al igual que en el anterior caso, un motor se encarga de mover un tornillo sin fin, y basados en el mecanismo tornillo tuerca, se transmite movimiento lineal a un carro que se desplaza en el eje transversal.



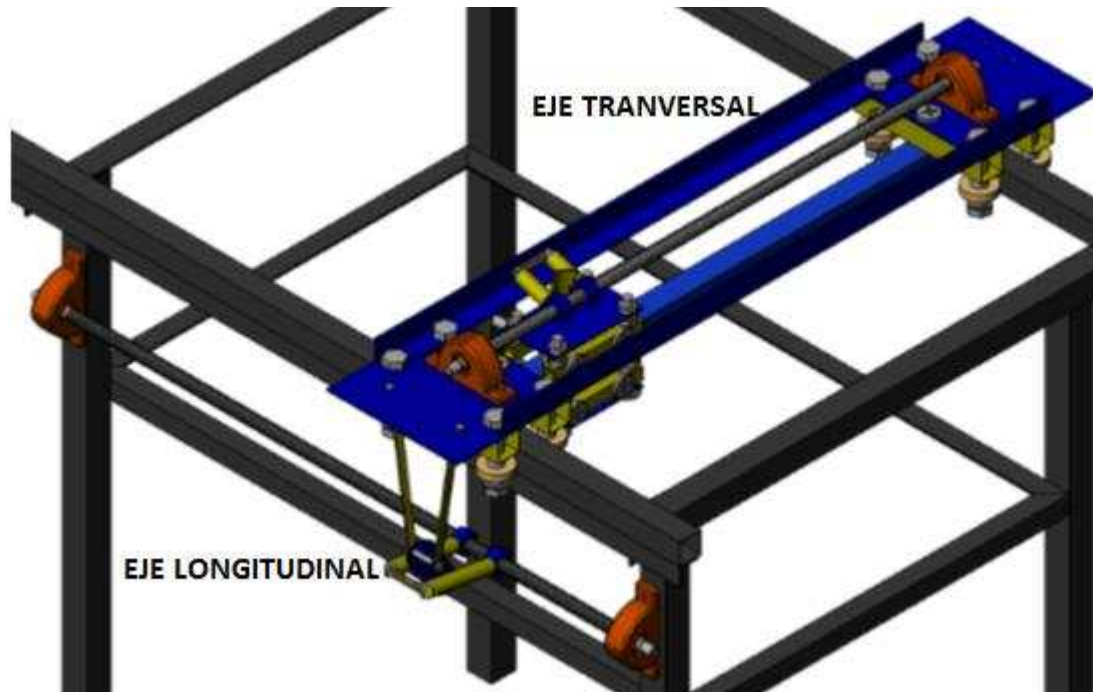


Figura 3.7 Sistema de desplazamiento en eje longitudinal y transversal

La combinación de los movimientos da lugar a un recorrido completo en el plano, permitiendo así realizar el corte de cualquier figura.

3.2 DISEÑO DEL SISTEMA MECÁNICO

Como primer punto se diseña e implementa una mesa para soportar una plancha de metal, dos motores (12Vdc/5A /12kg), dos encoders² y dos pernos sin fin.

El presente proyecto realizara cortes de figuras que se encuentren dentro de un formato A1 (841 mm x 594 mm), lo cual indica que el soplete de oxicorte debe moverse en este rango, más para que esto sea posible, la mesa a implementar posee un tamaño mayor al formato A1, lo que evita el roce del soplete con la mesa de trabajo. Para obtener el sistema final se implementan cuatro prototipos mecánicos, buscando mejorar problemas de precisión y dureza.

² Transductor rotativo que transforma un movimiento angular en una serie de impulsos digitales.

3.2.1 PRIMER PROTOTIPO MECÁNICO

El primer diseño implementado consta de dos tornillos sin fin paralelos, unidos por una banda de caucho, y un tornillo sin fin que atraviesa los tornillos paralelos en un ángulo de 90°.



Figura 3.8 Primer prototipo mecánico

Ayudados por poleas colocadas en los extremos de dos tornillos sin fin, se proporciona movimiento en el eje longitudinal, y el tornillo atravesado, proporciona movimiento en el eje transversal. Los dos tornillos paralelos se encargan de mover en el eje longitudinal al tornillo transversal, lo que da lugar a un movimiento en dos dimensiones. Para dar movimiento a los tornillos se usan dos motores con un piñón acoplado en su extremo, y a su vez se acopla un piñón en el extremo de cada tornillo sin fin.



Figura 3.9 Acoplamiento de piñones en extremos de tornillos sin fin (primer prototipo mecánico)



Figura 3.10 Ejes longitudinal y transversal (primer prototipo mecánico)

Inconvenientes:

- Se pierde transmisión de movimiento entre los tornillos paralelos puesto que la banda resbala en las poleas. Para evitar este inconveniente se presiona la banda, más se produce curvatura en el tornillo, lo cual endurece el sistema.



Figura 3.11 Transmisión de movimiento entre tornillos sin fin del eje longitudinal (primer prototipo mecánico)

- Se busca tener una relación perpendicular entre el eje transversal y longitudinal, pero el presente prototipo mecánico no permite obtener dicho propósito.

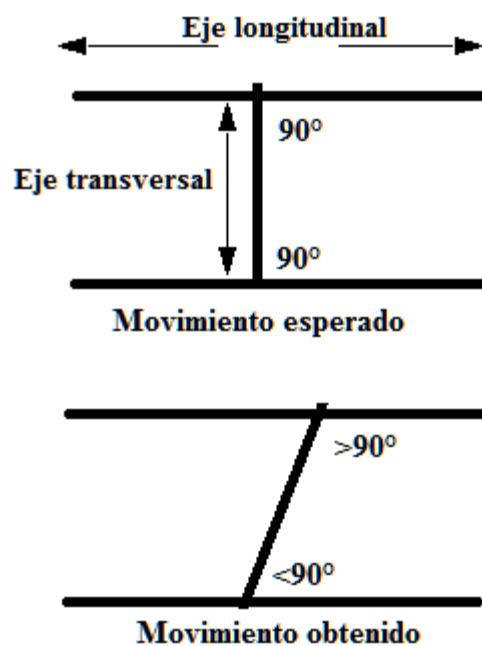


Figura 3.12 Falta de perpendicularidad entre ejes (primer prototipo mecánico)

Los inconvenientes mencionados dan lugar a un pésimo funcionamiento del prototipo mecánico, por lo cual no es posible controlar el sistema.

3.2.2 SEGUNDO PROTOTIPO MECÁNICO

Se reemplazan los dos tornillos sin fin paralelos presentes en el primer prototipo, por un solo tornillo sin fin con el objetivo de mejorar el avance perpendicular del eje transversal con respecto al eje longitudinal.



Figura 3.13 Eje longitudinal y transversal (segundo prototipo mecánico)

Se incorporan rodamientos en los extremos del eje transversal, para permitir su movimiento.



Figura 3.14 Rodamiento que soporta al eje transversal (segundo prototipo mecánico)

Al igual que en el primer prototipo, se incorporan piñones tanto a los motores como a los extremos de los tornillos sin fin, con el objetivo de transmitir movimiento.

Para mejorar el movimiento longitudinal, se incorporan guías que buscan mantener la relación perpendicular entre los ejes.

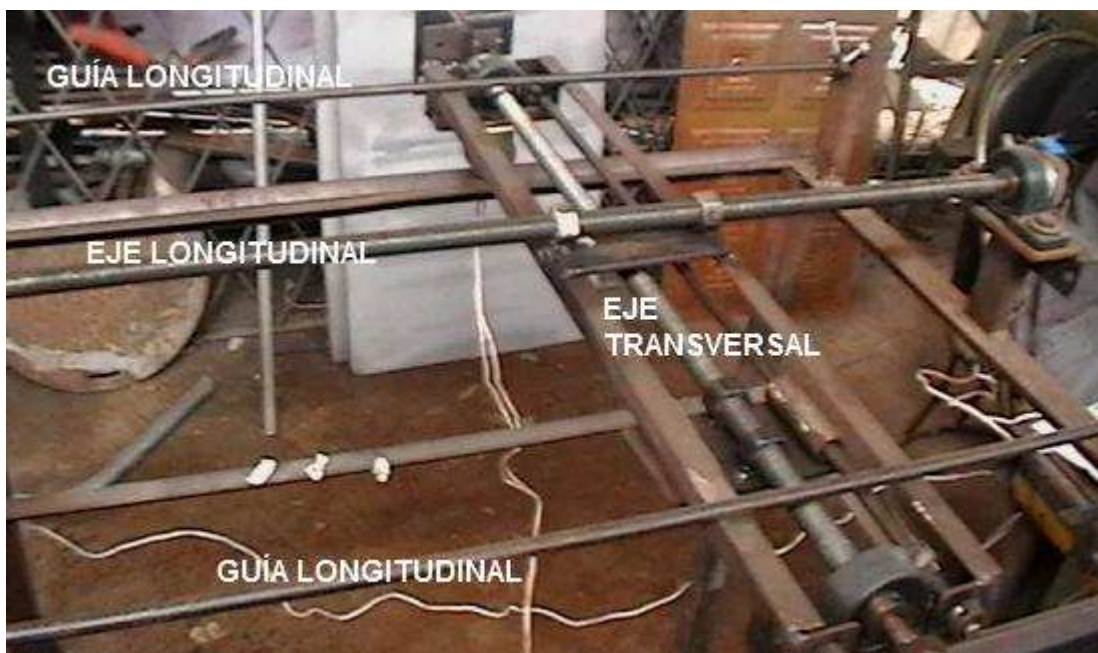


Figura 3.15 Guías para eje longitudinal (segundo prototipo mecánico)

Inconvenientes:

- Los rodamientos incorporados en el eje transversal, no poseen la estabilidad requerida, debido a su falta de sujeción con el riel angular.



Figura 3.16 Falta de estabilidad en rodamientos eje transversal (segundo prototipo mecánico)

- A pesar de las guías usadas para mantener perpendiculares los ejes, no se consigue el objetivo, y de la misma manera que el primer prototipo, se tiene una inclinación no deseada.

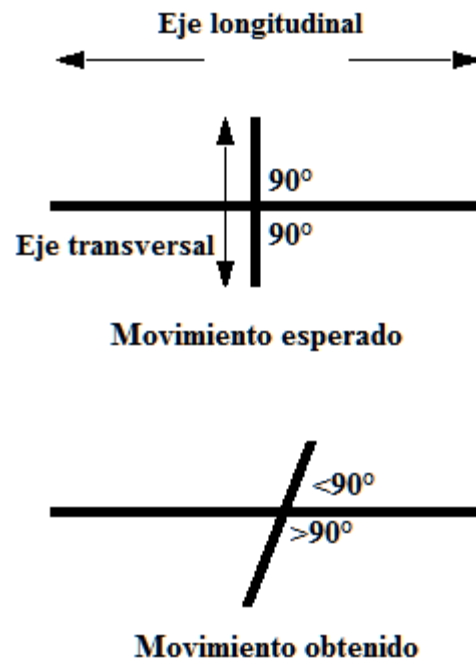


Figura 3.17 Falta de perpendicularidad entre ejes (segundo prototipo mecánico)

3.2.3 TERCER PROTOTIPO MECÁNICO

Se incorporan dos carros en los extremos del eje transversal, los cuales poseen tubería redonda como guía. Además, se realiza un carro para dar movimiento en el eje transversal.



Figura 3.18 Carros que permiten movimiento en eje longitudinal y transversal
(tercer prototipo mecánico)

Se incorpora un tornillo sin fin con dos tuercas que arrastran un carro transversal y un tornillo cuyas tuercas arrastran a dos carros longitudinales, dando lugar al movimiento deseado en dos dimensiones.

El inconveniente con este sistema radica en la falta de sujeción de los carros longitudinales y transversales con sus respectivas guías, lo cual no permite transmitir adecuadamente el movimiento lineal proporcionado por las tuercas presentes en los tornillos sin fin.

La falta de sujeción se debe a que el área de contacto entre las guías y los rodamientos de los carros es mínima, pues el rodamiento es cilíndrico y la guía circular.



Figura 3.19 Falta de sujeción entre guías y rodamientos (tercer prototipo mecánico)

3.2.4 CUARTO PROTOTIPO MECÁNICO

Como se observa en los anteriores diseños, la precisión y suavidad del movimiento lineal depende en gran medida del avance de la tuerca en el tornillo sin fin, este hecho implica que se requiera de mucha precisión en la colocación de tornillos, tuercas y rieles, mas tomando todas las precauciones necesarias para conseguir este fin, resulta muy complejo conseguir tal objetivo, a partir de este hecho nace la idea de independizar en la medida de lo posible el movimiento de los ejes con respecto a los tornillos sin fin, para evitar que los mismos realicen la función de proporcionar movimiento y ser guías a la vez. Basándose en esta idea se detalla el diseño final del sistema.

3.2.4.1 Fabricación De Mesa

Se realizan dos marcos rectangulares, uno para el soporte del sistema encargado del control y otro para el soporte de la plancha de metal a ser cortada. (Todas las medidas se encuentran expresadas en mm).

3.2.4.1.1 *SOPORTE PARA LA PLANCHA DE METAL:*

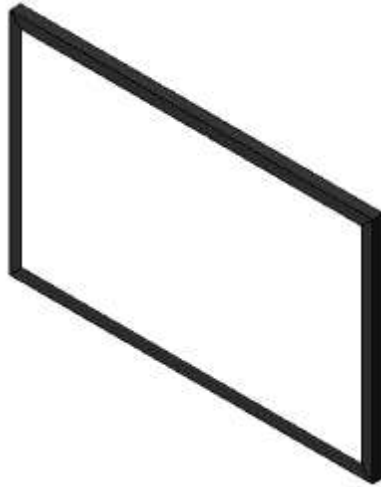


Figura 3.20 Soporte para plancha de metal. Ver ANEXO 3-1

3.2.4.1.2 *SOPORTE PARA SISTEMA DE CONTROL:*

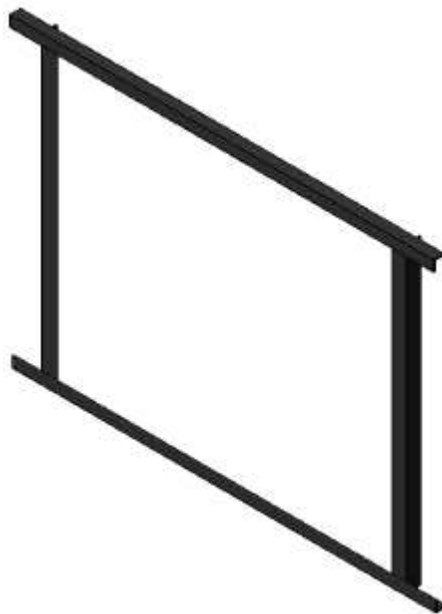


Figura 3.21 Soporte para sistema de control. Ver ANEXO 3-2

Para proseguir se incorporan cuatro soportes en cada esquina de los marcos realizados, y se añaden dos tubos cuadrados que servirán como guías para el desplazamiento del eje transversal.



Figura 3.22 Mesa armada. Ver Anexo 3-3

3.2.4.2 Incorporación Del Eje Longitudinal

Para conseguir que la antorcha de la máquina oxicorte se mueva en sentido longitudinal se realizan dos carros, los cuales se desplazarán por las rieles de tubo cuadrado incorporadas en la mesa.

3.2.4.2.1 *CONSTRUCCIÓN DE CARROS LONGITUDINALES.*

- Se realiza un carro que soporta un motor dc y una chumacera.

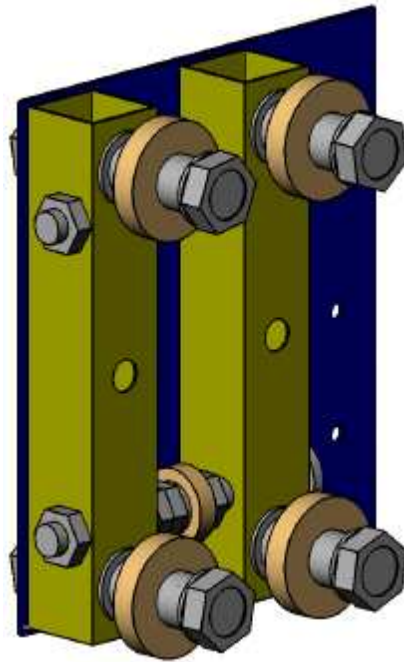


Figura 3.23 Carro longitudinal 1. Ver ANEXO 3-4.

- Se realiza un carro que soporta un motor dc y un encoder.

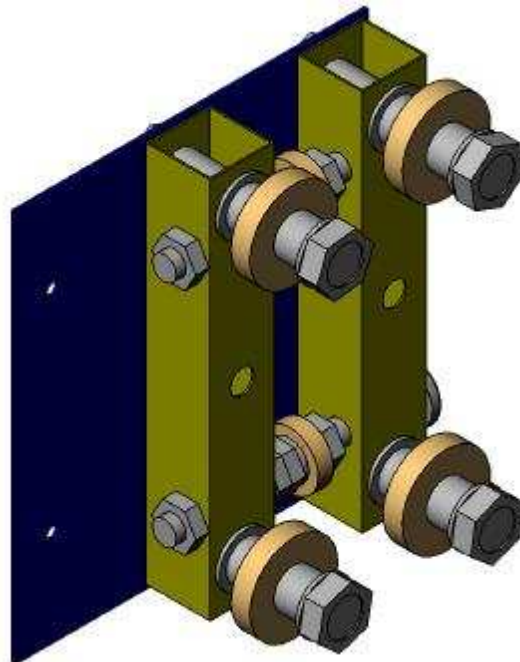


Figura 3.24 Carro longitudinal 2. Ver ANEXO 3-5

Los soportes realizados se unen mediante dos ángulos de hierro paralelos como se indica a continuación.

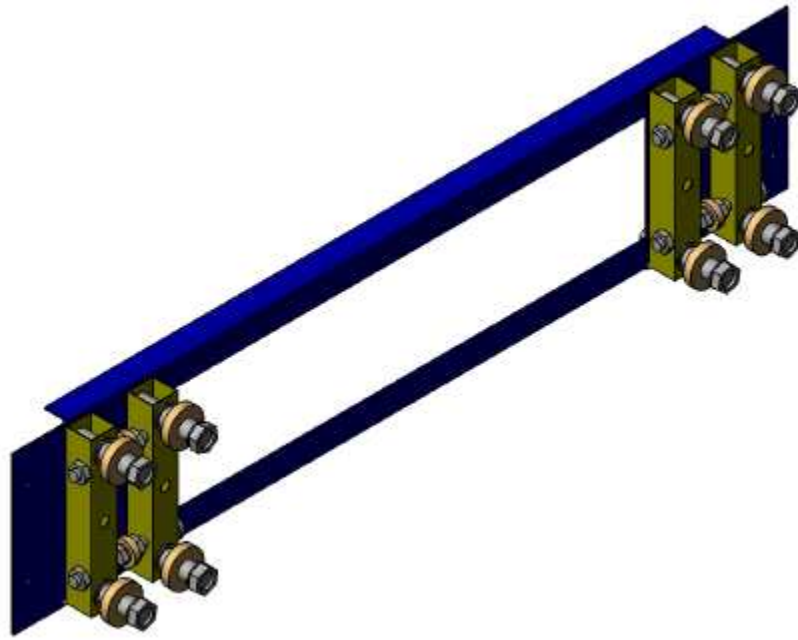


Figura 3.25 Sistema para movimiento longitudinal. Ver ANEXO 3-6

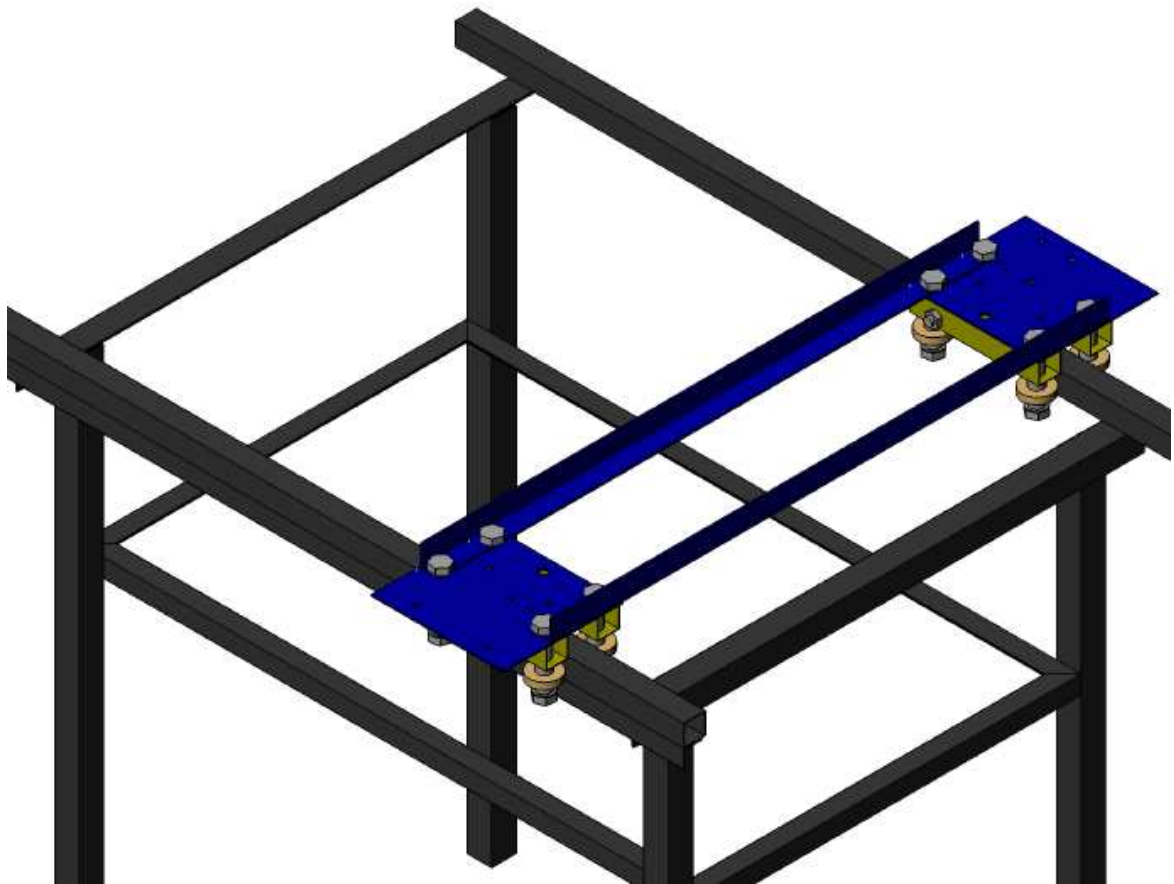


Figura 3.26 Movimiento en eje longitudinal

3.2.4.3 Incorporación Del Eje Transversal

Se incorpora un tubo cuadrado al mecanismo que realiza movimiento en el eje longitudinal, el cual servirá como guía para el movimiento transversal.



Figura 3.27 Incorporación del eje transversal

Como siguiente paso se incorpora un carro que proporciona movimiento en el eje transversal y además, da soporte a la antorcha de la máquina oxicorte.

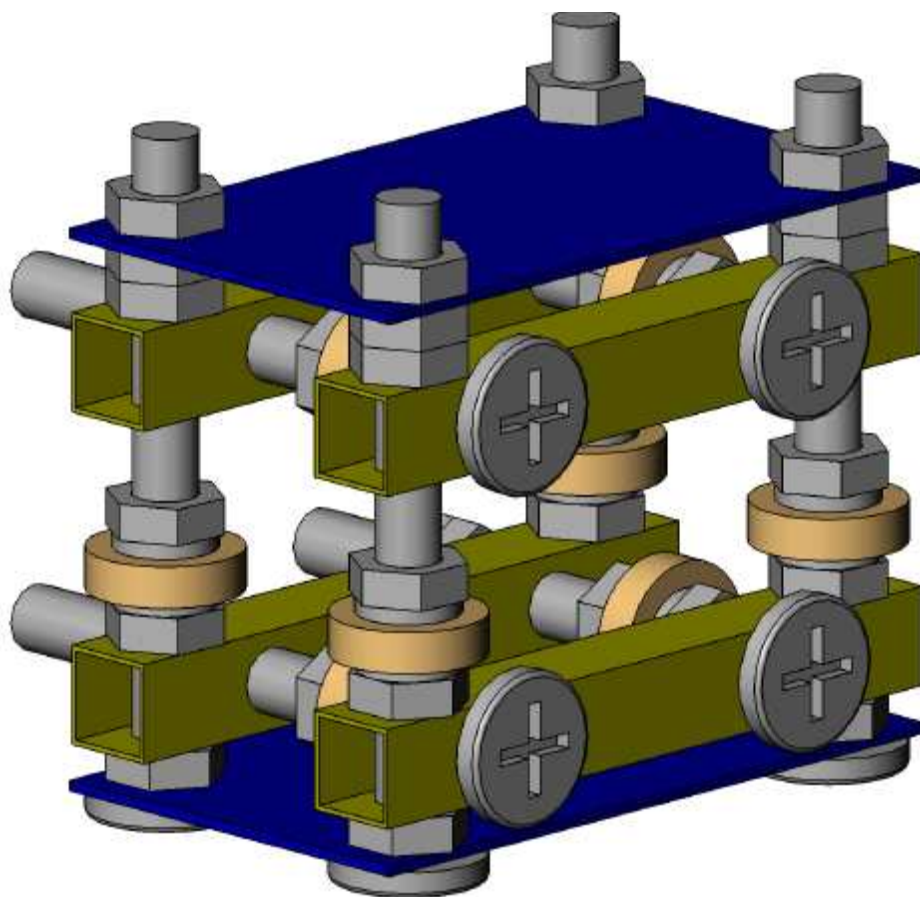


Figura 3.28 Carro para movimiento en eje transversal. Ver ANEXO 3-7

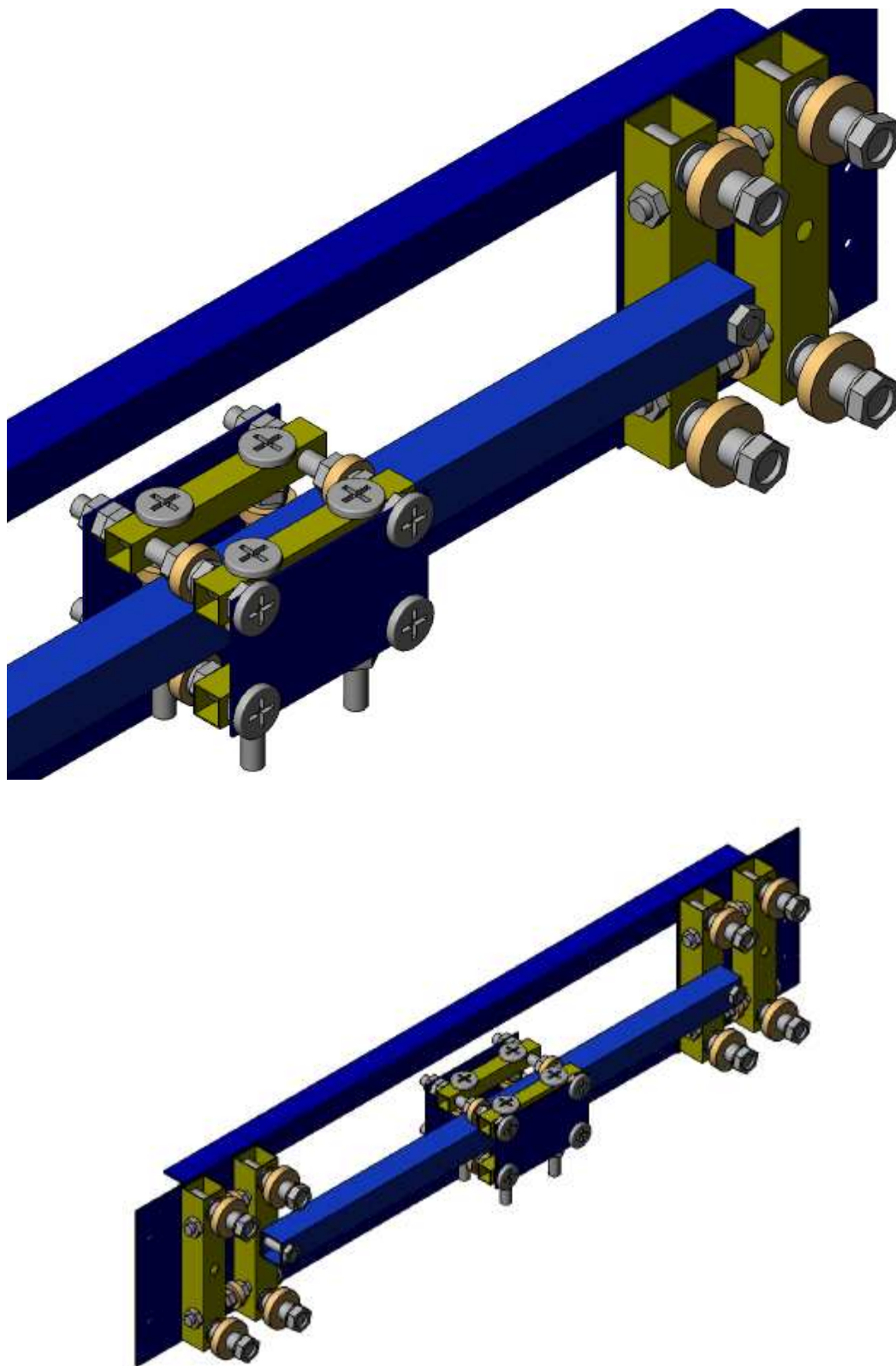


Figura 3.29 Incorporación de carro en guía transversal

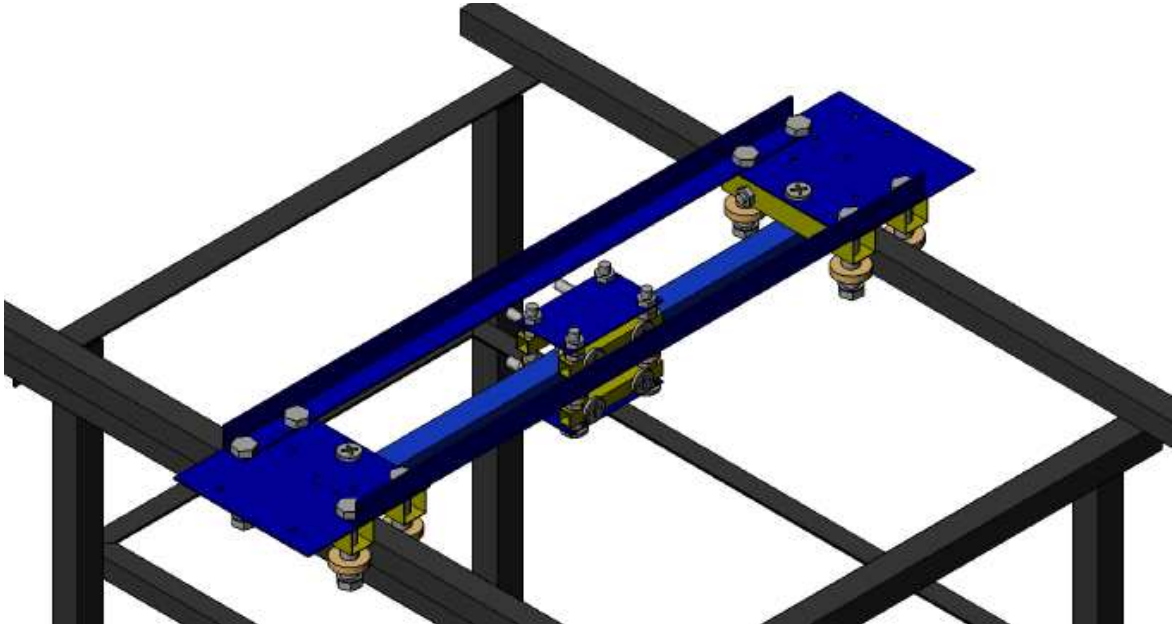


Figura 3.30 Mesa con mecanismo para movimiento longitudinal y transversal

3.2.5 DESPLAZAMIENTO EN EJES LONGITUDINAL Y TRANSVERSAL

Una vez incorporado el mecanismo necesario para dar lugar al movimiento en dos dimensiones, se procede a incluir el sistema arrastre, encargado de automatizar el desplazamiento del eje longitudinal y transversal. Para lo cual se usan dos tornillos sin fin y cuatro chumaceras, como se muestra a continuación:

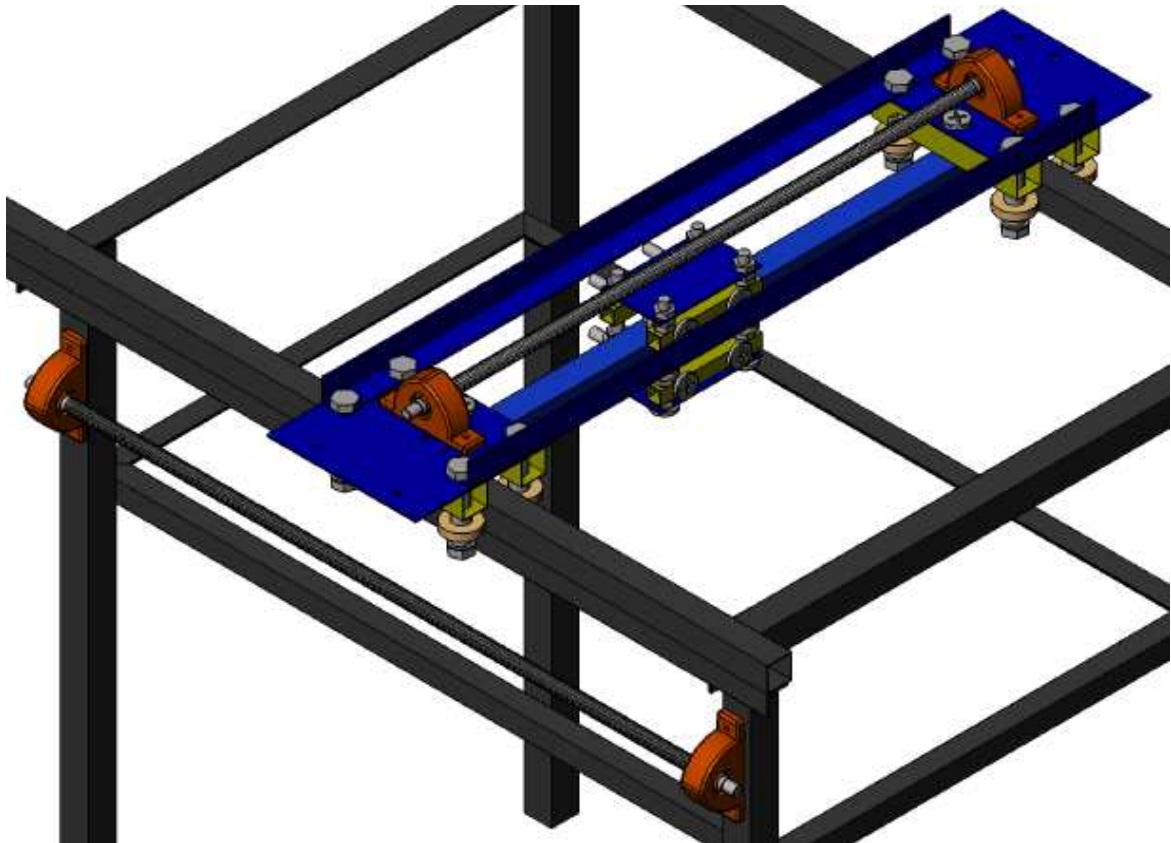


Figura 3.31 Sistema de arrastre para movimiento en dos dimensiones

Para automatizar el eje longitudinal, se acoplan dos chumaceras en los extremos de dos de los cuatro soportes que posee el sistema, las mismas que sostienen un tornillo sin fin.

En el caso del eje transversal se colocan dos chumaceras sobre cada carro longitudinal, las mismas que soportan un tornillo sin fin.

Para acoplar el movimiento lineal de una tuerca³ a los carros que proporcionan movimiento longitudinal y transversal, se busca eliminar dos problemas, el primero es el juego que existe entre la tuerca y el tornillo sin fin, y el segundo es la curvatura del tornillo sin fin (por fabricación).

Al no tener un tornillo sin fin del todo recto, no es posible acoplar directamente una tuerca a un carro longitudinal o transversal, puesto que no transmitiría sólo un

³ Mecanismo tornillo tuerca , refiérase al literal 3.1.1

movimiento lineal, sino también movimientos causados por su defecto de curvatura.

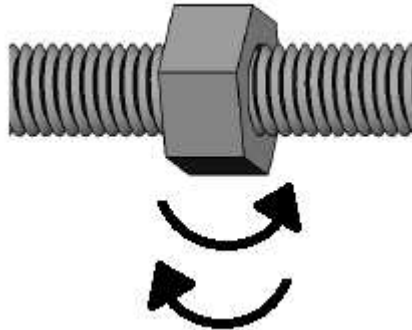


Figura 3.32 Juego de tuerca en tornillo



Figura 3.33 Curvatura del tornillo sin fin

Para eliminar los dos problemas mencionados, se usan dos tuercas con un pedazo de varilla soldada en una de sus caras laterales, unidas por un resorte y separadas por un tubo de metal, el cual se encuentra acoplado a un carro longitudinal o transversal, el resorte presiona las varillas contra el tubo de metal, consiguiendo así eliminar en la medida de lo posible el juego que existe entre el tornillo y la tuerca, además el presente sistema evita transmitir movimientos no lineales causados por la curvatura del tornillo, puesto que las varillas forman un pivote al ser presionadas contra el tubo central.

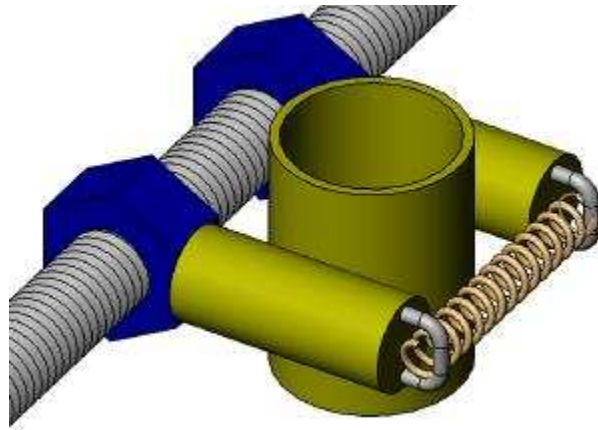


Figura 3.34 Mecanismo tornillo tuerca movimiento longitudinal. Ver ANEXO 3-8

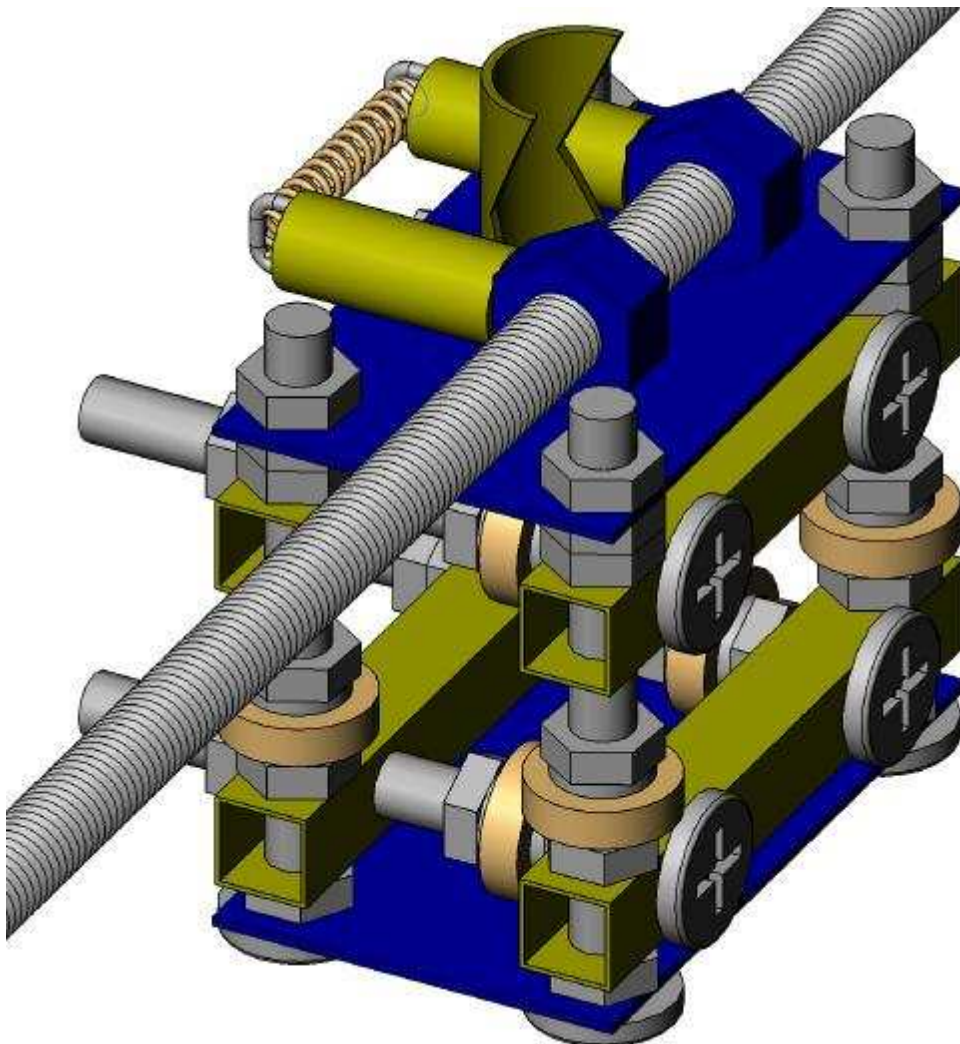


Figura 3.35 Mecanismo tornillo tuerca movimiento transversal. Ver ANEXO 3-9

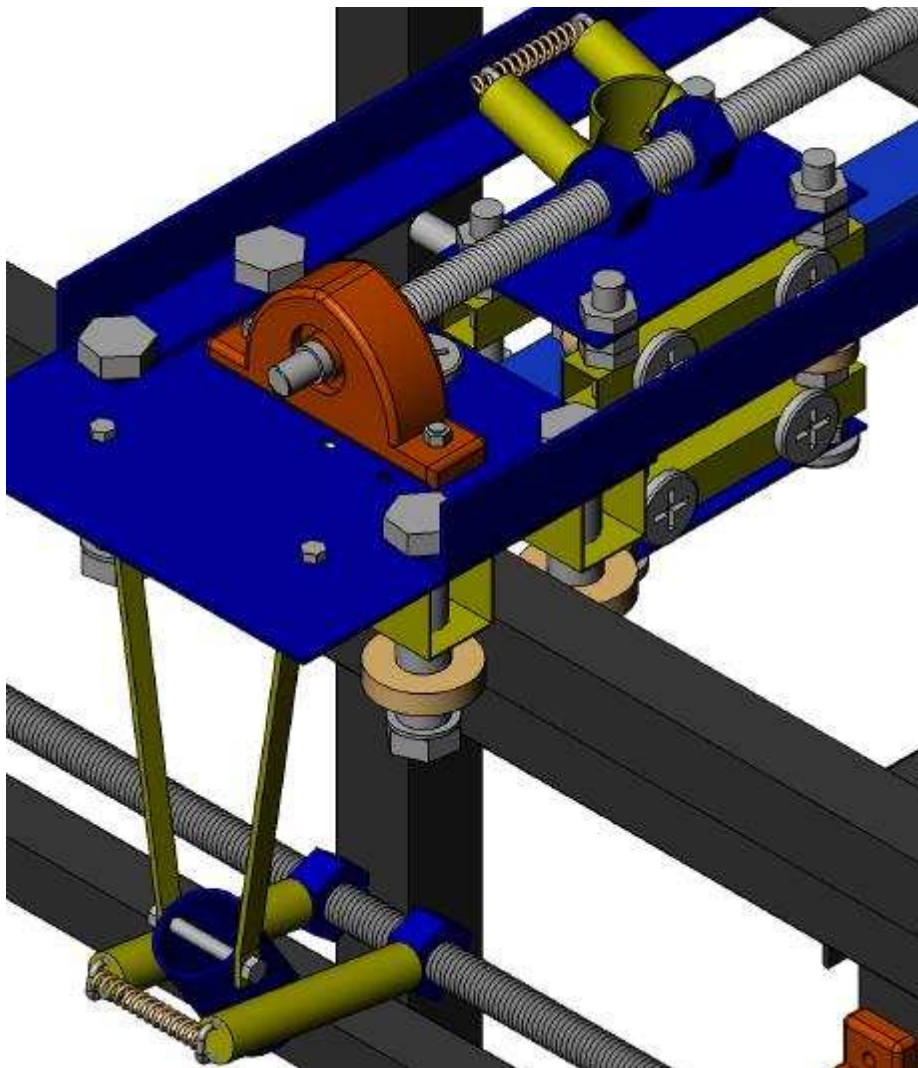


Figura 3.36 Incorporación de mecanismo tornillo tuerca en ejes longitudinal y transversal

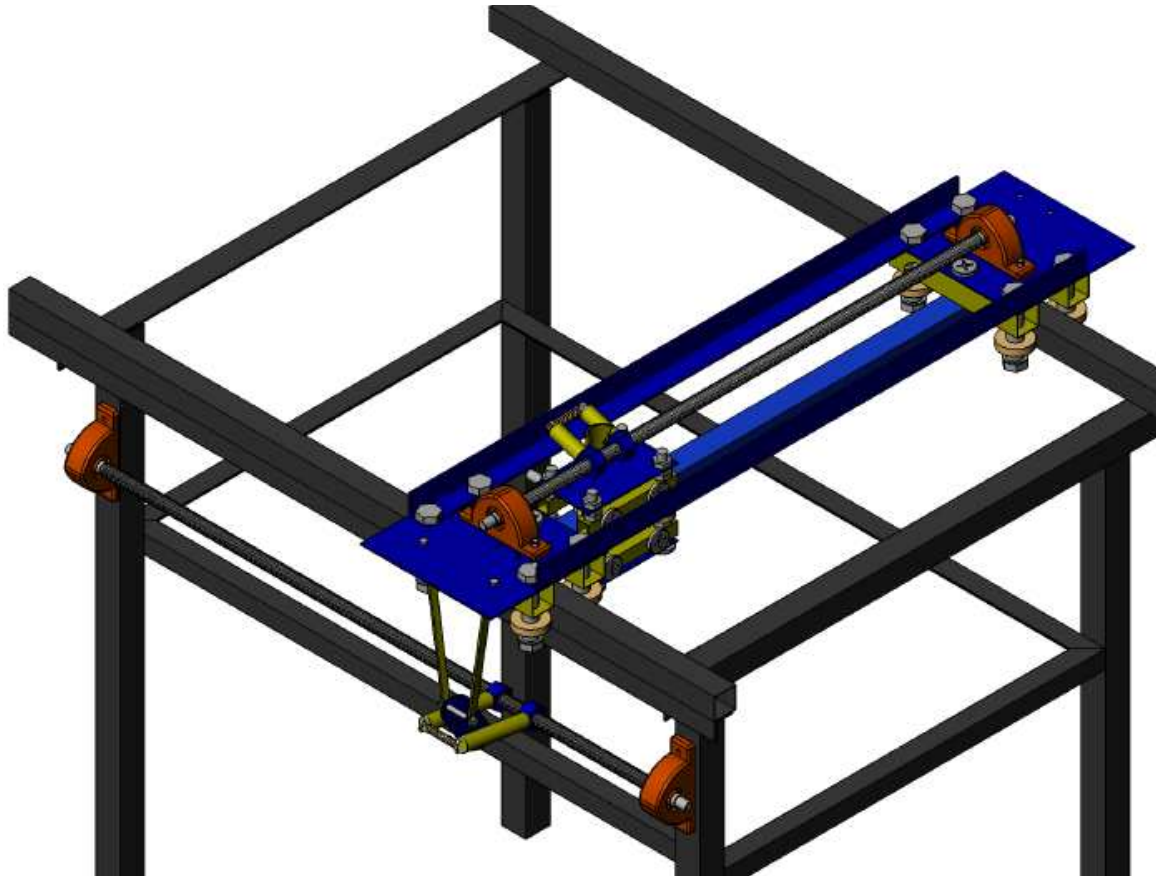


Figura 3.37 Vista completa de la mesa incluyendo mecanismo tornillo tuerca

3.2.6 SISTEMA DE MONTAJE PARA MOTORES

Se realiza un soporte encargado de sostener un motor, el cual se coloca sobre una base, y mediante piñones se acopla el movimiento del motor al tornillo sin fin.

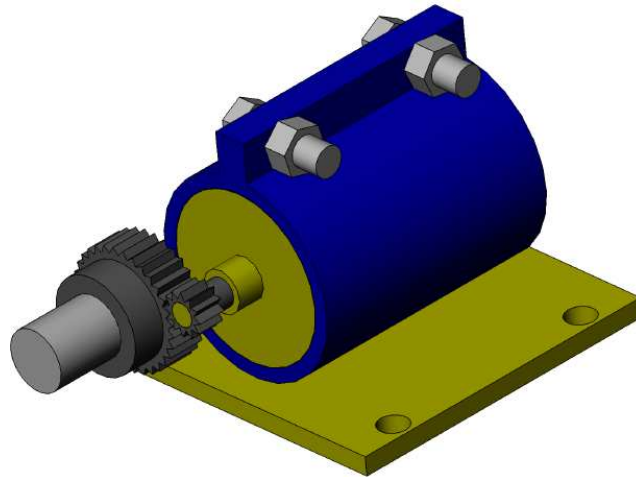


Figura 3.38 Montaje y acoplamiento de motor

3.2.6.1 Soporte Para El Motor

Se usa un tubo de 2" de diámetro, 55 mm de alto y 3 mm de espesor cortado lateralmente; en cada extremo de corte se suelda una platina con dos orificios, por los que se atraviesan dos pernos, con tuercas en sus extremos, los mismos que permiten ajustar el motor contra el soporte.

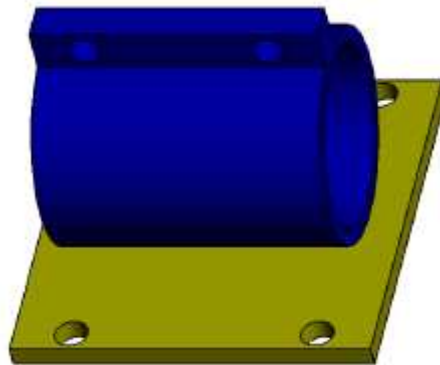


Figura 3.39 Soporte para motor. Ver ANEXO 3-10

3.2.6.2 Base Para El Motor

Se realizan dos bases, la primera para el motor encargado del movimiento en el eje lateral, se usa platina de 4 mm de espesor, 70 mm de ancho x 69 mm de

largo, soldada a lado de una chumacera en el eje longitudinal. La segunda base, está presente en un carro longitudinal encargado de dar movimiento en dicho eje.

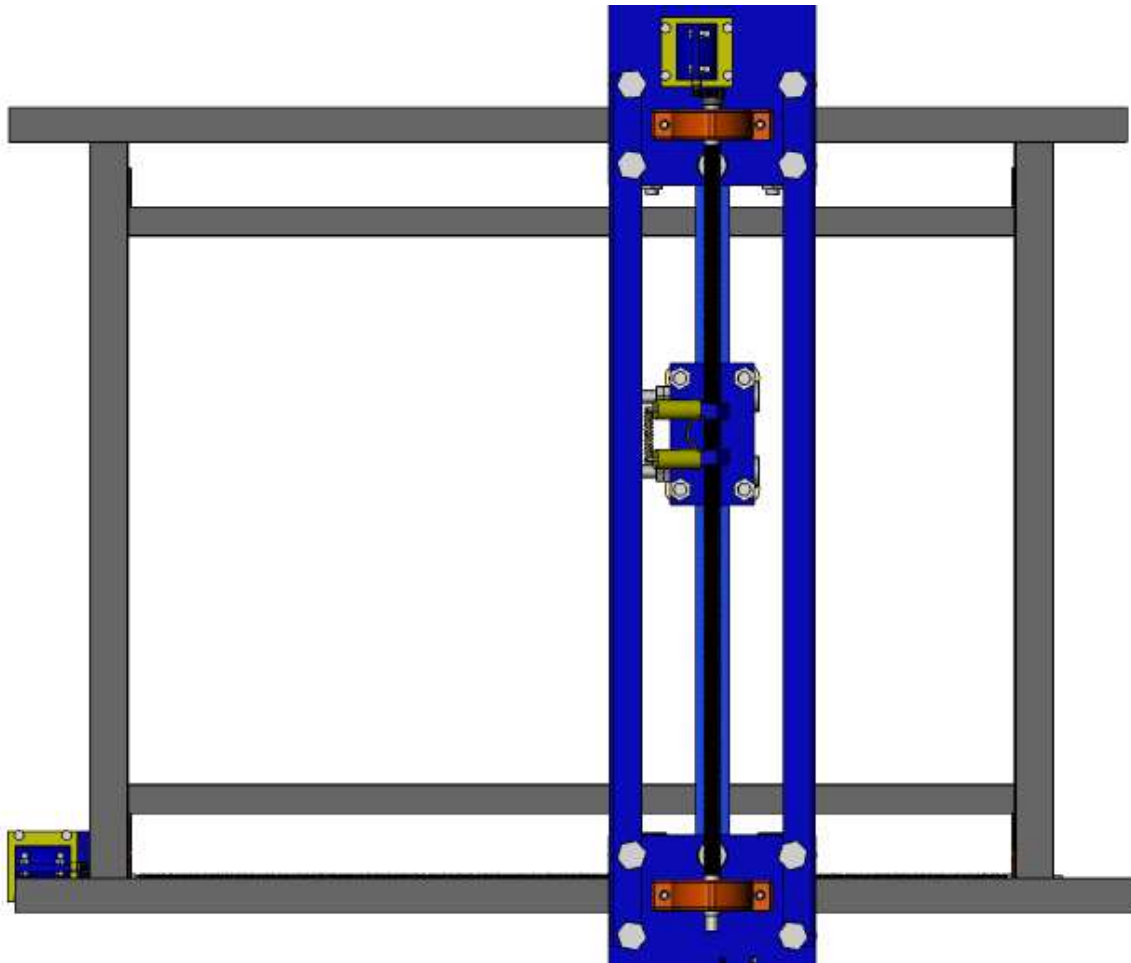


Figura 3.40 Vista superior del sistema con motores incorporados

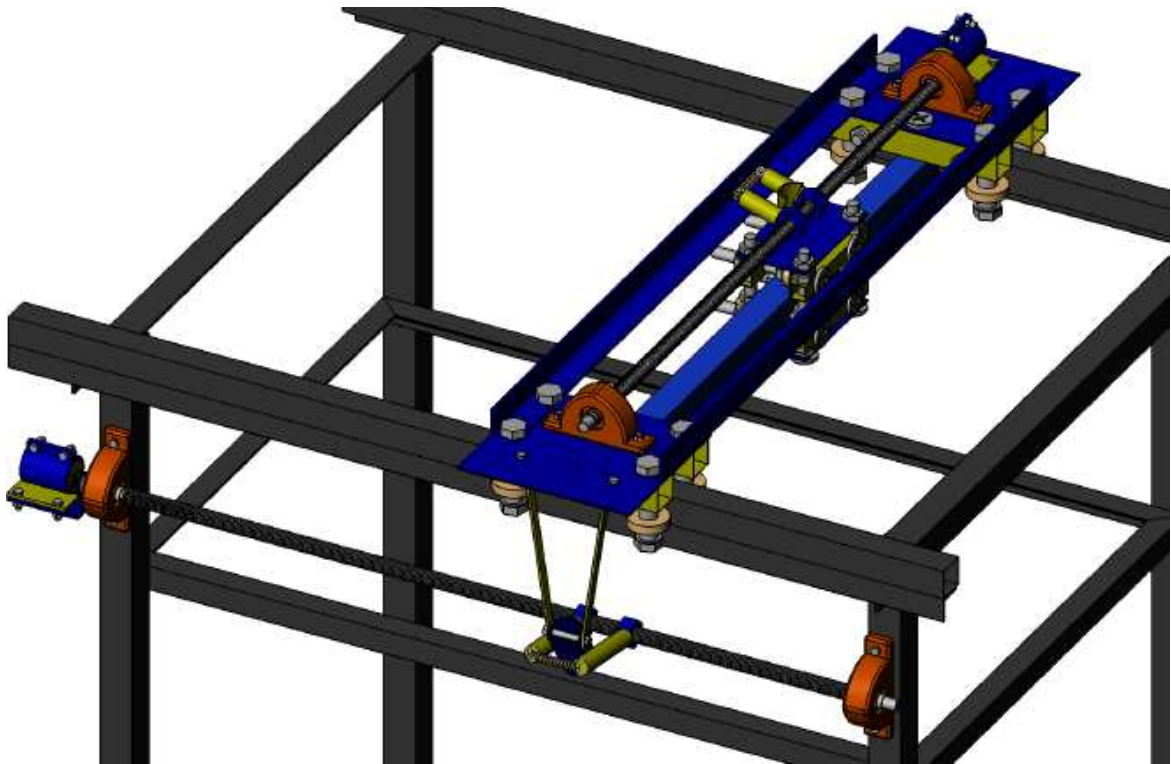


Figura 3.41 Vista completa del sistema con motores incorporados

3.2.7 SISTEMA DE MONTAJE PARA ENCODERS

El soporte para cada encoder se basa en un canal U de metal, en el cual se realizan dos orificios para acoplar el mismo a la mesa, y un corte que permite sujetar a un encoder.

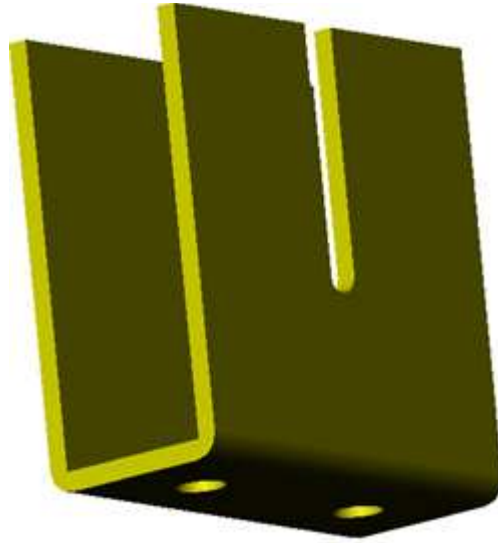


Figura 3.42 Soporte para encoder. Ver ANEXO 3-11

En el siguiente paso se acopla un piñón en el extremo de un perno sin fin y un piñón en un encoder, lo cual ayuda a transmitir el movimiento del perno sin fin hacia el encoder.

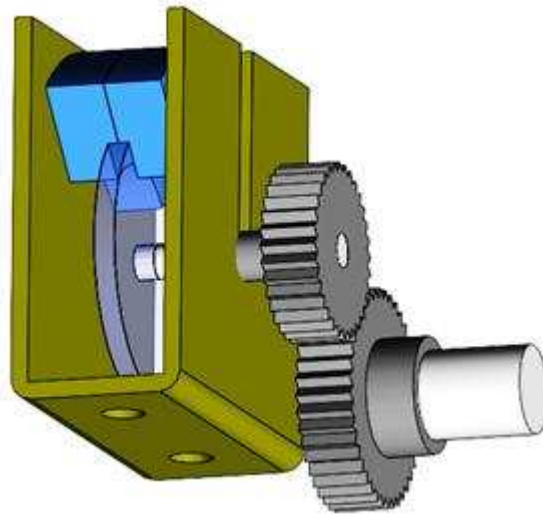


Figura 3.43 Acoplamiento de encoder a tornillo

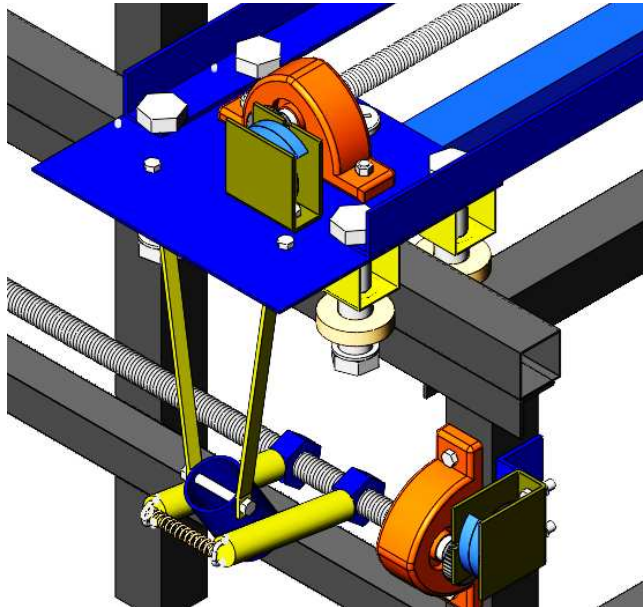


Figura 3.44 Vista de encoders acoplados al sistema

3.2.8 SOPORTE PARA SOPLETE DE MÁQUINA OXICORTE

El soporte se sujeta a la cara inferior de carro transversal.

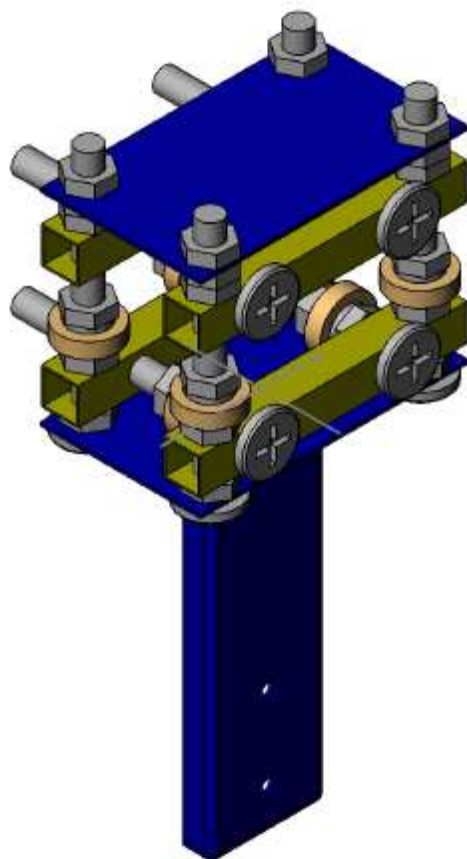


Figura 3.45 Soporte para soplete de máquina oxicorte

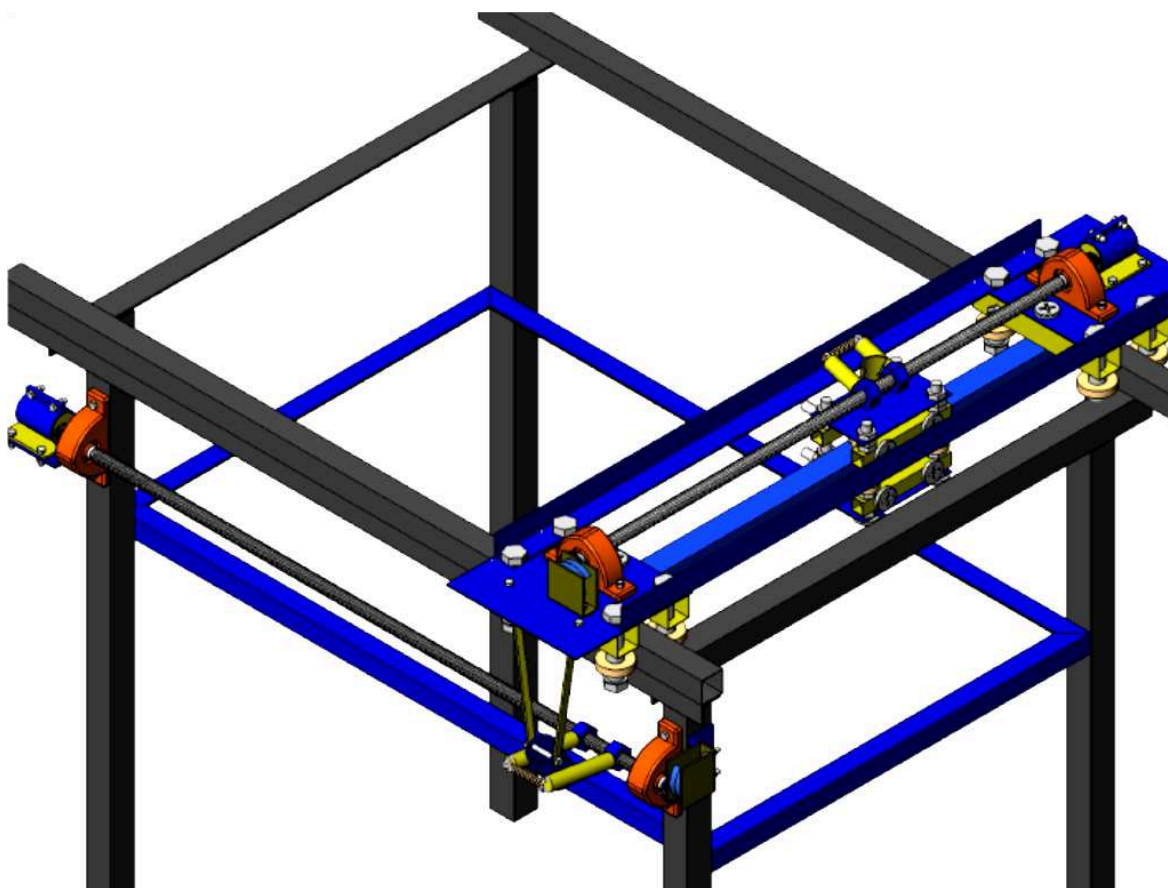


Figura 3.46 Vista del prototipo completo

CAPÍTULO 4

PRUEBAS Y ANÁLISIS DE RESULTADOS

4.1 SISTEMA DE ARRASTRE PARA LA ANTORCHA

Los principales factores a tomar en cuenta son la estabilidad y la suavidad de avance de los móviles sobre las guías de cada eje. El emplear un tipo de guía no siempre asegura una buena estabilidad y suavidad, ya que también depende del tipo de rueda o sistema de rodamientos.

4.1.1 RIEL EN “V” CON RUEDA EN “V”

Un riel en V o ángulo, garantiza una trayectoria recta, sin embargo el principal problema es la vibración que existe entre la rueda y el riel, ya que existe una leve separación entre el labrado de la rueda y el riel. Esto aplicado a sistemas de peso considerable resulta una muy buena opción debido a la suavidad que se obtiene. En este caso, al tratarse de un peso ligero, existe vibración que provoca inestabilidad de las piezas albergadas en esta parte móvil, debido a que entre el riel y la rueda no existe un engranaje perfecto.

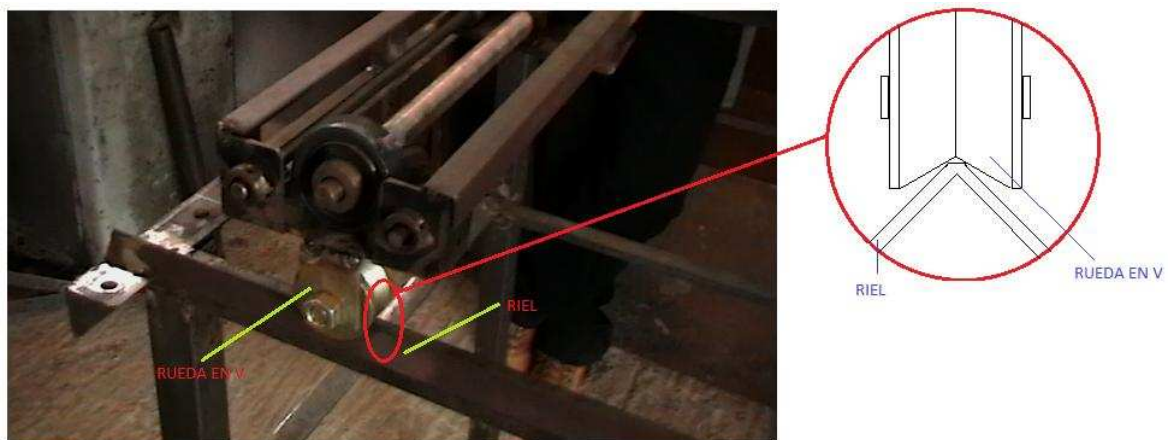


Figura 4.1 Sistema de riel en V y rueda en V

A pesar de obtenerse un recorrido suave, uno de los mayores problemas es la vibración, debido a la construcción misma de la rueda.

4.1.2 TUBO CIRCULAR CON RODAMIENTOS.

Utilizando como guía a un tubo circular y como medios móviles rodamientos, se obtiene una suavidad excelente, independientemente del peso al que esté sometido, sin embargo si no se cuenta con sujeciones precisas, los rodamientos no siempre harán contacto en su recorrido a lo largo de la guía, lo que ocasiona inestabilidad.

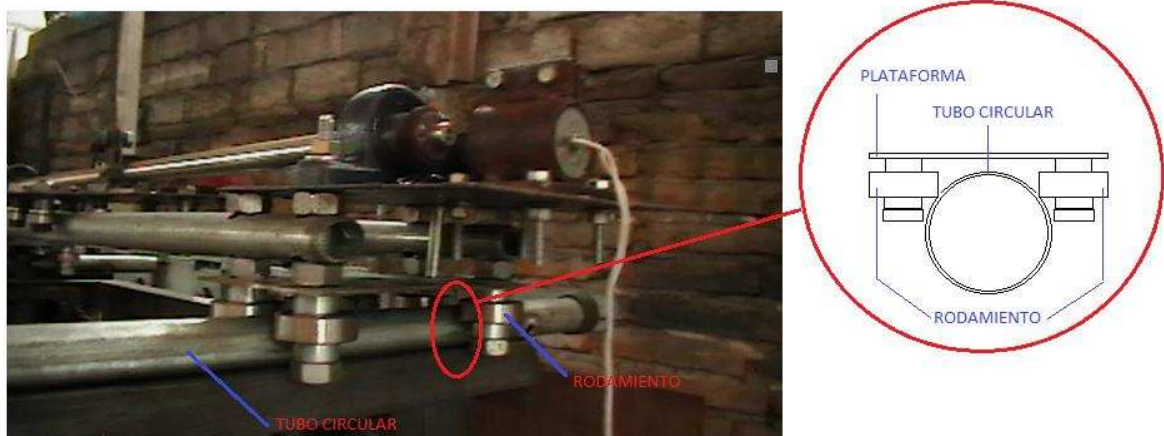


Figura 4.2 Sistema tubo circular con rodamiento.

La principal desventaja de este sistema es la perpendicularidad entre las guías y su repercusión a los ejes, es decir el sistema debe estar perfectamente alineado horizontal y perpendicularmente, ya que al no darse esto, los móviles tienden a subir o bajar de nivel con respecto al nivel superior de la mesa, este problema sobretodo en el móvil portador de la antorcha ocasiona una trayectoria a manera de ondas.

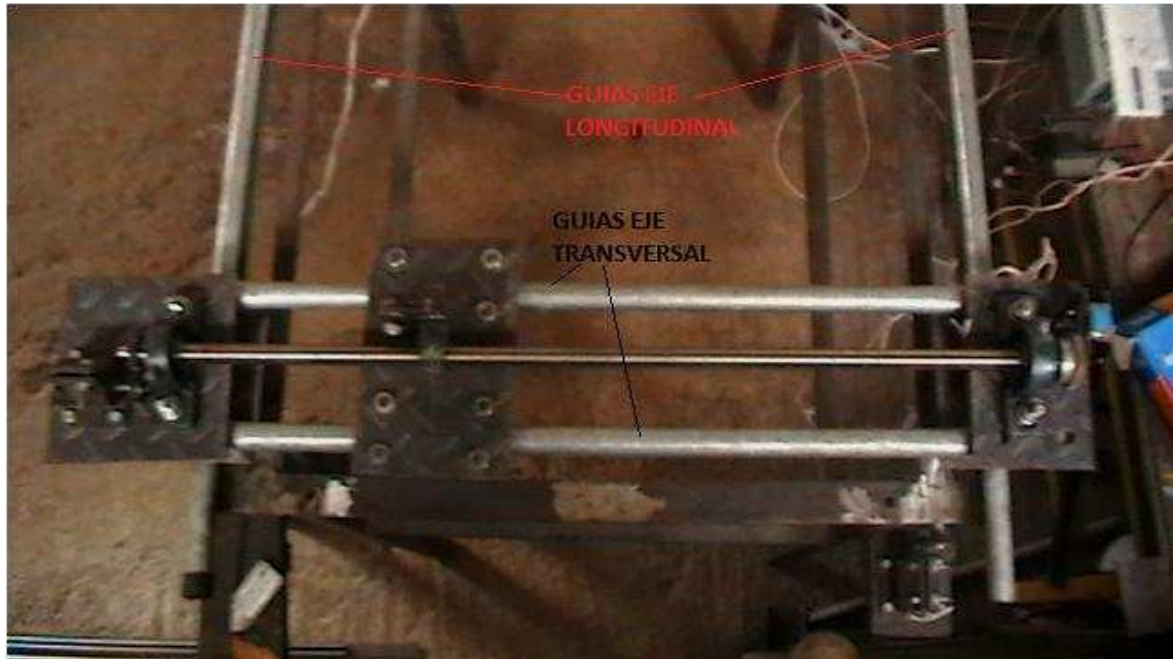


Figura 4.3 Ejes longitudinal y transversal de la mesa

Uno de los principales problemas es garantizar que las guías sean paralelas debido a que no se tiene una misma medida a lo largo de ellas, además no se garantiza que sean perfectamente rectos, esto por defectos de construcción.

4.1.3 TUBO CUADRADO CON RODAMIENTOS

A diferencia de los tubos circulares, éstos permiten tener una referencia para realizar una medida de la distancia existente a lo largo de las dos guías. Al contar con lados perpendiculares se garantiza mayor estabilidad y una trayectoria recta aunque con una suavidad relativamente baja, sin embargo es un sistema de guías muy robusto.

El eje transversal ahora cuenta con un par de soportes que son los encargados de unir a los dos móviles del eje longitudinal, mientras que el móvil portador de la antorcha se traslada por el eje transversal mediante un tubo cuadrado que se encuentra unido a los móviles del eje longitudinal mediante tornillos, lo que permite realizar una mejor calibración.



Figura 4.4 Sistema móvil de tubo cuadrado con rodamientos

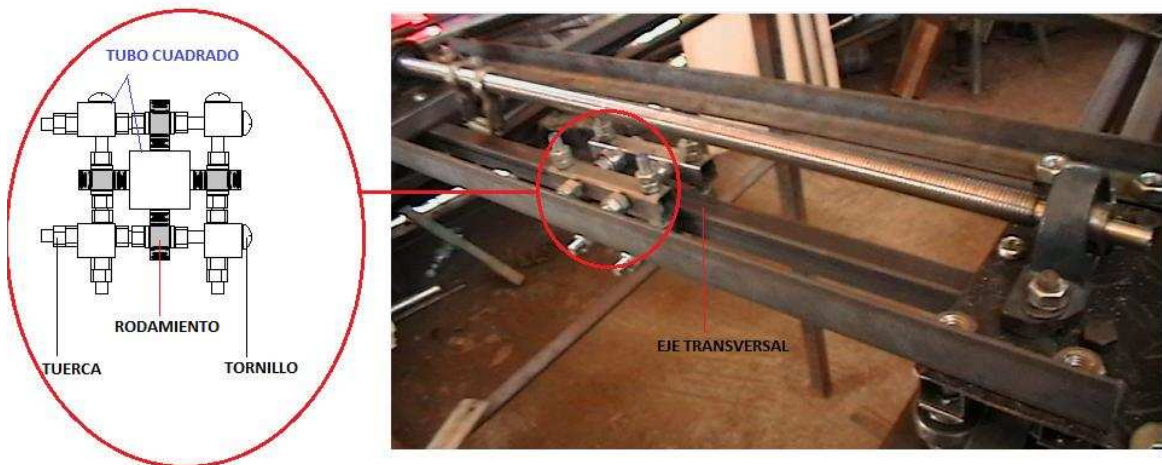


Figura 4.5 Eje transversal con sistema de móvil portador de la antorcha

Este sistema permite una mayor estabilidad, ya que cuenta con rodamientos en los cuatro lados del tubo guía, sin embargo debe corregirse la suavidad.

4.1.4 CORRECCIÓN DE AVANCE MUERTO

Al emplear una tuerca como medio móvil para arrastrar a una estructura, existe un tiempo en el que la tuerca debe acoplarse al tornillo sin fin y lograr su avance. Este problema se resuelve empleando dos tuercas separadas una distancia adecuada y unidas por un medio moderadamente flexible, en este caso se emplea un resorte para unir el par de tuercas, de esta manera se obtiene una fuerza que une a las dos tuercas, pero dependiendo de la fuerza del mismo puede darse el caso que el sistema se vuelva rígido; es por esto que se emplea un elemento que realice una fuerza opuesta a la del resorte, de esta manera el sistema se vuelve flexible y con un avance acorde al movimiento del tornillo sin fin, como se lo explica en la Figura 4.6.

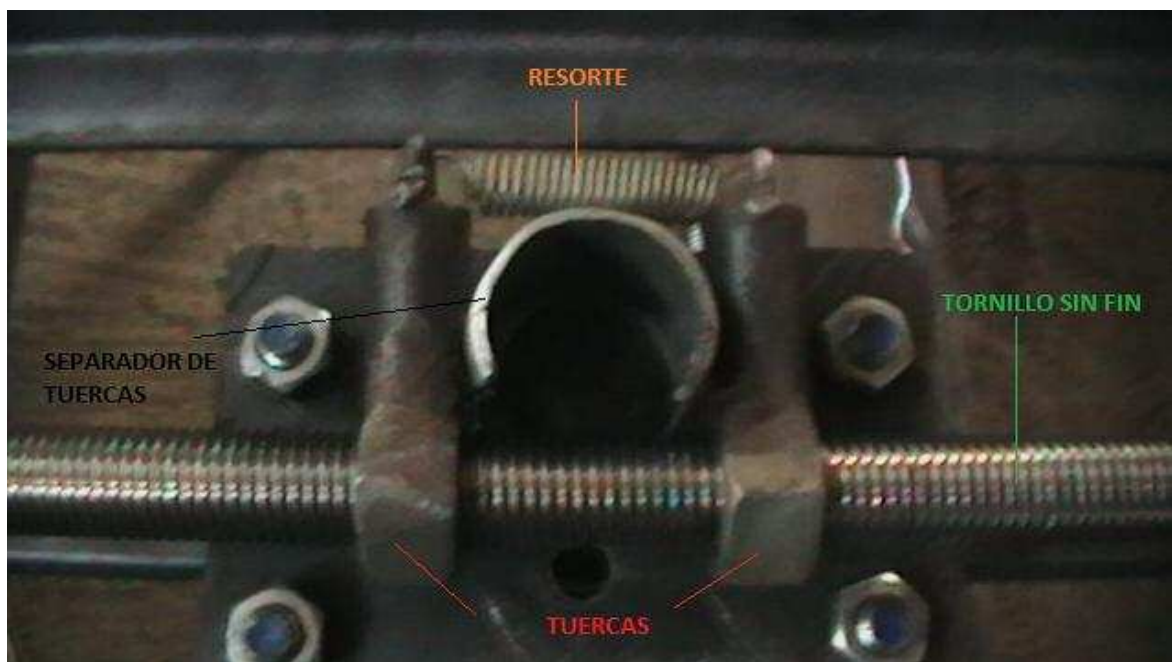


Figura 4.6 Unión de tuercas con resortes y separador

Este sistema brinda un avance del móvil independiente de la curvatura del tornillo sin fin, así como también independiente si el tornillo sin fin y la guía sean perfectamente paralelos, como se muestra en la Figura 4.7.

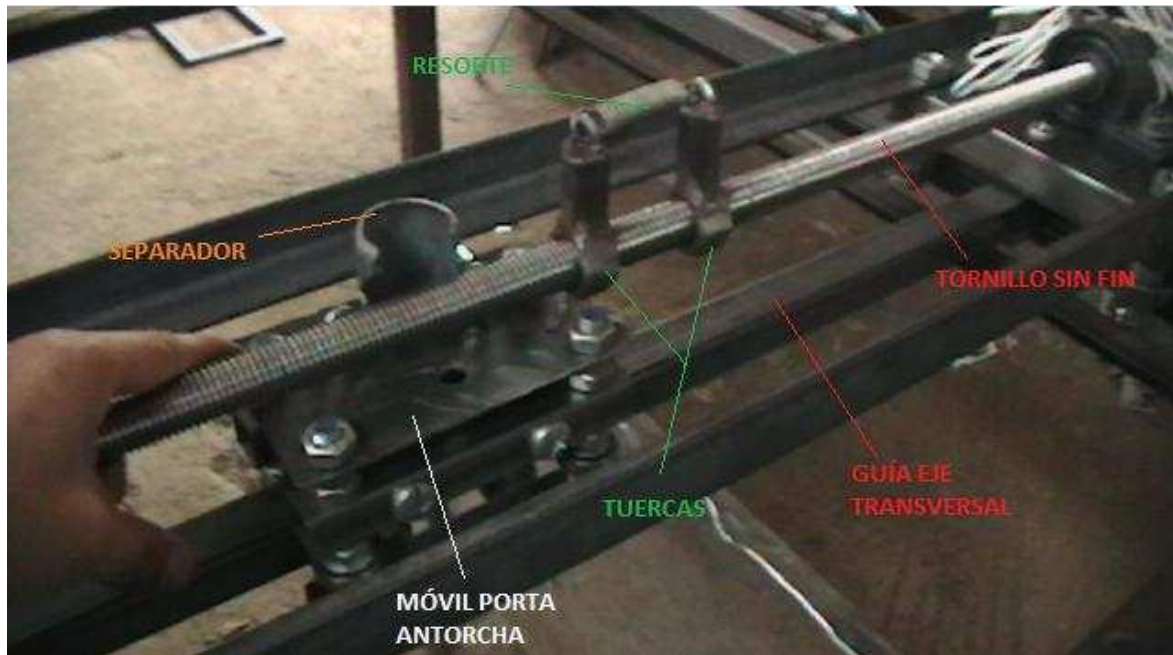


Figura 4.7 Sistema móvil porta antorcha acoplado al tornillo sin fin

4.2 ACONDICIONAMIENTO DE ENCODERS INCREMENTALES

4.2.1 OBTENCIÓN DE LA SEÑAL

El empleo de este tipo de encoder requiere de una etapa de acondicionamiento, ya que la señal de salida es de tipo analógico - sinusoidal y depende de la frecuencia de rotación. El empleo de encoders incrementales con altos impulsos comienza a tener problemas desde muchos puntos de vista, tal es el caso de tratar la señal digital al trabajar a altas velocidades, pues la frecuencia obtenida será alta [12]. Las señales entregadas por el encoder se encuentran desfasadas 90° una de la otra, como se muestra en la Figura 4.8.

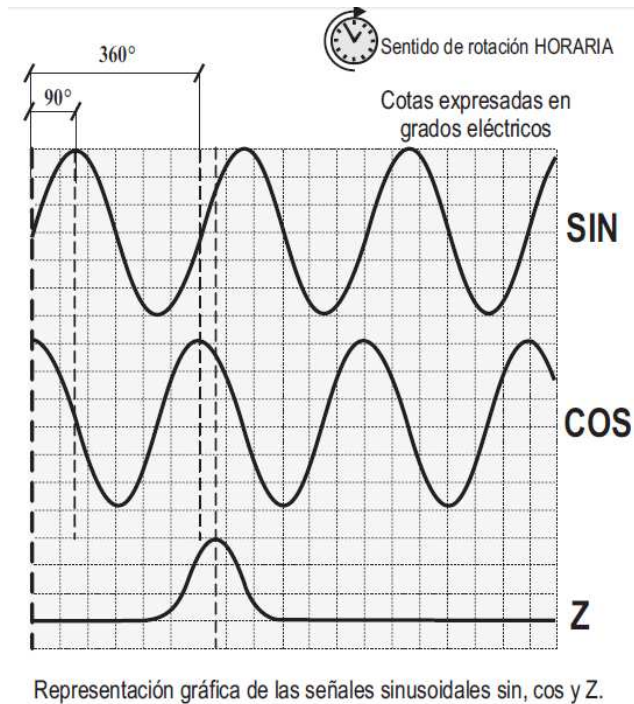


Figura 4.8 Representación gráfica de las señales sinusoidales. Tomado de ELTRA

4.2.2 CIRCUITO DE ANCONDICIONAMIENTO

Se observa que la calidad de la señal de salida depende tanto de la velocidad de giro así como también del voltaje aplicado a los terminales de colector-emisor del foto transistor, es por esto que se realiza la polarización de los elementos con 15 Vdc y de esta manera se obtiene una amplitud apreciable para realizar una etapa de comparación y obtener una señal binaria.

La amplitud de la señal de salida también depende del valor de la resistencia que se emplee para cerrar el circuito entre colector y emisor. Emplear una resistencia de alto valor como 100 k Ω provocará que el mayor voltaje DC caiga sobre la misma, de esta manera la señal sinusoidal será casi plana y con un offset alto, por tanto, no se obtiene una señal con una amplitud apreciable y sobre todo no se distingue un rango de la señal en donde se pueda realizar una comparación.

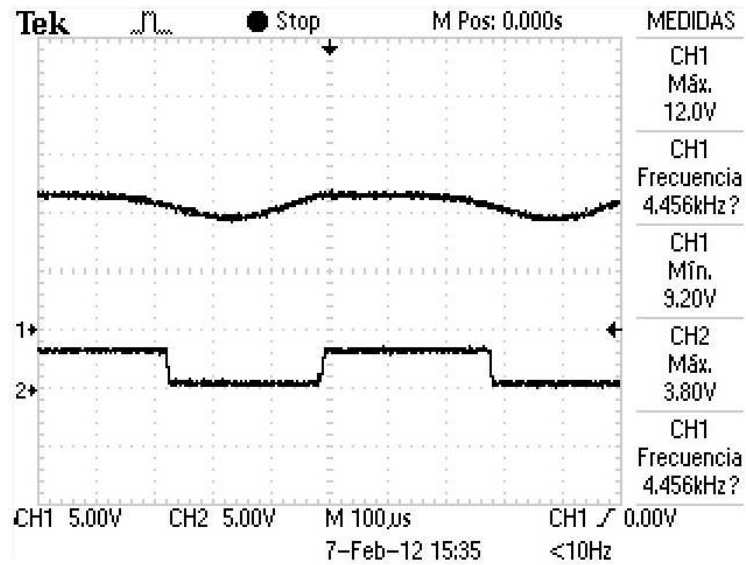


Figura 4.9 Señal de salida del encoder con alta velocidad y resistencia de valor alto

Como se observa en la Figura 4.9, no se garantiza una señal que permita realizar una comparación adecuada y obtener la señal de dos estados con periodo constante. La solución es emplear una resistencia de menor valor, en este caso 22 k Ω , que al contrario de la resistencia de mayor valor no alcanzará un valor máximo que sea próximo al voltaje de polarización, y por tanto, el offset será mucho menor.

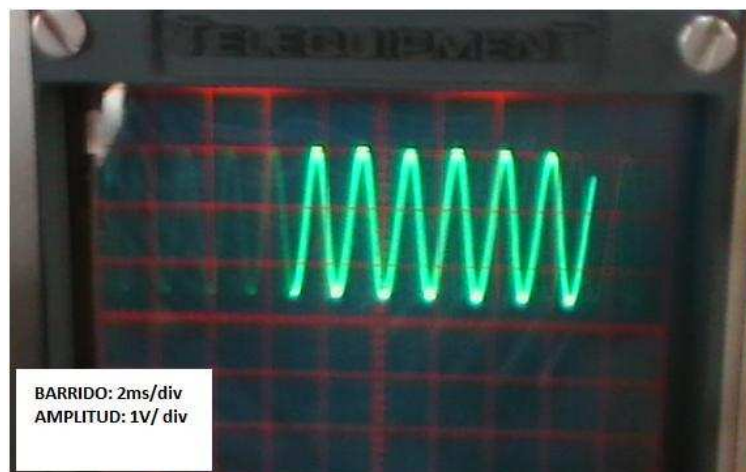


Figura 4.10 Señal de salida de encoder con resistencia de 22 k Ω

Es importante aclarar que por características propias de cada encoder, a pesar de ser de un mismo fabricante, la respuesta es distinta para cada uno, es por esto que deben emplearse voltajes de referencia RV2 y RV3 para la etapa de comparación.

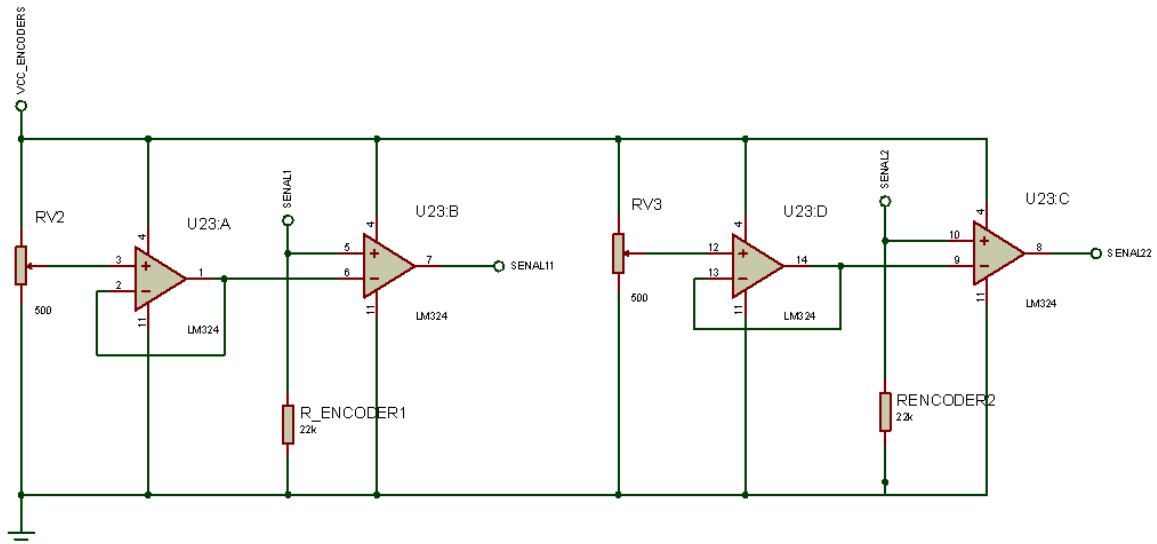


Figura 4.11 Etapa de comparación

Los terminales denominados SENAL1 y SENAL2 corresponden a los terminales de salida de los encoder de los ejes transversal y longitudinal, respectivamente.

La última etapa es convertir la señal a un nivel de voltaje con el que un microcontrolador pueda trabajar, para esto se emplea un Schmitt trigger junto con una etapa configurada como seguidor.

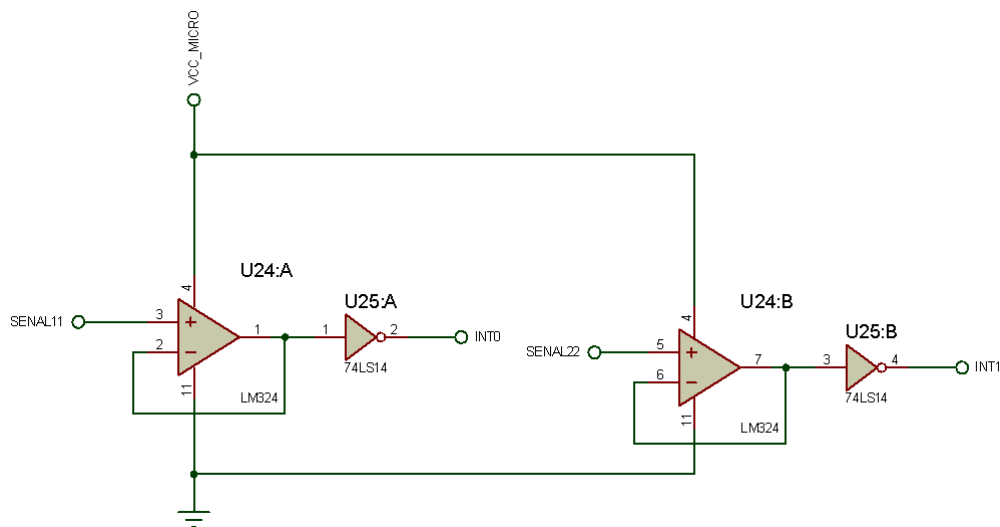


Figura 4.12 Obtención de señales de encoder a nivel TTL

4.2.3 VELOCIDAD DE GIRO DEL ENCODER

La velocidad aplicada al eje del encoder hará que la frecuencia de la señal de salida se encuentre en el orden de los kHz, esto repercute en la respuesta de la propia salida, es decir que a pesar de que se obtenga una señal en la misma, no se tendrá una señal puramente sinusoidal ni de amplitud constante, es por esto que la velocidad debe ser tal que no se pierda definición de pulsos en un recorrido de 1 mm ni exista distorsión de la señal. Para cumplir con estas dos condiciones se debe variar la velocidad aplicada al eje del encoder. Esto se hace mediante el acoplamiento de piñones a los ejes.

4.2.3.1 Primer Acoplamiento

Este primer acoplamiento consta de una relación 30:30 entre los piñones del motor y el tornillo sin fin, y con un sistema de piñones entre el tornillo sin fin y el encoder de relación 50:12, la frecuencia se obtiene:

$$\omega_1 \times n_1 = \omega_2 \times n_2$$

Donde:

ω : Velocidad [rpm]

n : Número de dientes de los piñones

$$200 \times 50 = \omega_2 \times 12$$

$$\omega_2 = \frac{833.33 \text{ [rpm]}}{60} = 13.88 \text{ [rev/seg]}$$

Los encoders tienen una resolución de 250 pulsos por revolución por tanto:

$$f = \omega_2 \times N$$

Donde:

f : Frecuencia de pulso

N : Número de pulsos por revolución

$$f = 13.88 \times 250 = 3472.22 \text{ Hz}$$

Teniendo un periodo de:

$$T = \frac{1}{f} = 288 \mu\text{s}$$

Al hacer una relación entre los dos piñones, por una revolución se tiene:

$$\omega_1 \times n_1 = \omega_2 \times n_2$$

$$1 \times 50 = \omega_2 \times 12$$

$$\omega_2 = \frac{50}{12} \text{ [rev]}$$

Multiplicando este valor por la resolución del encoder se obtiene el número de pulsos por cada revolución del tornillo sin fin, por tanto éste será el número de pulsos equivalente a un desplazamiento de 2.54 mm.

$$N = 250 \times \omega_2$$

$$N = 250 \times \frac{50}{12} \approx 1042 \text{ pulsos/rev}$$

Es decir se tienen 1042 pulsos en un desplazamiento de 2.54 mm, por tanto el número de pulsos por cada milímetro será:

$$N_{1mm} = \frac{N}{2.54} = \frac{1042}{2.54} \approx 410 \text{ pulsos}$$

Por lo que, el tiempo entre el envío y confirmación de un nuevo dato de la DAQ6501 es de:

$$t = T \times N_{1mm} = 288us \times 410 = 118ms$$

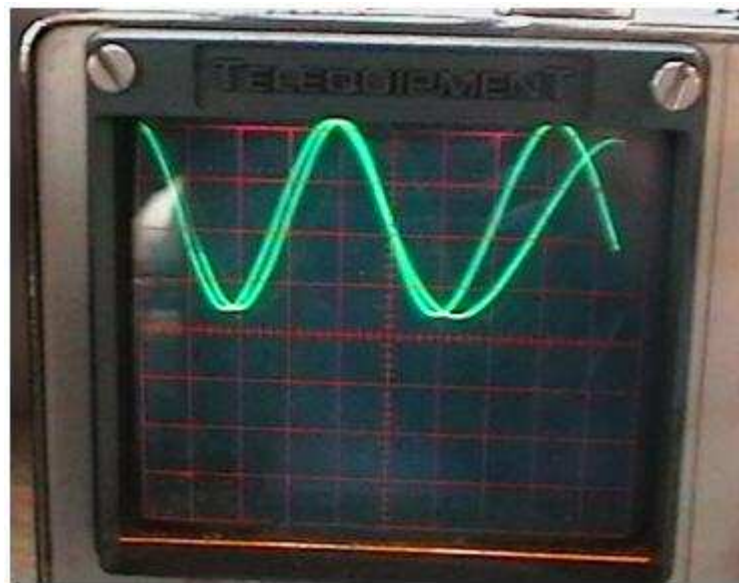


Figura 4.13 Señal de salida del encoder con una relación 30:30 entre el motor y el tornillo sin fin

4.2.3.1.1 Resultados

El principal problema encontrado es la inercia existente al apagar el motor, por tanto aún con la señal de apagado, el encoder es capaz de detectar varias cuentas, lo que en una trayectoria que requiera de una cantidad considerable de valores existirán errores acumulados, por lo que existirán desfases.

Una ventaja es la resolución obtenida, ya que por cada milímetro recorrido se requieren de 410 pulsos.

Al emplear piñones con el mismo número de dientes, el torque aplicado al tornillo es el mismo del motor, lo que provoca que la corriente incremente.

4.2.3.2 Segundo Acoplamiento

Este acoplamiento consta de una relación 12:30 entre los piñones del motor y el tornillo sin fin y con un sistema de piñones entre el tornillo sin fin y el encoder de relación 50:30, la frecuencia se obtiene:

$$\omega_1 \times n_1 = \omega_2 \times n_2$$

$$200 \times 50 = \omega_2 \times 30$$

Donde:

ω : Velocidad [rpm]

n : Número de dientes de los piñones

$$\omega_2 = \frac{333.33 \text{ [rpm]}}{60} = 5.56 \text{ [rev/seg]}$$

Los encoders tienen una resolución de 250 pulsos por revolución por tanto:

$$f = \omega_2 \times N$$

Donde:

f : Frecuencia de pulso

N : Número de pulsos por revolución

$$f = 5.56 \times 250 = 1390 \text{ Hz}$$

Teniendo un periodo de:

$$T = \frac{1}{f} = 719.4 \mu\text{s}$$

Al hacer una relación entre los dos piñones por una revolución se tiene:

$$\omega_1 \times n_1 = \omega_2 \times n_2$$

$$1 \times 50 = \omega_2 \times 30$$

$$\omega_2 = \frac{5}{3} [rev]$$

Multiplicando este valor por la resolución del encoder se obtienen el número de pulsos por cada revolución del tornillo sin fin, por tanto este será el número de pulsos equivalente a un desplazamiento de 2.54 mm.

$$N = 250 \times \omega_2$$

$$N = 250 \times \frac{5}{3} \approx 417 \text{ pulsos/rev}$$

Es decir, se tienen 417 pulsos en un desplazamiento de 2.54 mm, por tanto el número de pulsos por cada milímetro será:

$$N_{1mm} = \frac{N}{2.54} = \frac{417}{2.54} \approx 164 \text{ pulsos}$$

Por lo que, el tiempo entre el envío y confirmación de un nuevo dato de la DAQ6501 es de:

$$t = T \times N_{1mm} = 719.4us \times 164 \approx 118ms$$

4.2.3.2.1 Resultados

En este caso al emplear un sistema de piñones con una relación 12:30 entre el motor y el tornillo sin fin el torque incrementa en 2.5 veces el torque del motor, por lo que la corriente del mismo es menor, además la inercia propia del motor en este caso ya no es transmitida directamente al tornillo sin fin, y por tanto, el encoder ya no posee mayor sensibilidad, de esta manera se eliminan las cuentas de pulsos indeseables.

Una desventaja de este sistema es la relativamente baja resolución de pulsos por milímetro recorrido, sin embargo se obtienen buenos resultados aún con contornos circulares.

4.3 REALIMENTACIÓN HACIA DAQ6501

El algoritmo del envío del código G se basa en un lazo de repeticiones, en el que los datos a enviar se encuentran en una matriz, el algoritmo envía un dato cada vez que se realiza una confirmación dada por el microcontrolador que realimenta una señal a la DAQ6501, indicando que se recorrió 1 mm. Mediante pruebas se determina que a medida que los valores enviados desde la matriz que contiene el código G, van avanzando en posición el programa responde con suma lentitud, por tanto los perfiles a dibujar se deforman.

4.3.1 TIEMPO DE RESPUESTA DEL PROGRAMA CON ANIMACIÓN DEL PROCESO

Como principales causas se determina que influyen los ciclos de máquina empleados para realizar la animación del proceso, ya que debe realizarse una actualización de la ventana donde se grafica la trayectoria, esto con cada punto del contorno, además de enviar un dato de la matriz que contiene el código G, código que es enviado después de barrer la matriz desde el inicio hasta llegar al elemento que corresponde ser enviado. Por ejemplo;

Si el elemento a enviar está en la posición 2056, el programa barre la matriz contenedora del código G desde el elemento 1, y al tratarse de contornos de un considerable número de puntos los resultados son a gran escala.

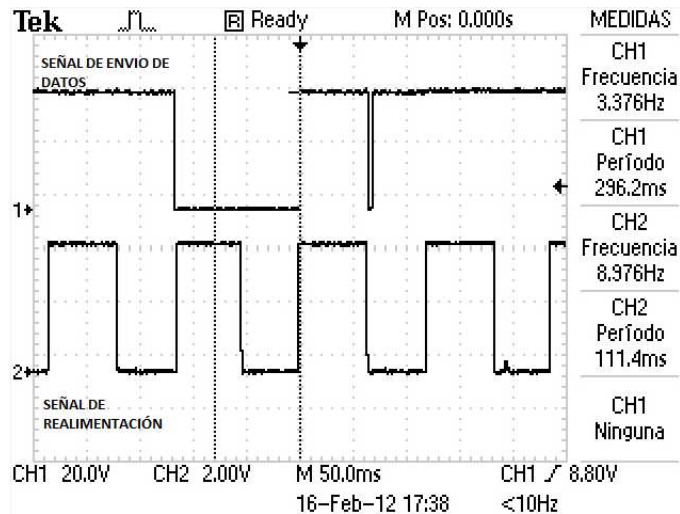


Figura 4.14 Señales de envío y realimentación de datos

La primera señal indica el tiempo que tarda el programa en enviar un dato, mientras que la segunda señal a manera de señal ideal, generada por un micro controlador con periodo y tiempo en alto constante, indica el pulso de confirmación de avance de 1 mm; como se observa en la Figura 4.14 existe un desfase entre las dos señales, lo que indica que se pierden datos, es decir que la distancia recorrida es mucho mayor, ya que no se logra detectar en el programa la confirmación de este recorrido.

4.3.2 REDUCCIÓN DE CICLOS DE MÁQUINA

Para obtener una respuesta óptima del programa frente a la señal de realimentación, se deben reducir los ciclos de máquina empleados para presentar la trayectoria de corte de la antorcha en una ventana, ya que este algoritmo depende de un lazo de repetición en el que debe presentarse el punto actual en el que la antorcha se encuentra, así como también los puntos anteriores. Para evitar este inconveniente se realiza un algoritmo que presente la trayectoria total de corte que se va a realizar, diferenciando trayectorias de posicionamiento y las trayectorias de corte que son de color azul y rojo, respectivamente.

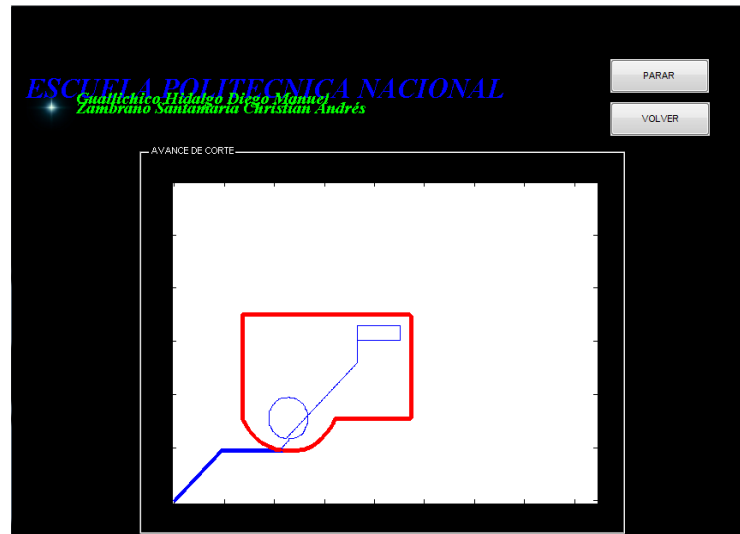


Figura 4.15 Trayectorias de posicionamiento y corte

El algoritmo de confirmación de posicionamiento de 1 mm depende de dos entradas digitales de la DAQ6501 en forma alternada, es decir existen dos señales de realimentación complementadas, por tanto para una señal de realimentación en alto deben existir dos estados en alto de la señal de envío de datos, en donde el tiempo en alto indica el tiempo de ejecución del programa mientras se envía un dato como se muestra en la Figura 4.16.

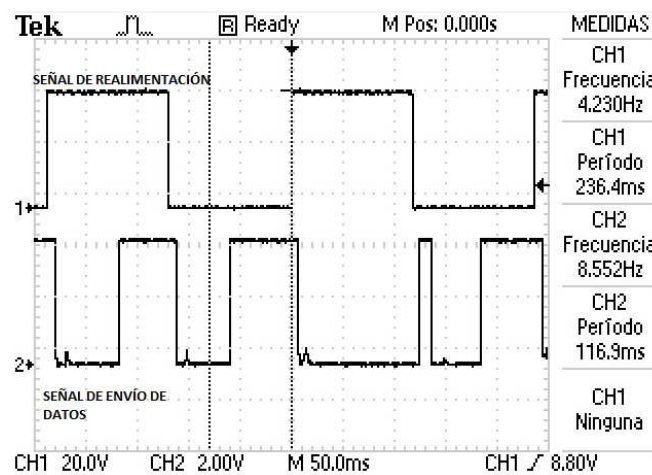


Figura 4.16 Señales de realimentación y de envío de datos con reducción de ciclos de máquina

4.4 CONTROL DE LA TRAYECTORIA DE CORTE

El tipo de trayectoria que se emplee para realizar un corte o posicionamiento implica optimizar tiempo u obtener buena definición en el contorno a cortar, es por esto que se contemplan dos opciones.

4.4.1 TRAYECTORIA DIAGONAL

Con un desplazamiento diagonal se emplea menor tiempo tanto en el corte en sí como en el posicionamiento de la antorcha, sin embargo la definición de corte no es fina, una idea a gran escala se la representa en la Figura 4.17.

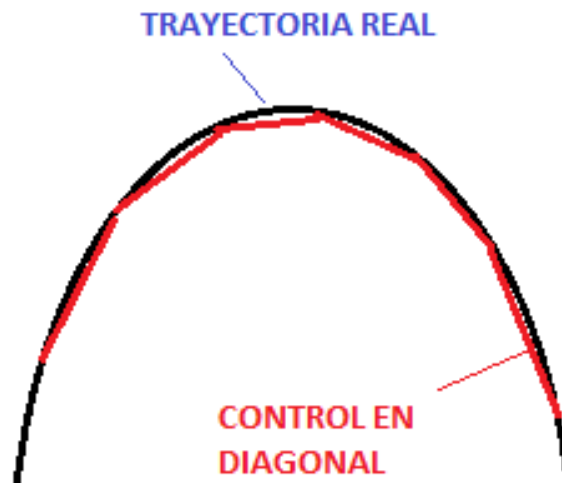


Figura 4.17 Trayectoria de corte diagonal

Una condición necesaria para obtener un buen control en el desplazamiento diagonal es asegurar una sincronización perfecta de los motores, lamentablemente el tiempo de respuesta de los motores es muy lento, aún empleando un control PID, en comparación al tiempo que en el que se debe realizar la detección del posicionamiento, que está en el orden de los milisegundos. Una de las consecuencias al no contar con velocidades iguales de avance es obtener un perfil a cortar alargado en uno de los ejes, como se muestra en la Figura 4.18, el eje longitudinal tiene mayor velocidad.



Figura 4.18 Resultado del control de trayectoria en diagonal

Un problema con este tipo de trayectoria es el posicionamiento, ya que si uno de los ejes avanza una mayor distancia en un mismo tiempo no se logrará llegar al punto correcto para iniciar un corte, en especial con cortes internos, esto debido a los errores acumulados durante cualquier instante de las trayectorias.

4.4.2 TRAYECTORIA EN ESCALONES

A diferencia del desplazamiento diagonal se emplea mayor tiempo en el corte y en el posicionamiento de la antorcha, pero un buen punto a favor es la definición de corte que se obtiene, una idea a gran escala se la representa en la Figura 4.19.

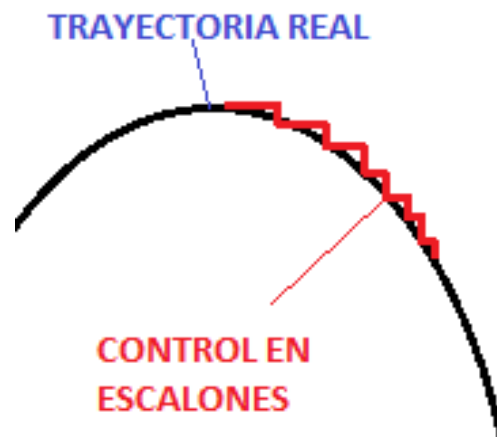


Figura 4.19 Control de trayectoria en escalones

En este caso la trayectoria de corte no depende de la velocidad de los motores, simplemente depende del conteo de los pulsos necesarios para detectar un desplazamiento de 1 mm.

De este modo la definición obtenida es mucho más fina, incluso en perfiles circulares.



Figura 4.20 Figura realizada con control de trayectoria en escalones

Como se observa, a diferencia de la Figura 4.18, este perfil (Figura 4.20) tiene mayor definición y trayectorias más cercanas a una circunferencia, así como también el posicionamiento final del contorno.

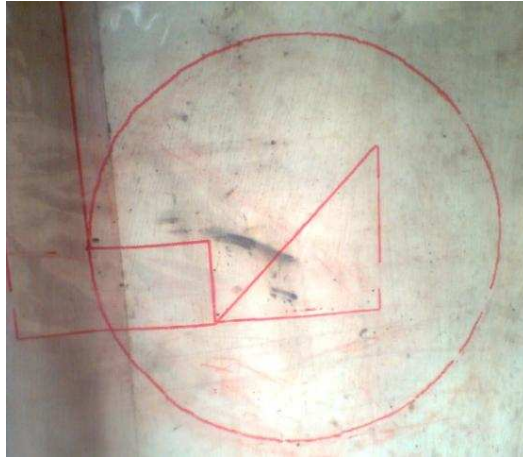


Figura 4.21 Resultado de contorno circular de radio 18.5 cm

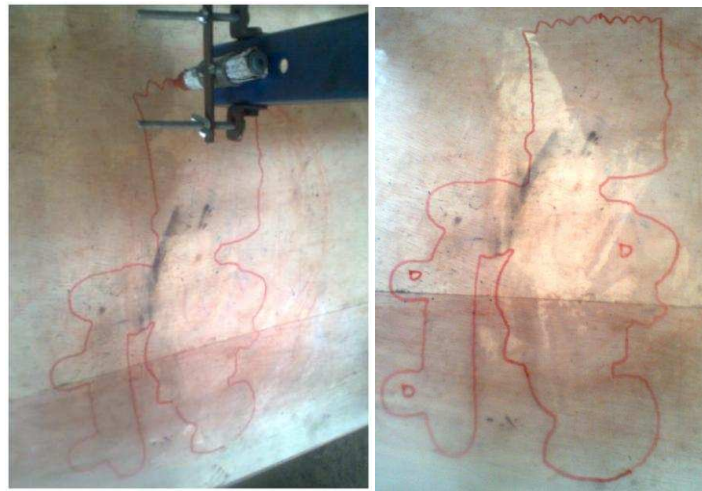


Figura 4.22 Dibujo de contornos, figuras artísticas

4.5 CALIBRACIÓN DEL SISTEMA

Debe garantizarse que en el sistema mecánico no existan vibraciones ni trayectorias rígidas, también debe cumplirse que los ejes mantengan siempre 90° a lo largo del eje longitudinal, y ajustar el número de pulsos que indiquen un desplazamiento de 1 mm.

4.5.1 PERPENDICULARIDAD ENTRE EJES

Los ejes deben mantener 90° entre ellos, de no cumplirse con esto los contornos a dibujar se deforman, una representación de este problema se observa en la Figura 4.23, donde se ilustra un supuesto cuadrado, se puede notar que a pesar de recorrerse una distancia igual la figura claramente no tiene ángulos rectos, por tanto si se tratara de una figura a cortar más compleja pudo haberse dado el caso en el que el contorno no se cierre.

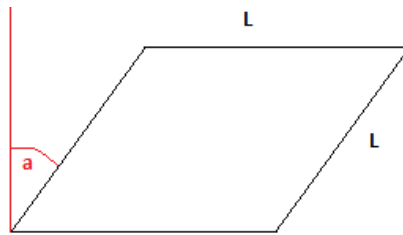


Figura 4.23 Representación de un cuadrado obtenido con ejes no perpendiculares

Para lograr que los ejes estén perpendiculares entre ellos se emplea un tornillo que pueda desplazarse en un agujero de una longitud, tal que el tubo cuadrado permita ajustar dicho ángulo.

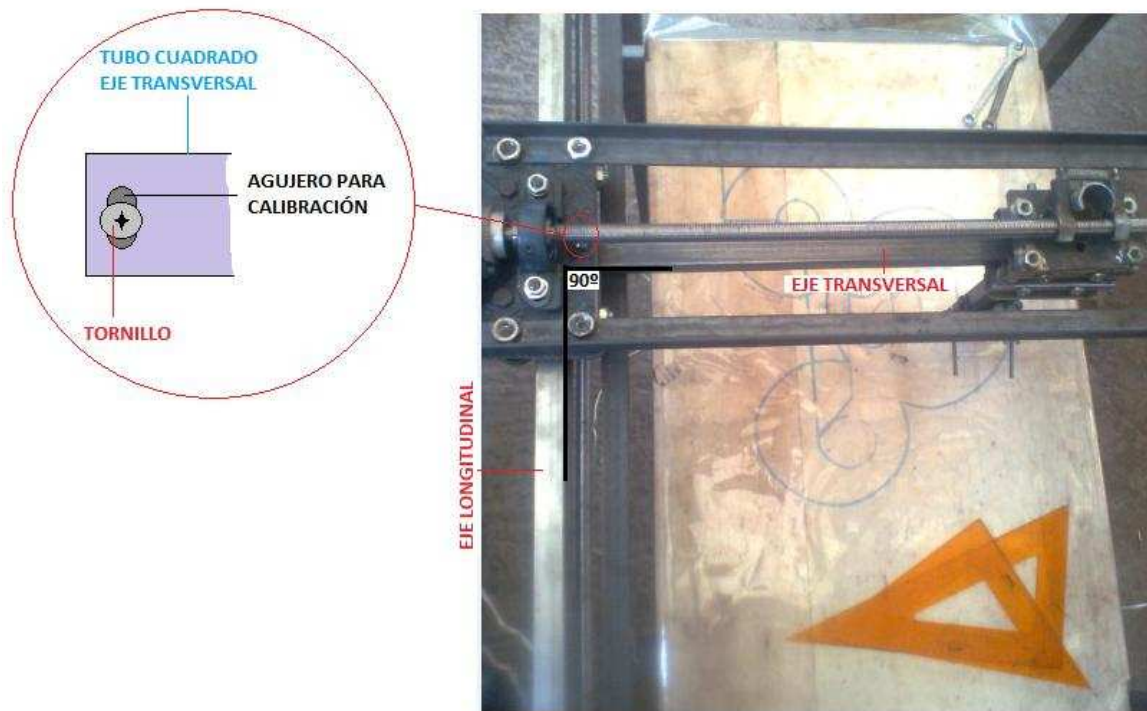


Figura 4.24 Modo de calibración a 90° entre el eje longitudinal y transversal

4.5.2 DETECCIÓN DEL DESPLAZAMIENTO DE 1 mm

En forma ideal los pulsos que se necesitan para detectar un desplazamiento de 1 mm son 164, pero debido a que los tornillos no son perfectamente rectos se debe variar en un pequeño rango el número de pulsos en cada eje.

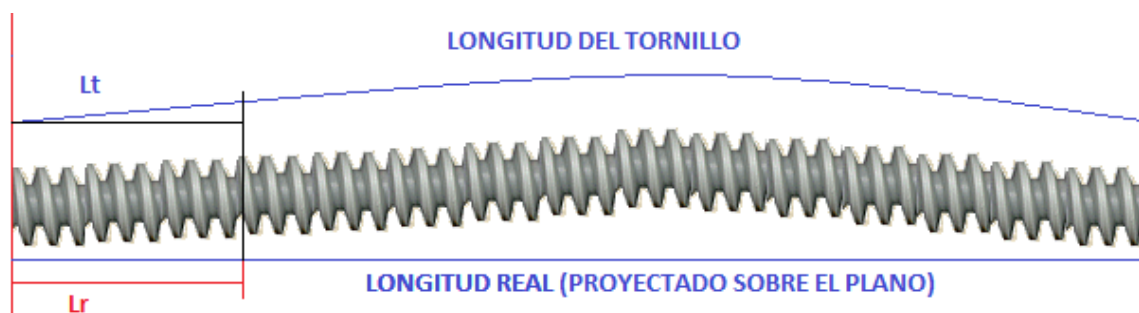


Figura 4.25 Avance real debido a la curvatura del tornillo

Como se observa en la Figura 4.25, no se tiene una correspondencia directa entre el avance del tornillo y el avance sobre la proyección en el plano, lo que indica que deben existir un número mayor de pulsos que el que se indica en los cálculos. Esta calibración necesariamente debe realizarse mediante el método prueba-error.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- En la actualidad la necesidad de la industria en cuanto a la eficiencia y productividad pueden ser satisfechas por sistemas que combinan la mecánica con el control electrónico y digital, comandados por un software y una interfaz amigable al operador, tal es el caso de los sistemas CNC ofertados por varios fabricantes que van desde el software hasta sistemas dedicados que brindan precisión, ahorro de tiempo y recursos, y sobretodo confiabilidad durante el proceso, sin embargo uno de los inconvenientes principales es su costo. En la actualidad para incrementar la productividad de una industria se opta por automatizar los equipos existentes, en este proyecto se solventa la necesidad de reducir el tiempo de procesado de piezas metálicas de espesor considerable y optimización de recursos.
- Una de las ventajas de un sistema que cuenta con software propio desarrollado, es la flexibilidad que éste brinda frente a las posibles modificaciones que se requieran de acuerdo a la naturaleza del proceso, es decir, netamente se depende de la forma de programación, además permite expandir el tipo de mecanizado ya que pueden hacerse procesos que abarquen más de dos ejes, dependiendo así de la habilidad del diseñador y del programador, así como también del tipo de plataformas que se requieran. En este caso el sistema mecánico es relativamente flexible ya que se contempla la posibilidad de adaptar otro tipo de herramienta.
- En un sistema CNC debe tomarse muy en cuenta la calidad del sistema mecánico, tal es el caso de la estabilidad que se obtiene dependiendo de los métodos de transmisión de movimiento y de arrastre, que netamente dependen de los componentes que se empleen tomando en cuenta el tipo de material, precisión o rectitud, como el caso de los tornillos sin fin que

por su naturaleza de construcción poseen una curvatura que influye directamente en la calidad del proceso, ya que causa vibraciones o rigidez, por tanto deben buscarse sistemas de acoplamiento que garanticen suavidad con buena transmisión de movimiento y sobretodo sin vibraciones.

- Es importante considerar en el diseño de sistemas mecánicos, que los elementos a usarse no son del todo perfectos, como es el caso de los tornillos sin fin, que a pesar de ser de acero inoxidable poseen una ligera curvatura, pudiendo de esta manera afectar al sistema. Por esta razón es importante tomar en cuenta los posibles defectos de cada elemento, para buscar una solución puntual, antes de tener un problema global.
- Al realizar un control en lazo cerrado, del tipo implementado en el presente proyecto, es necesario considerar que la velocidad de envío y recepción de datos sea la misma, para evitar pérdida de señal que daría lugar a un mal funcionamiento del equipo, lo cual repercute en el producto final.
- El desarrollo de la plataforma de control numérico permite conocer los recursos con los que cuenta MATLAB, pues es un software de ambiente científico, alto potencial y flexibilidad es por esto que se recurre a sus principales prestaciones como la manipulación de matrices, procesamiento de imágenes, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos hardware como es el caso del empleo de la DAQ6501 de la NI (National Instruments), obteniendo una plataforma relativamente reducida y muy flexible para la aplicación para la que fue creada, ya que el envío y recepción de señales desde y hacia el computador es fácil y práctico.

5.2 RECOMENDACIONES

- Como principal recomendación deben conocerse los componentes de la máquina, desde las partes que conforman su estructura, componentes mecánicos como rodamientos y guías, así como también el entorno de la interfaz de usuario, de esta manera se puede garantizar un correcto funcionamiento de todo el sistema, así como también se podrá brindar un correcto mantenimiento, sobre todo a la parte mecánica, reduciendo riesgos y evitando accidentes.
- Para aumentar la confiabilidad del pantógrafo se recomienda incorporar al sistema un UPS, de esta manera se pueden realizar las maniobras que se crea conveniente como retornar al inicio una vez apagada la antorcha o culminar el corte, que queda exclusivamente a criterio del operador, ya que por la naturaleza del proceso no se debe mantener la antorcha sobre un mismo punto por un tiempo prolongado, ya que el material a cortar sufre graves daños.
- Si se desea realizar un sistema CNC de dos ejes con una precisión menor a un milímetro, es recomendable usar motores a pasos, y realizar un correcto acoplamiento de piñones para conseguir la velocidad deseada, además es importante considerar la aplicación de la máquina, puesto que al existir rozamiento entre el material y la herramienta de corte, el torque que posea el motor pasa a considerarse un importante factor de selección.
- Considerar factores de seguridad antes del proceso de corte, tanto en el equipo, como el uso personal, como son: correcto funcionamiento del equipo de oxicorte, y vestimenta adecuada del operador.
- Con respecto a MATLAB es recomendable conocer el lenguaje de programación que utiliza, la estructura de las sentencias y sobre todo conocer las funciones que posee, ya que puede evitarse la implementación de algoritmos complejos para una misma aplicación; también es necesario mencionar que debe investigarse acerca de la compatibilidad entre este software y hardware de distinto fabricante, como es el caso de la DAQ6501

y otras de recursos más complejos como tarjetas de adquisición análogo-digitales, por tanto para utilizarlas de manera óptima es necesario conocer la estructura de las sentencias que se emplea tanto para su configuración como para su correcto funcionamiento.

REFERENCIAS BIBLIOGRÁFICAS

- [1] Wikipedia. [Online].
http://es.wikipedia.org/wiki/Control_num%C3%A9rico
- [2] (2011, Nov.) Control numérico. [Online].
http://www.profesores.frc.utn.edu.ar/industrial/sistemasinteligentes/FFlexible/EL_CONTROL_NUMERICO_COMPUTARIZADO_EN_EL_DESARROLLO_INDUSTRIAL.pdf
- [3] (2012, Feb.) Wikipedia. [Online].
http://es.wikipedia.org/wiki/M%C3%A1quina_herramienta
- [4] Monografias.com. [Online].
<http://www.monografias.com/trabajos14/maquinacontrnum/maquinacontrnum.shtml>
- [5] (2011, June) Ingeniería Técnica Industrial. [Online].
<http://isa.umh.es/asignaturas/tf/Tema11.pdf>
- [6] Documentos. [Online].
http://catarina.udlap.mx/u_dl_a/tales/documentos/lep/garcia_bs/capitulo3.pdf
- [7] (2012, Jan.) Wikipedia. [Online].
<http://es.wikipedia.org/wiki/MATLAB>
- [8] National Instruments. [Online].
<http://sine.ni.com/nips/cds/view/p/lang/en/nid/201630>
- [9] Diego Orlando Barragán. (2007, Aug.) [Online].
http://www.dspace.espol.edu.ec/bitstream/123456789/10740/19/%255Bmatlab%255D_MATLAB_GUIDE.pdf
- [10] Softonic. (2011) <http://www.softonic.com/s/convertir-jpg-dxf>.

- [11] Aprendemos Tecnología. [Online].
<http://aprendemostecnologia.org/maquinas-y-mecanismos/mecanismos-de-transformacion-del-movimiento/>
- [12] ELTRA. (2000)
<http://facultad.bayamon.inter.edu/arincon/encoderIncrementales.pdf>.

ANEXO 1

MANUAL DE USUARIO

PARTES CONSTITUTIVAS DEL PANTÓGRAFO:

- 1) Prototipo mecánico.
- 2) Caja de control.
- 3) Computador.

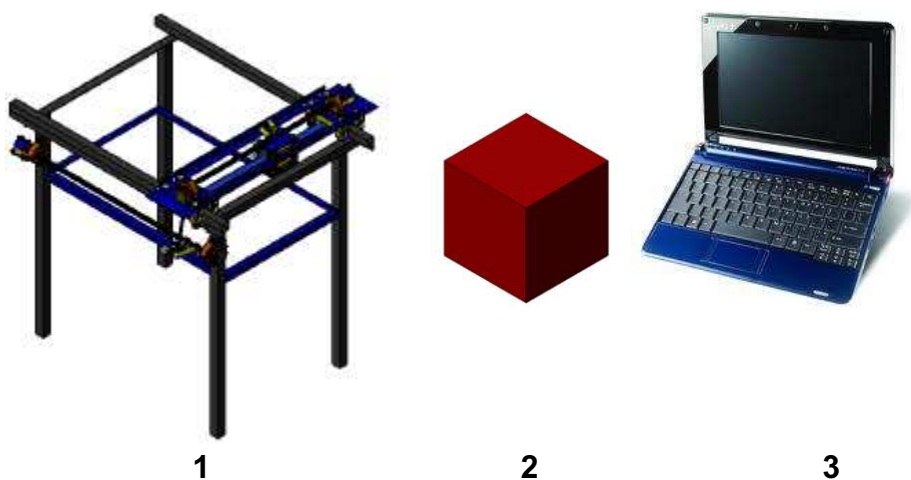


Figura A-1 Partes constitutivas del pantógrafo

1. **Prototipo mecánico:** Posee el mecanismo encargado de proporcionar movimiento al soplete de oxicorte. Como principales elementos que interactúan en el proceso están presentes:

- 2 motores 12Vdc, 5A, 12kg.
- 2 encoders incrementales con 250 pulsos por vuelta.
- 2 tornillos sin fin con paso 2.54 mm.
- Mesa que soporta elementos.

Las señales provenientes de los encoders y enviadas a los motores, llegan a una caja de paso con borneras en su interior, la cual tiene como salida dos mangueras metálicas, con terminales de conexión para potencia y control, mismos que tienen como destino una tarjeta electrónica. Para la señal de control se tiene un conector DB9 macho, para la señal de potencia un conector polarizado de 4 pines macho.



Figura A-2 Prototipo mecánico



Figura A-3 Conectores de potencia y control

2. **Caja de control:** Es la interfaz entre una computadora y el prototipo mecánico, y posee los siguientes elementos:

- Tarjeta electrónica para control.
- Tarjeta electrónica de potencia.
- Fuente 15Vdc, 8A(Encoders, Microcontroladores).
- Fuente 12Vdc, 20A(Motores DC).
- DAQ 6501, National Instruments.

La caja de control tiene los siguientes puertos:

- Puerto de comunicación con computador (Entrada a DAQ 6501).
- Puerto de conexión para potencia (Conector polarizado hembra).
- Puerto de conexión para control (Conector DB9 hembra).

Además se tiene un switch de encendido general, y un cable de poder para alimentación de 120Vac.

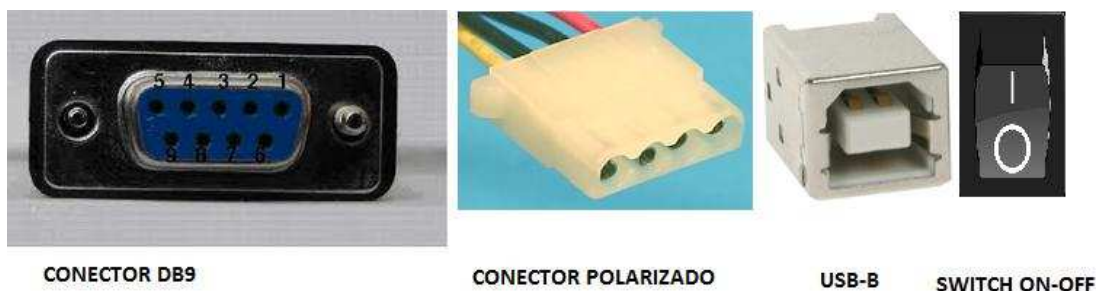


Figura A-4 Puertos presentes en la caja de control

3. **Computador:** Se recomienda que el computador a usar posea las siguientes características mínimas:

- 2 GB memoria RAM.
- 2.93 GHz Procesador.
- 4 GB de espacio disponible en disco duro.
- Sistema operativo no inferior a Windows XP.
- Un puerto USB.

PASOS PARA PUESTA EN MARCHA DEL PANTÓGRAFO:

1. **Conectar los siguientes cables:**

- Cable desde computador (USB), hasta la caja de control (entrada DAQ 6501).



Figura A-5 Cable USB-A, USB-B

- Cable desde prototipo mecánico (DB9 macho), hasta caja de control (DB9 hembra)



Figura A-6 Cable DB9, conexión a caja de control

- Cable desde prototipo mecánico (conector polarizado macho), hasta caja de control (conector polarizado hembra)



Figura A 7 Conector polarizado, conexión a caja de control

- Cable de potencia a 120Vac, 60Hz.



Figura A-8 Cable de alimentación 120Vac, 60Hz

2. Encendido:

- Habilitar el switch de encendido general presente en la caja de control.
- Posicionar el origen del corte en el punto (0,0), con la ayuda de los botones de posicionamiento situados en el interior de la caja de control.

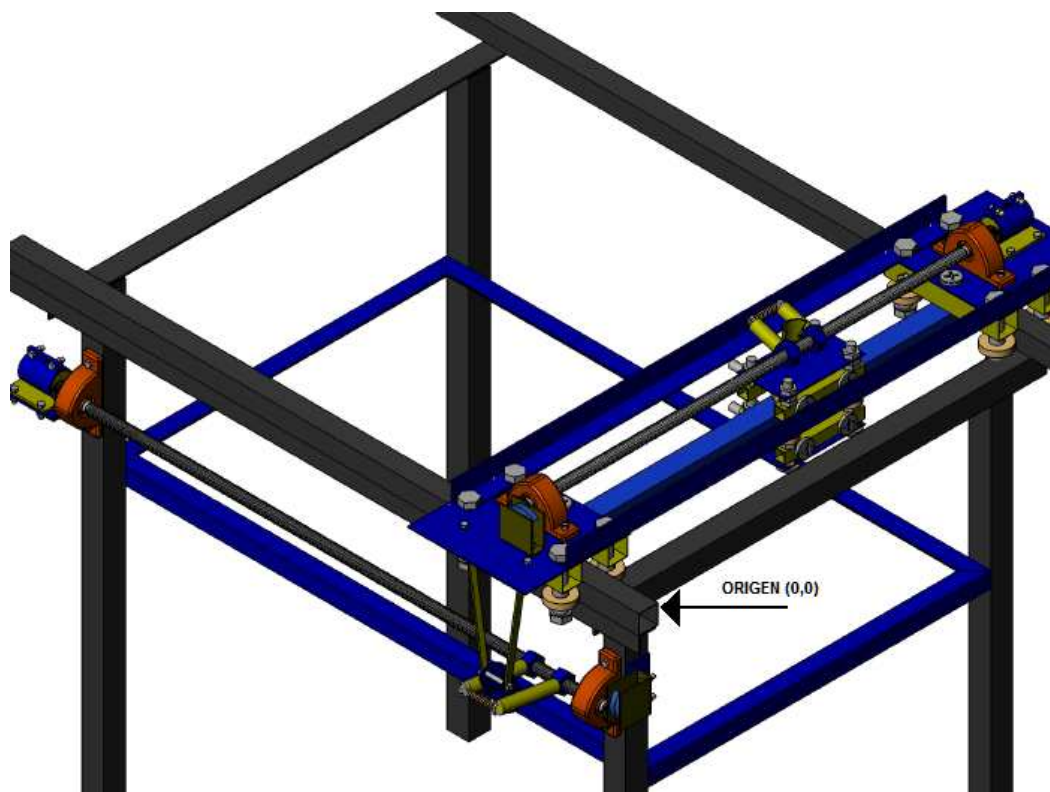


Figura A-9 Origen del corte

- Encender un computador, con el software para control del pantógrafo previamente instalado.
- Ejecutar el software.

3. Manejo de programa:

- Ejecutar el archivo .exe "PANTÓGRAFO", se despliega la pantalla principal.
- Presionar el botón "IMPORTAR IMAGEN", buscar la imagen prediseñada a cortar, y abrir la misma (Aparecerá la imagen importada y la imagen procesada en la pantalla).

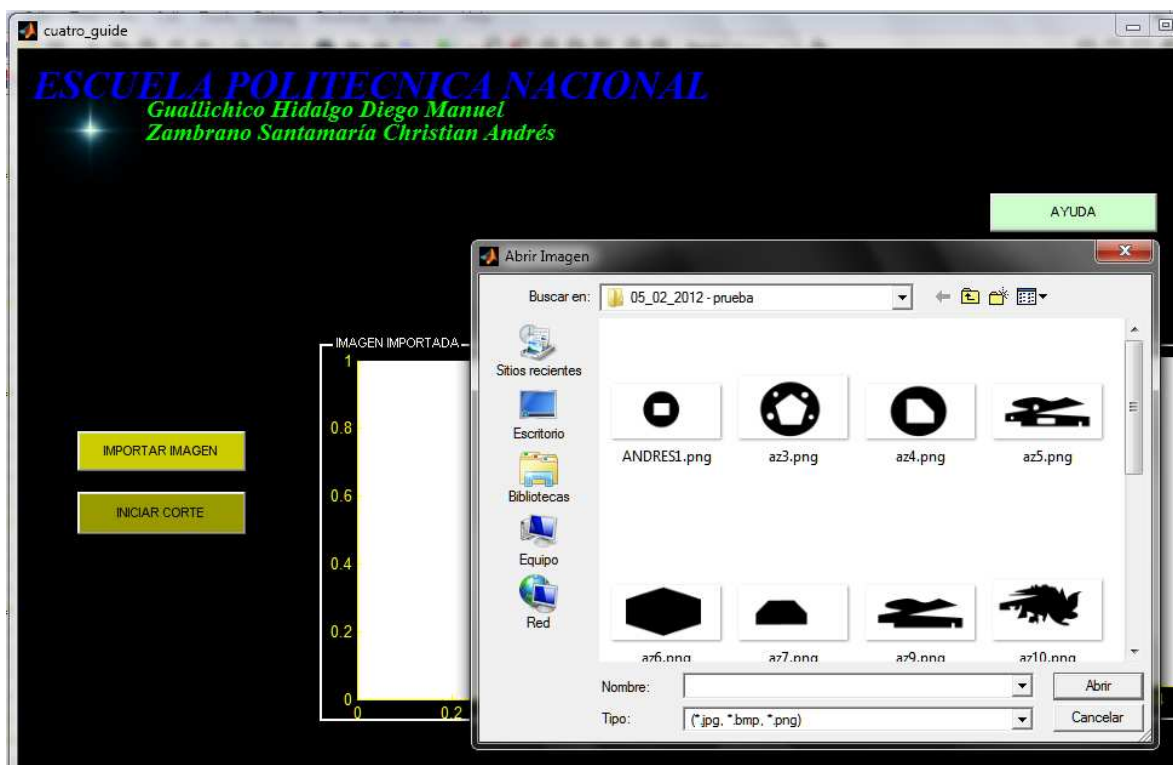


Figura A-10 Importación de imagen (Pantalla Principal)

- Presionar el botón “INICIAR CORTE”, entonces se desplegará un anuncio indicando si desea iniciar el corte, al aceptar se despliega una pantalla donde se indica de color azul las líneas de posicionamiento y de color rojo los contornos a cortar, luego inicia el proceso, caso contrario es posible importar una nueva imagen.

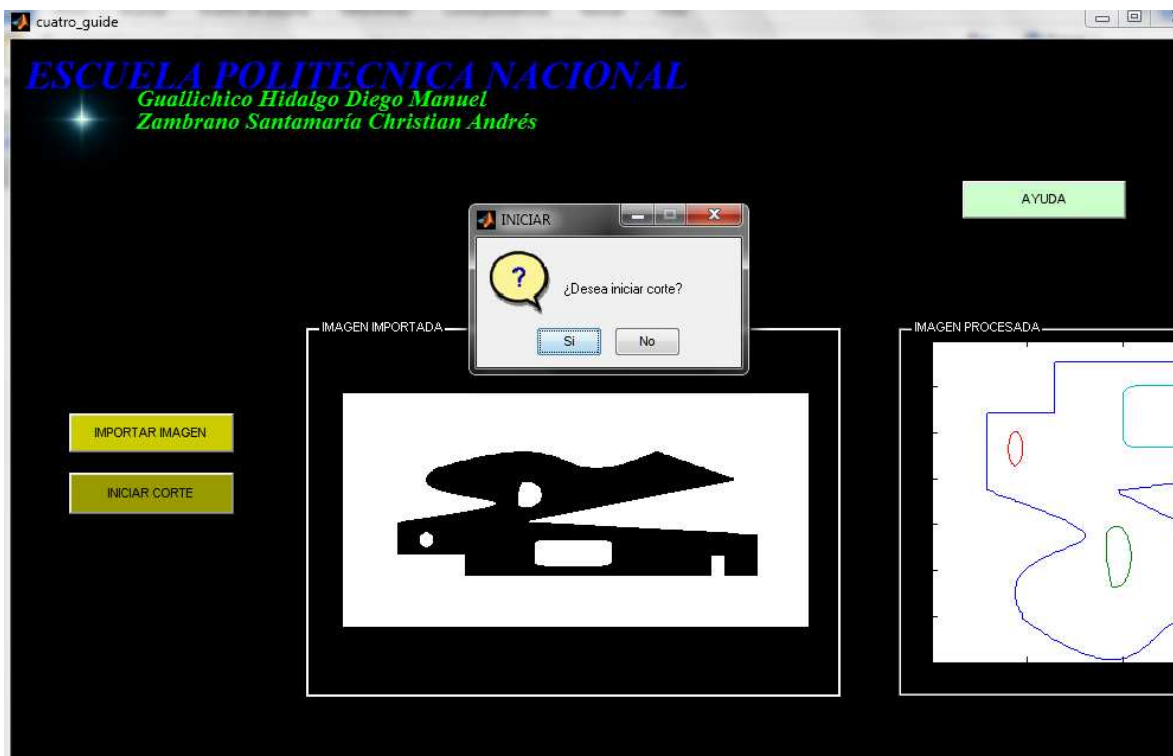


Figura A-11 Inicio de corte

- Después de haber aceptado iniciar el proceso, la máquina sigue los siguientes pasos:
 - Se posiciona el soplete en el origen del corte de la figura, en este punto se detiene esperando una señal por parte del usuario para continuar, el mismo que debe encender⁴ el soplete de la oxicorte y esperar que se caliente el material.

⁴ Indica encender con fuego la mezcla de oxígeno y propano, para generar la llama de corte.

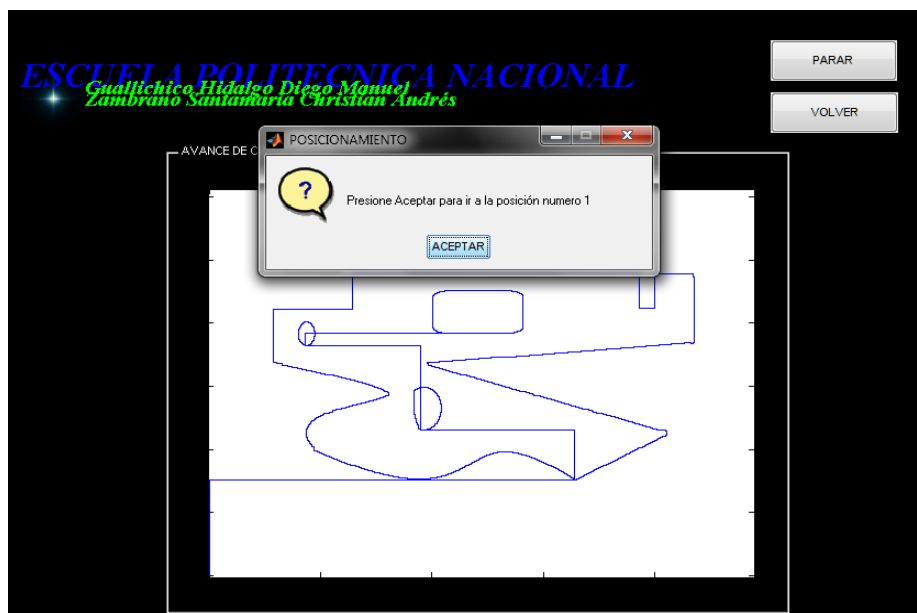


Figura A-12 Inicio de posicionamiento

- Una vez que se cumpla con los requisitos mencionados, el usuario debe presionar “ACEPTAR” para iniciar el corte del contorno de la figura.

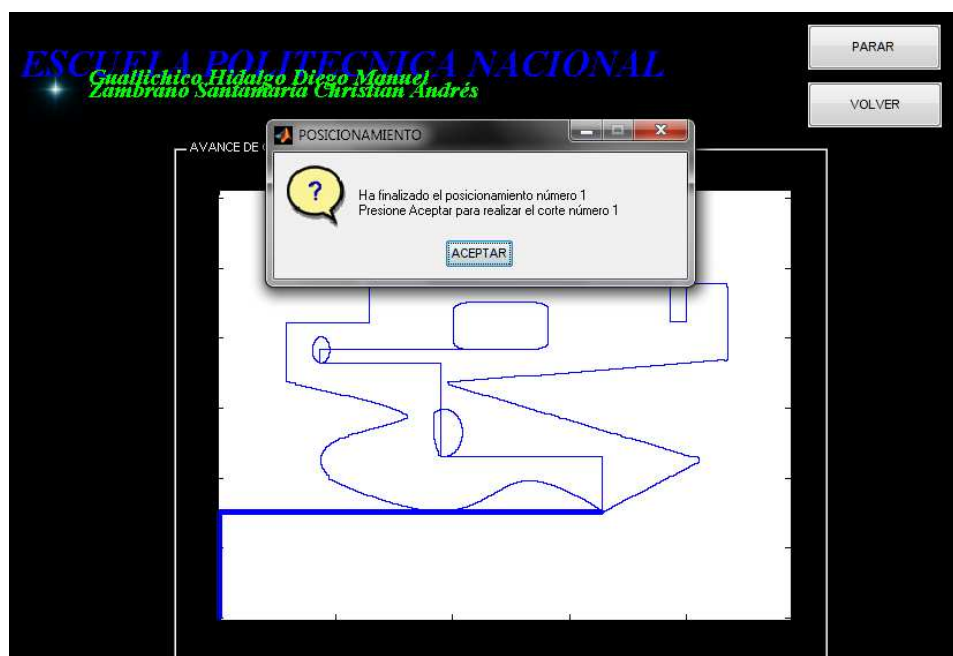


Figura A-13 Inicio de corte del primer contorno

- Al terminar de cortar el contorno, el soplete se detiene, y el usuario debe apagar el mismo.

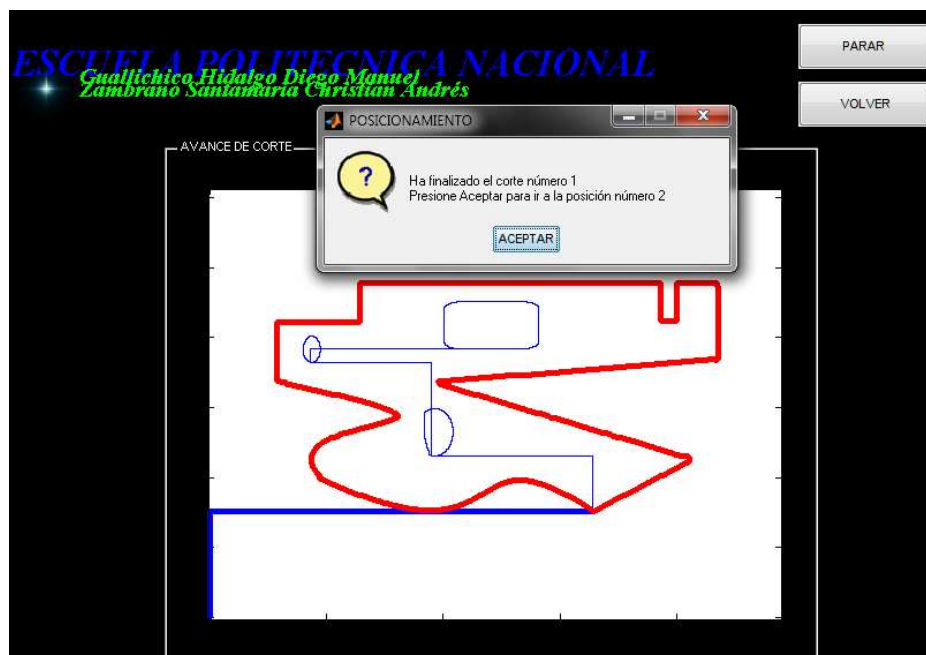


Figura A-14 Finalización del primer corte en inicio del siguiente posicionamiento en caso de existir contornos internos.

- Si la figura tiene contornos internos el proceso se repite hasta terminar de cortar todos los contornos, luego se presiona “ACEPTAR” para retornar al punto de origen.



Figura A-15 Retorno a punto inicial

- Se presiona el botón “VOLVER” para regresar a la pantalla principal, donde se puede realizar otro corte si es requerido.

Adicionalmente se tiene en la pantalla principal un botón de ayuda, el cual permite al usuario aprender a interactuar con el programa.

En la pantalla de corte se tiene un botón de “PARO”, el cual permite detener el movimiento del soplete oxicorte en cualquier punto del proceso.

ANEXO 2

HOJAS DE DATOS DE LOS ELEMENTOS

MOTOROLA
SEMICONDUCTOR TECHNICAL DATA

 Order this document
 by 4N35/D

6-Pin DIP Optoisolators
Transistor Output

The 4N35, 4N36 and 4N37 devices consist of a gallium arsenide infrared emitting diode optically coupled to a monolithic silicon phototransistor detector.

- Current Transfer Ratio — 100% Minimum @ Specified Conditions
- Guaranteed Switching Speeds
- Meets or Exceeds all JEDEC Registered Specifications
- *To order devices that are tested and marked per VDE 0884 requirements, the suffix "V" must be included at end of part number. VDE 0884 is a test option.*

Applications

- General Purpose Switching Circuits
- Interfacing and coupling systems of different potentials and impedances
- Regulation Feedback Circuits
- Monitor & Detection Circuits
- Solid State Relays

MAXIMUM RATINGS ($T_A = 25^\circ\text{C}$ unless otherwise noted)

Rating	Symbol	Value	Unit
INPUT LED			
Reverse Voltage	V_R	6	Volts
Forward Current — Continuous	I_F	60	mA
LED Power Dissipation @ $T_A = 25^\circ\text{C}$ with Negligible Power in Output Detector Derate above 25°C	P_D	120	mW
		1.41	mW/ $^\circ\text{C}$

OUTPUT TRANSISTOR

Collector-Emitter Voltage	V_{CEO}	30	Volts
Emitter-Base Voltage	V_{EBO}	7	Volts
Collector-Base Voltage	V_{CBO}	70	Volts
Collector Current — Continuous	I_C	150	mA
Detector Power Dissipation @ $T_A = 25^\circ\text{C}$ with Negligible Power in Input LED Derate above 25°C	P_D	150	mW
		1.76	mW/ $^\circ\text{C}$

TOTAL DEVICE

Isolation Source Voltage ⁽¹⁾ (Peak ac Voltage, 60 Hz, 1 sec Duration)	V_{ISO}	7500	Vac(pk)
Total Device Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	250	mW
		2.94	mW/ $^\circ\text{C}$
Ambient Operating Temperature Range ⁽²⁾	T_A	-55 to +100	$^\circ\text{C}$
Storage Temperature Range ⁽²⁾	T_{stg}	-55 to +150	$^\circ\text{C}$
Soldering Temperature (10 sec, 1/16" from case)	T_L	260	$^\circ\text{C}$

1. Isolation surge voltage is an internal device dielectric breakdown rating.
For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.
2. Refer to Quality and Reliability Section in Opto Data Book for information on test conditions.
Preferred devices are Motorola recommended choices for future use and best overall value.

GlobalOptoisolator is a trademark of Motorola, Inc.

REV 2

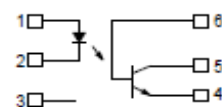
© Motorola, Inc. 1995

4N35*
4N36
4N37

[CTR = 100% Min]

*Motorola Preferred Device

STYLE 1 PLASTIC

STANDARD THRU HOLE
CASE 730A-04
SCHEMATIC

 PIN 1. LED ANODE
 2. LED CATHODE
 3. N.C.
 4. EMITTER
 5. COLLECTOR
 6. BASE

MOTOROLA

4N35 4N36 4N37ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise noted)⁽¹⁾

Characteristic	Symbol	Min	Typ ⁽¹⁾	Max	Unit	
INPUT LED						
Forward Voltage ($I_F = 10\text{ mA}$)	$T_A = 25^\circ\text{C}$ $T_A = -55^\circ\text{C}$ $T_A = 100^\circ\text{C}$	V_F	0.8 0.9 0.7	1.15 1.3 1.05	1.5 1.7 1.4	V
Reverse Leakage Current ($V_R = 6\text{ V}$)		I_R	—	—	10	μA
Capacitance ($V = 0\text{ V}$, $f = 1\text{ MHz}$)		C_J	—	18	—	pF
OUTPUT TRANSISTOR						
Collector–Emitter Dark Current ($V_{CE} = 10\text{ V}$, $T_A = 25^\circ\text{C}$) ($V_{CE} = 30\text{ V}$, $T_A = 100^\circ\text{C}$)		I_{CEO}	— —	1 —	50 500	nA μA
Collector–Base Dark Current ($V_{CB} = 10\text{ V}$)	$T_A = 25^\circ\text{C}$ $T_A = 100^\circ\text{C}$	I_{CBO}	—	0.2 100	20 —	nA
Collector–Emitter Breakdown Voltage ($I_C = 1\text{ mA}$)		$V_{(BR)CEO}$	30	45	—	V
Collector–Base Breakdown Voltage ($I_C = 100\text{ }\mu\text{A}$)		$V_{(BR)CBO}$	70	100	—	V
Emitter–Base Breakdown Voltage ($I_E = 100\text{ }\mu\text{A}$)		$V_{(BR)EBO}$	7	7.8	—	V
DC Current Gain ($I_C = 2\text{ mA}$, $V_{CE} = 5\text{ V}$)		h_{FE}	—	400	—	—
Collector–Emitter Capacitance ($f = 1\text{ MHz}$, $V_{CE} = 0$)		C_{CE}	—	7	—	pF
Collector–Base Capacitance ($f = 1\text{ MHz}$, $V_{CB} = 0$)		C_{CB}	—	19	—	pF
Emitter–Base Capacitance ($f = 1\text{ MHz}$, $V_{EB} = 0$)		C_{EB}	—	9	—	pF
COUPLED						
Output Collector Current ($I_F = 10\text{ mA}$, $V_{CE} = 10\text{ V}$)	$T_A = 25^\circ\text{C}$ $T_A = -55^\circ\text{C}$ $T_A = 100^\circ\text{C}$	I_C (CTR) ⁽²⁾	10 (100) 4 (40) 4 (40)	30 (300) — —	— — —	mA (%)
Collector–Emitter Saturation Voltage ($I_C = 0.5\text{ mA}$, $I_F = 10\text{ mA}$)		$V_{CE(sat)}$	—	0.14	0.3	V
Turn–On Time	$(I_C = 2\text{ mA}$, $V_{CC} = 10\text{ V}$, $R_L = 100\text{ }\Omega$) ⁽³⁾	t_{on}	—	7.5	10	μs
Turn–Off Time		t_{off}	—	5.7	10	
Rise Time		t_r	—	3.2	—	
Fall Time		t_f	—	4.7	—	
Isolation Voltage ($f = 60\text{ Hz}$, $t = 1\text{ sec}$)		V_{ISO}	7500	—	—	Vac(pk)
Isolation Current ⁽⁴⁾ ($V_{I-O} = 3550\text{ Vpk}$)	4N35	I_{ISO}	—	—	100	μA
($V_{I-O} = 2500\text{ Vpk}$)	4N36	—	—	—	100	
($V_{I-O} = 1500\text{ Vpk}$)	4N37	—	—	8	100	
Isolation Resistance ($V = 500\text{ V}$) ⁽⁴⁾		R_{ISO}	10^{11}	—	—	Ω
Isolation Capacitance ($V = 0\text{ V}$, $f = 1\text{ MHz}$) ⁽⁴⁾		C_{ISO}	—	0.2	2	pF

1. Always design to the specified minimum/maximum electrical limits (where applicable).

2. Current Transfer Ratio (CTR) = $I_C/I_F \times 100\%$.

3. For test circuit setup and waveforms, refer to Figure 11.

4. For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.

4N35 4N36 4N37

TYPICAL CHARACTERISTICS

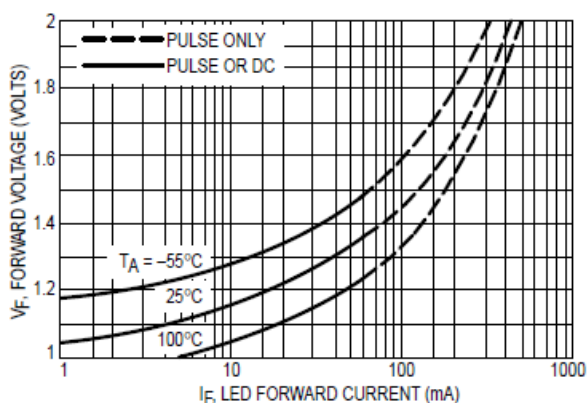


Figure 1. LED Forward Voltage versus Forward Current

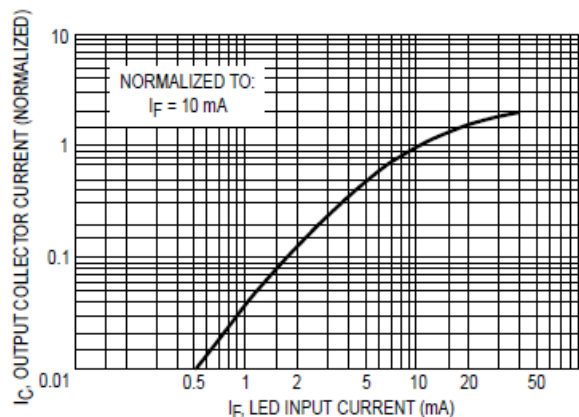


Figure 2. Output Current versus Input Current

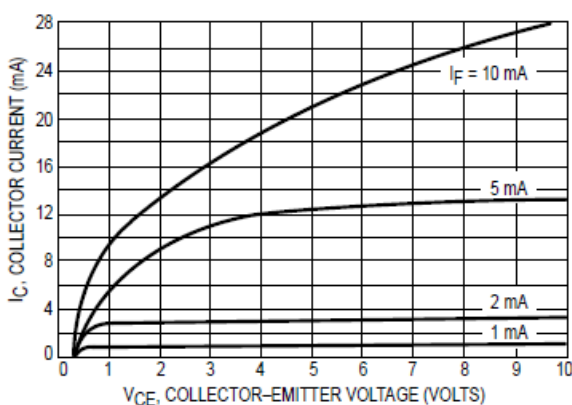


Figure 3. Collector Current versus Collector-Emitter Voltage

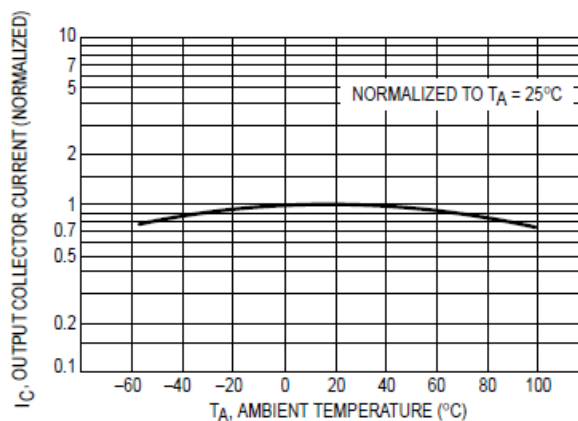


Figure 4. Output Current versus Ambient Temperature

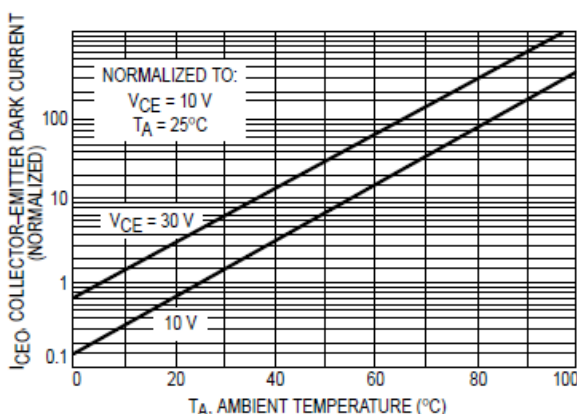


Figure 5. Dark Current versus Ambient Temperature

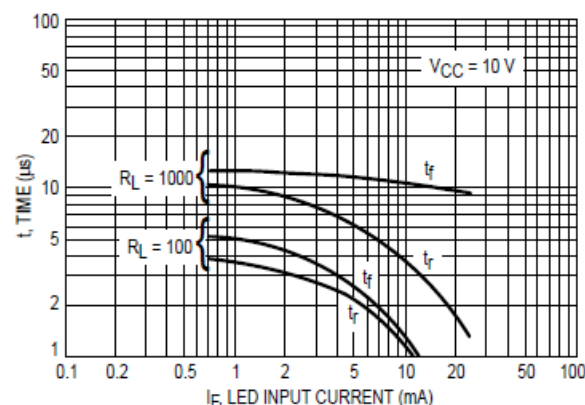


Figure 6. Rise and Fall Times (Typical Values)

4N35 4N36 4N37

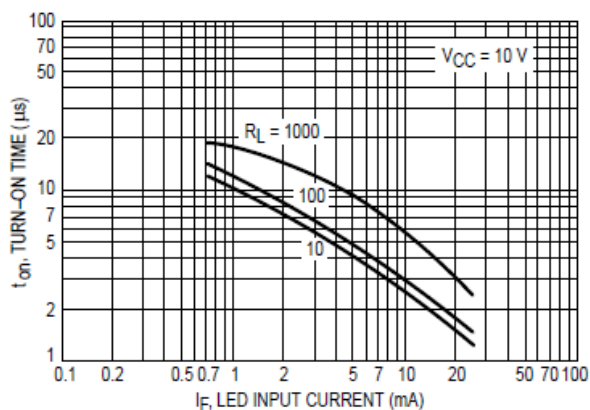


Figure 7. Turn-On Switching Times

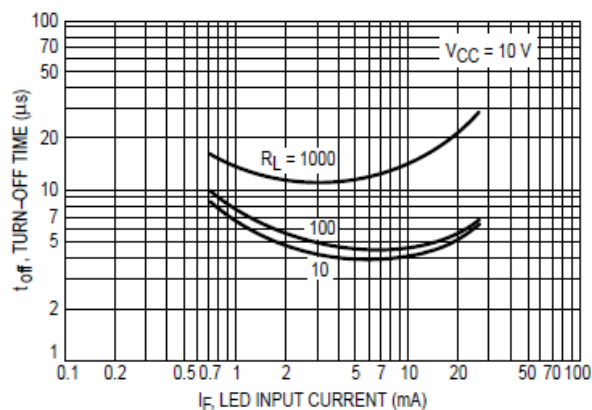


Figure 8. Turn-Off Switching Times

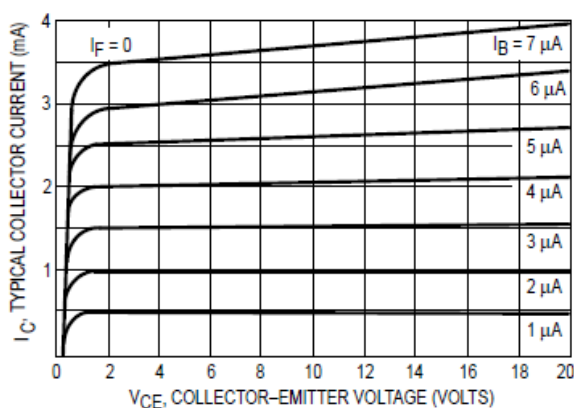


Figure 9. DC Current Gain (Detector Only)

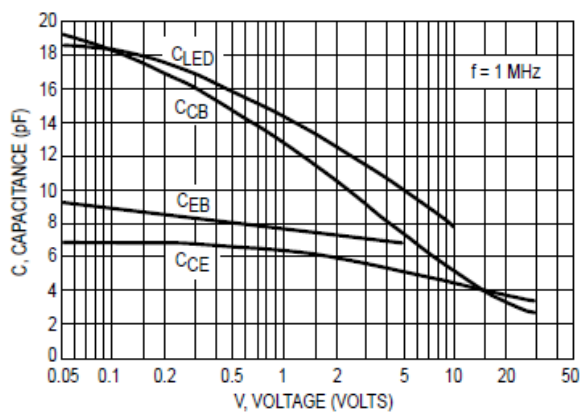


Figure 10. Capacitances versus Voltage

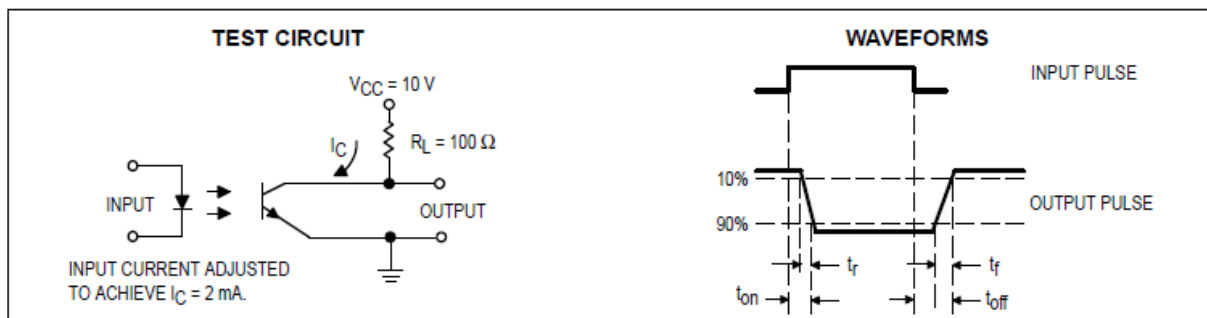
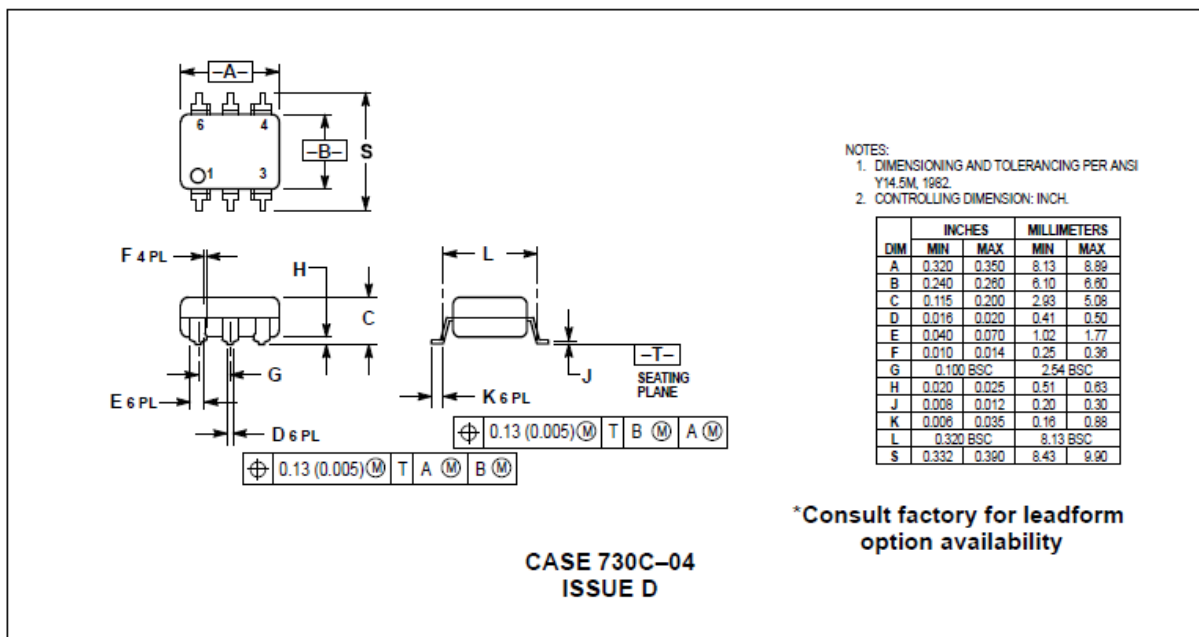
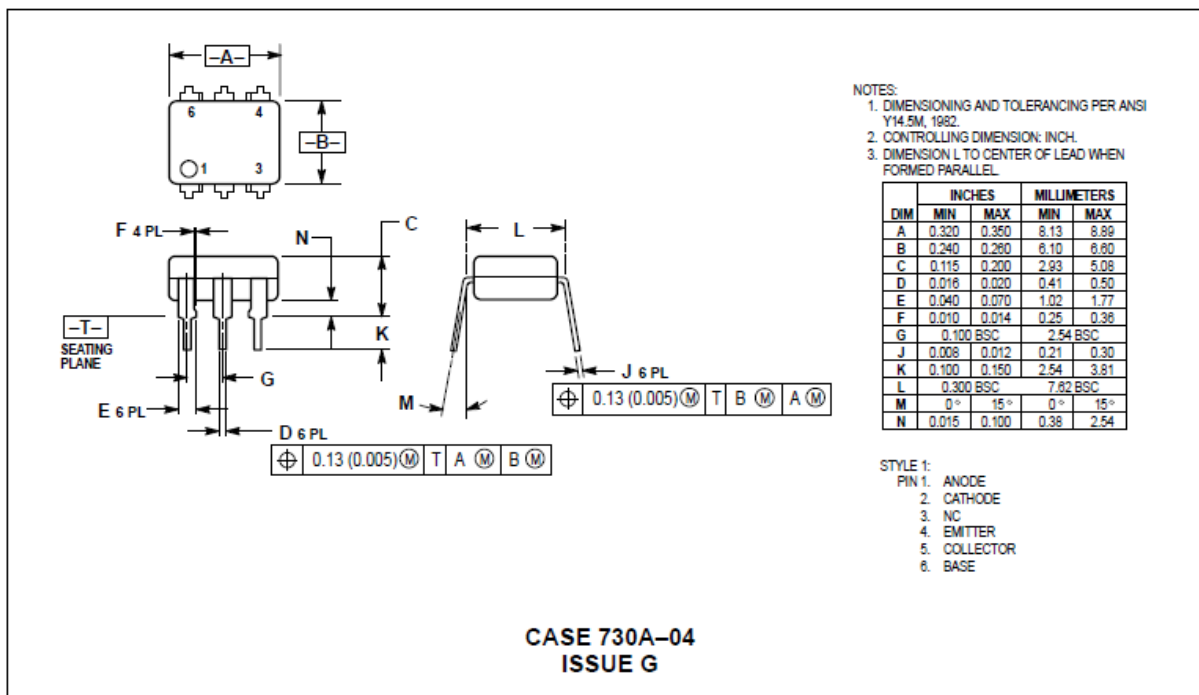


Figure 11. Switching Time Test Circuit and Waveforms

4N35 4N36 4N37

PACKAGE DIMENSIONS



4N35 4N36 4N37

NOTES:
 1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
 2. CONTROLLING DIMENSION: INCH.
 3. DIMENSION L TO CENTER OF LEAD WHEN FORMED PARALLEL.

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.320	0.350	8.13	8.89
B	0.240	0.280	6.10	6.80
C	0.115	0.200	2.93	5.08
D	0.016	0.020	0.41	0.50
E	0.040	0.070	1.02	1.77
F	0.010	0.014	0.25	0.36
G	0.100 BSC		2.54 BSC	
J	0.008	0.012	0.21	0.30
K	0.100	0.150	2.54	3.81
L	0.400	0.425	10.16	10.80
N	0.015	0.040	0.38	1.02

***Consult factory for leadform option availability**

**CASE 730D-05
ISSUE D**

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

How to reach us:
 USA / EUROPE: Motorola Literature Distribution;
 P.O. Box 20912; Phoenix, Arizona 85036. 1-800-441-2447

JAPAN: Nippon Motorola Ltd.; Tatsumi-SPD-JLDC, Toshikatsu Otsuki,
 6F Seiko-Butsuryu-Center, 3-14-2 Tatsumi Koto-Ku, Tokyo 135, Japan. 03-3521-8315

MFAX: RMFAX0@email.sps.mot.com - TOUCHSTONE (602) 244-6609
 INTERNET: http://Design-NET.com

HONG KONG: Motorola Semiconductors H.K. Ltd.; 6B Tai Ping Industrial Park,
 51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852-26629298



Features

- Utilizes the AVR[®] RISC Architecture
- AVR – High-performance and Low-power RISC Architecture
 - 120 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 20 MIPS Throughput at 20 MHz
- Data and Non-volatile Program and Data Memories
 - 2K Bytes of In-System Self Programmable Flash
Endurance 10,000 Write/Erase Cycles
 - 128 Bytes In-System Programmable EEPROM
Endurance: 100,000 Write/Erase Cycles
 - 128 Bytes Internal SRAM
 - Programming Lock for Flash Program and EEPROM Data Security
- Peripheral Features
 - One 8-bit Timer/Counter with Separate Prescaler and Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare and Capture Modes
 - Four PWM Channels
 - On-chip Analog Comparator
 - Programmable Watchdog Timer with On-chip Oscillator
 - USI – Universal Serial Interface
 - Full Duplex USART
- Special Microcontroller Features
 - debugWIRE On-chip Debugging
 - In-System Programmable via SPI Port
 - External and Internal Interrupt Sources
 - Low-power Idle, Power-down, and Standby Modes
 - Enhanced Power-on Reset Circuit
 - Programmable Brown-out Detection Circuit
 - Internal Calibrated Oscillator
- I/O and Packages
 - 18 Programmable I/O Lines
 - 20-pin PDIP, 20-pin SOIC, 20-pad QFN/MLF
- Operating Voltages
 - 1.8 – 5.5V (ATtiny2313V)
 - 2.7 – 5.5V (ATtiny2313)
- Speed Grades
 - ATtiny2313V: 0 – 4 MHz @ 1.8 - 5.5V, 0 – 10 MHz @ 2.7 – 5.5V
 - ATtiny2313: 0 – 10 MHz @ 2.7 - 5.5V, 0 – 20 MHz @ 4.5 – 5.5V
- Typical Power Consumption
 - Active Mode
 - 1 MHz, 1.8V: 230 μ A
 - 32 kHz, 1.8V: 20 μ A (including oscillator)
 - Power-down Mode
 - < 0.1 μ A at 1.8V



**8-bit AVR[®]
Microcontroller
with 2K Bytes
In-System
Programmable
Flash**

ATtiny2313/V

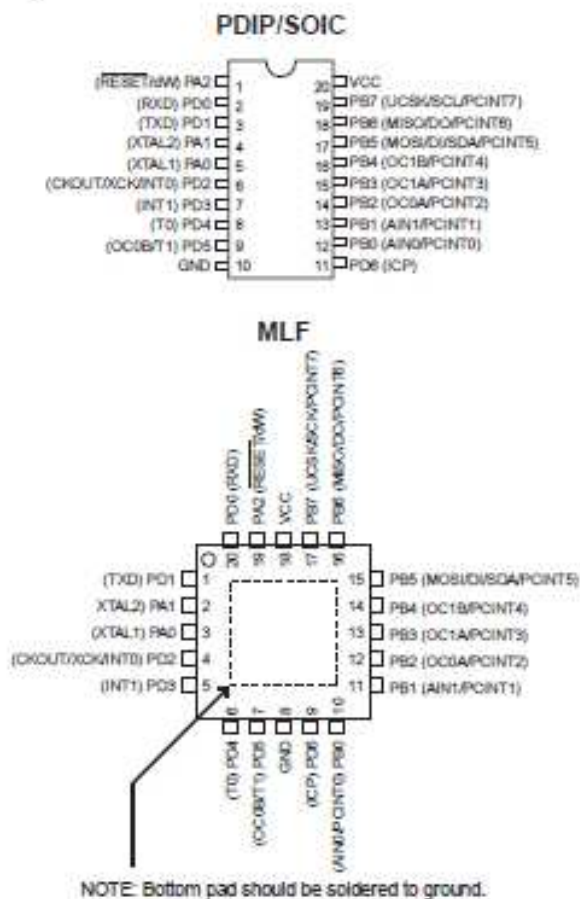
Preliminary





Pin Configurations

Figure 1. Pinout ATtiny2313

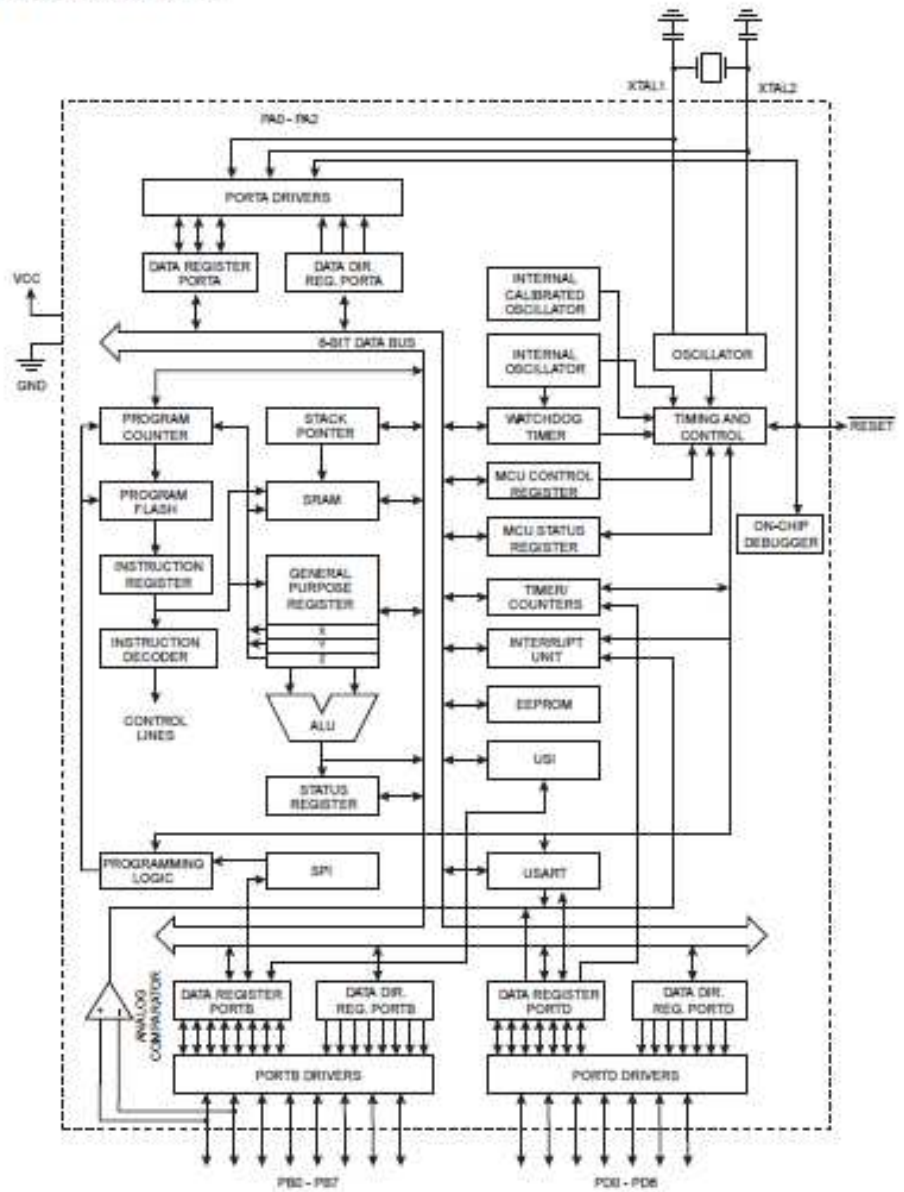


Overview

The ATtiny2313 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATtiny2313 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 2. Block Diagram



Features

- High Performance, Low Power AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
 - 120 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
- Non-volatile Program and Data Memories
 - 2/4/8K Bytes of In-System Programmable Program Memory Flash
 - Endurance: 10,000 Write/Erase Cycles
 - 128/256/512 Bytes In-System Programmable EEPROM
 - Endurance: 100,000 Write/Erase Cycles
 - 128/256/512 Bytes Internal SRAM
 - Programming Lock for Self-Programming Flash Program and EEPROM Data Security
- Peripheral Features
 - 8-bit Timer/Counter with Prescaler and Two PWM Channels
 - 8-bit High Speed Timer/Counter with Separate Prescaler
 - 2 High Frequency PWM Outputs with Separate Output Compare Registers
 - Programmable Dead Time Generator
 - USI – Universal Serial Interface with Start Condition Detector
 - 10-bit ADC
 - 4 Single Ended Channels
 - 2 Differential ADC Channel Pairs with Programmable Gain (1x, 20x)
 - Temperature Measurement
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - debugWIRE On-chip Debug System
 - In-System Programmable via SPI Port
 - External and Internal Interrupt Sources
 - Low Power Idle, ADC Noise Reduction, and Power-down Modes
 - Enhanced Power-on Reset Circuit
 - Programmable Brown-out Detection Circuit
 - Internal Calibrated Oscillator
- I/O and Packages
 - Six Programmable I/O Lines
 - 8-pin PDIP, 8-pin SOIC, 20-pad QFN/MLF, and 8-pin TSSOP (only ATtiny45/V)
- Operating Voltage
 - 1.8 - 5.5V for ATtiny25V/45V/85V
 - 2.7 - 5.5V for ATtiny25/45/85
- Speed Grade
 - ATtiny25V/45V/85V: 0 – 4 MHz @ 1.8 - 5.5V, 0 - 10 MHz @ 2.7 - 5.5V
 - ATtiny25/45/85: 0 – 10 MHz @ 2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V
- Industrial Temperature Range
- Low Power Consumption
 - Active Mode:
 - 1 MHz, 1.8V: 300 µA
 - Power-down Mode:
 - 0.1 µA at 1.8V



**8-bit AVR®
Microcontroller
with 2/4/8K
Bytes In-System
Programmable
Flash**

**ATtiny25/V
ATtiny45/V
ATtiny85/V**

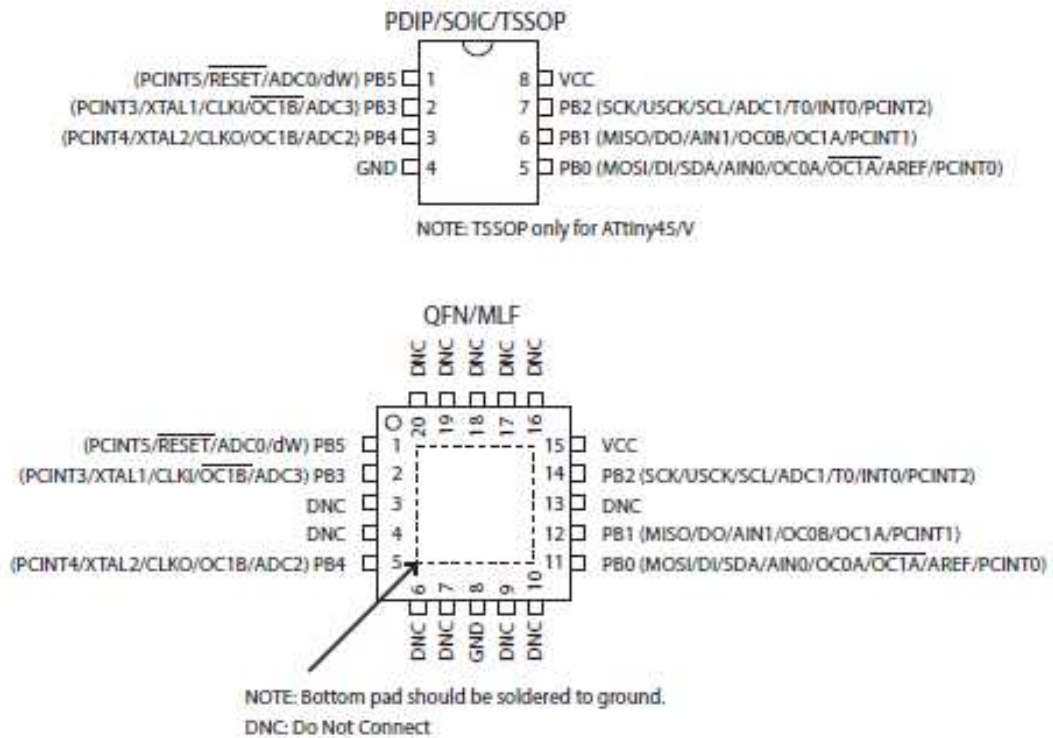
Rev. 2586N-AVR-04/11





1. Pin Configurations

Figure 1-1. Pinout ATtiny25/45/85



1.1 Pin Descriptions

1.1.1 VCC

Supply voltage.

1.1.2 GND

Ground.

1.1.3 Port B (PB5:PB0)

Port B is a 6-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up

2 ATtiny25/45/85



Technical Sales
Ecuador
512-683-0100
orders@ni.com

NI USB-6501 (24-ch, 8.5 mA)

Low-Cost USB Digital I/O Device

- 24 digital I/O lines, one 32-bit counter
- Over-voltage protection, 8.5 mA current drive
- Built-in screw terminals or generic 34-pin connector (OEM version) for easy integration
- USB 2.0 full-speed (12 Mb/s) bus interface
- Bus-powered design for convenience
- OEM version available, part number 192317-50, please call for pricing



Overview

The National Instruments USB-6501 is a portable digital I/O device, providing reliable data acquisition and control at a low price. With plug-and-play USB connectivity, the NI USB-6501 is simple enough for home / academic applications, but robust and versatile enough for laboratory / industrial applications. A board only USB 6501 OEM version is also available. The OEM version is ideal for embedded applications. Please contact NI for pricing information about the USB-6501 OEM version, part number: 192317-50.

Specifications

Specifications Documents

- Specifications (3)
- Data Sheet

Specifications Summary

General

Product Name	USB-6501
Product Family	Digital I/O
Form Factor	USB
Part Number	779205-01
Operating System/Target	Linux , Mac OS , Pocket PC , Windows
Measurement Type	Digital
RoHS Compliant	Yes

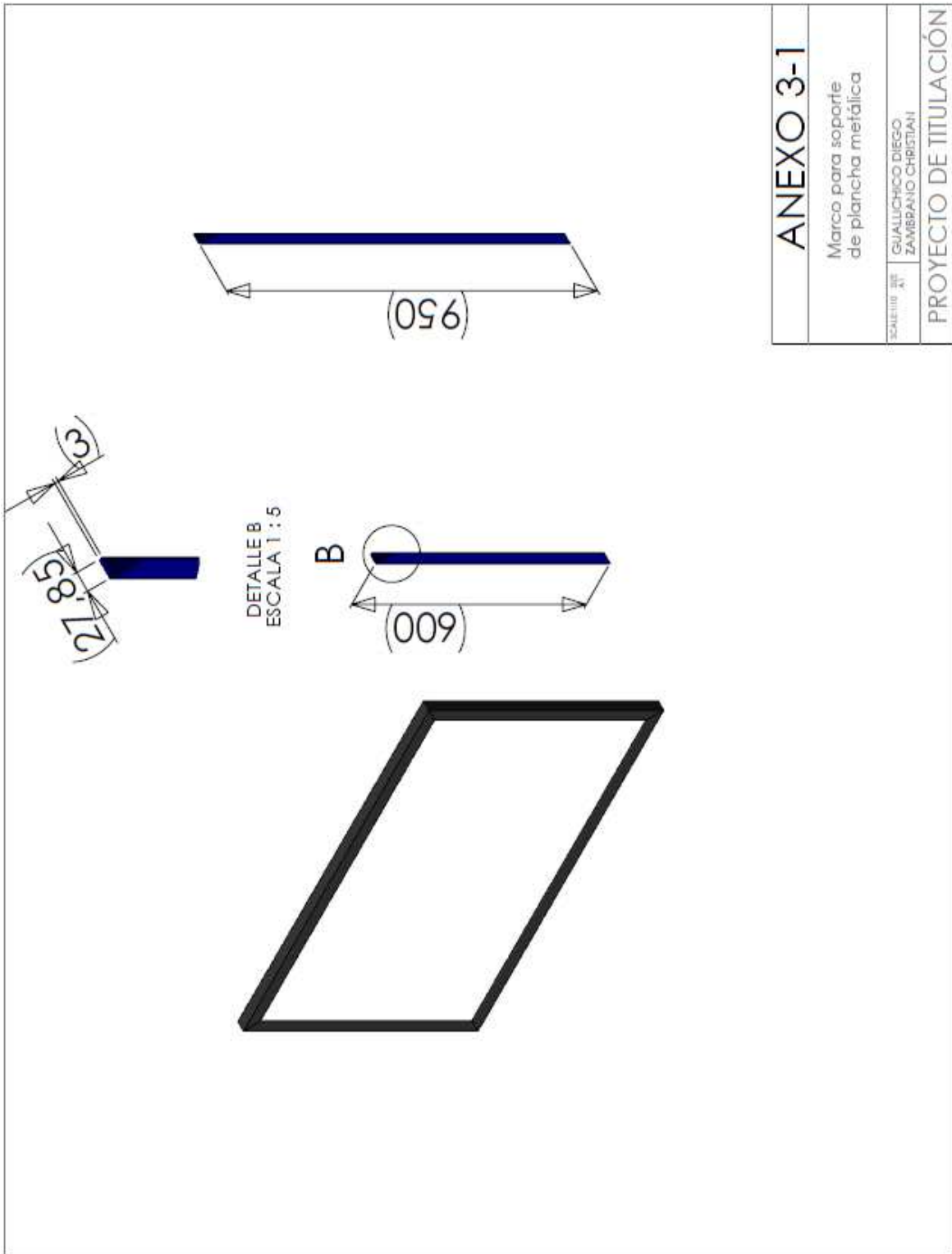
Analog Input

Channels	0
Single-Ended Channels	0
Differential Channels	0
Analog Output	
Channels	0
Digital I/O	
Bidirectional Channels	24
Input-Only Channels	0
Output-Only Channels	0
Number of Channels	24 , 0
Timing	Software
Logic Levels	TTL
Input Current Flow	Sinking , Sourcing
Output Current Flow	Sinking , Sourcing
Programmable Input Filters	No
Supports Programmable Power-Up State?	No
Current Drive Single	8.5 mA
Current Drive All	65 mA
Watchdog Timer	No
Supports Handshaking I/O?	No
Supports Pattern I/O?	No
Maximum Input Range	0 V , 5 V
Maximum Output Range	0 V , 5 V
Counter/Timers	
Counters	1
Number of DMA Channels	0
Buffered Operations	No
Debouncing/Glitch Removal	No
GP8 Synchronization	No
Maximum Range	0 V , 5 V
Max Source Frequency	5 MHz

Minimum Input Pulse Width	100 ns
Pulse Generation	No
Resolution	32 bits
Timebase Stability	50 ppm
Logic Levels	TTL
Physical Specifications	
Length	8.51 cm
Width	8.18 cm
Height	2.31 cm
I/O Connector	Screw terminals

ANEXO 3

DIAGRAMAS MECÁNICOS

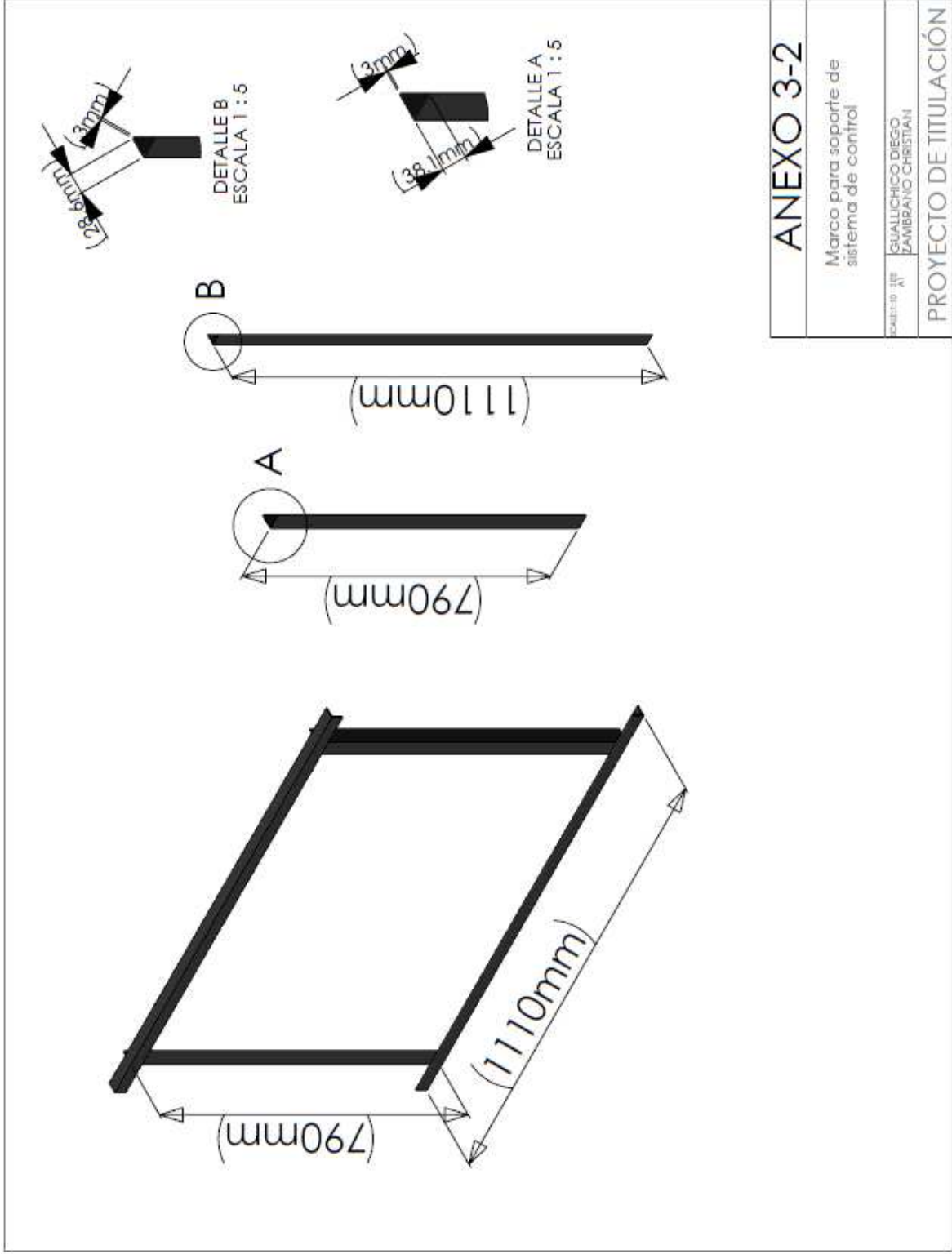


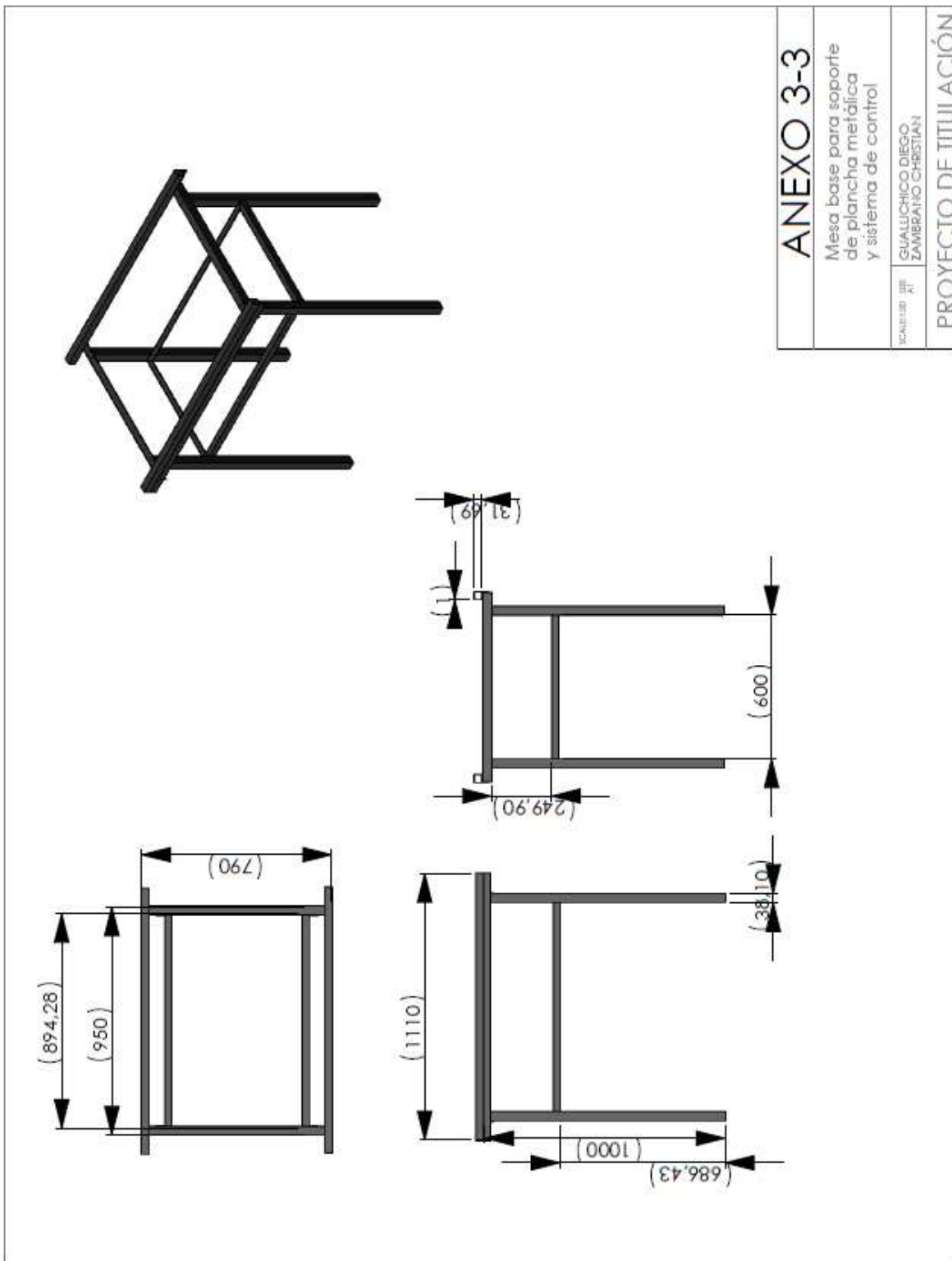
ANEXO 3-1

Marco para soporte
de plancha metálica

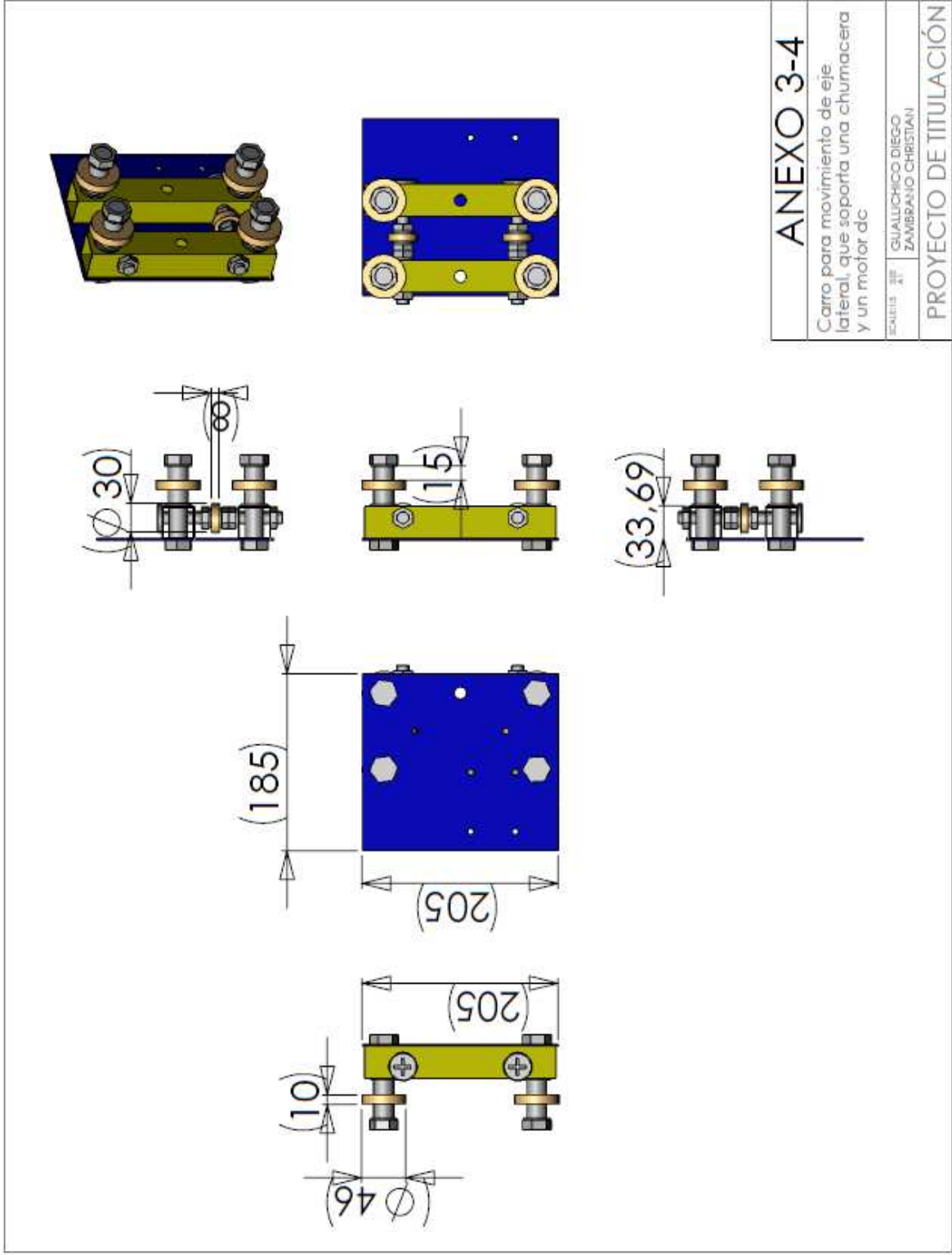
ESCUELA Nº 2001
GUALICHICO DIEGO
ZAMBRANO CHRISTIAN

PROYECTO DE TITULACIÓN

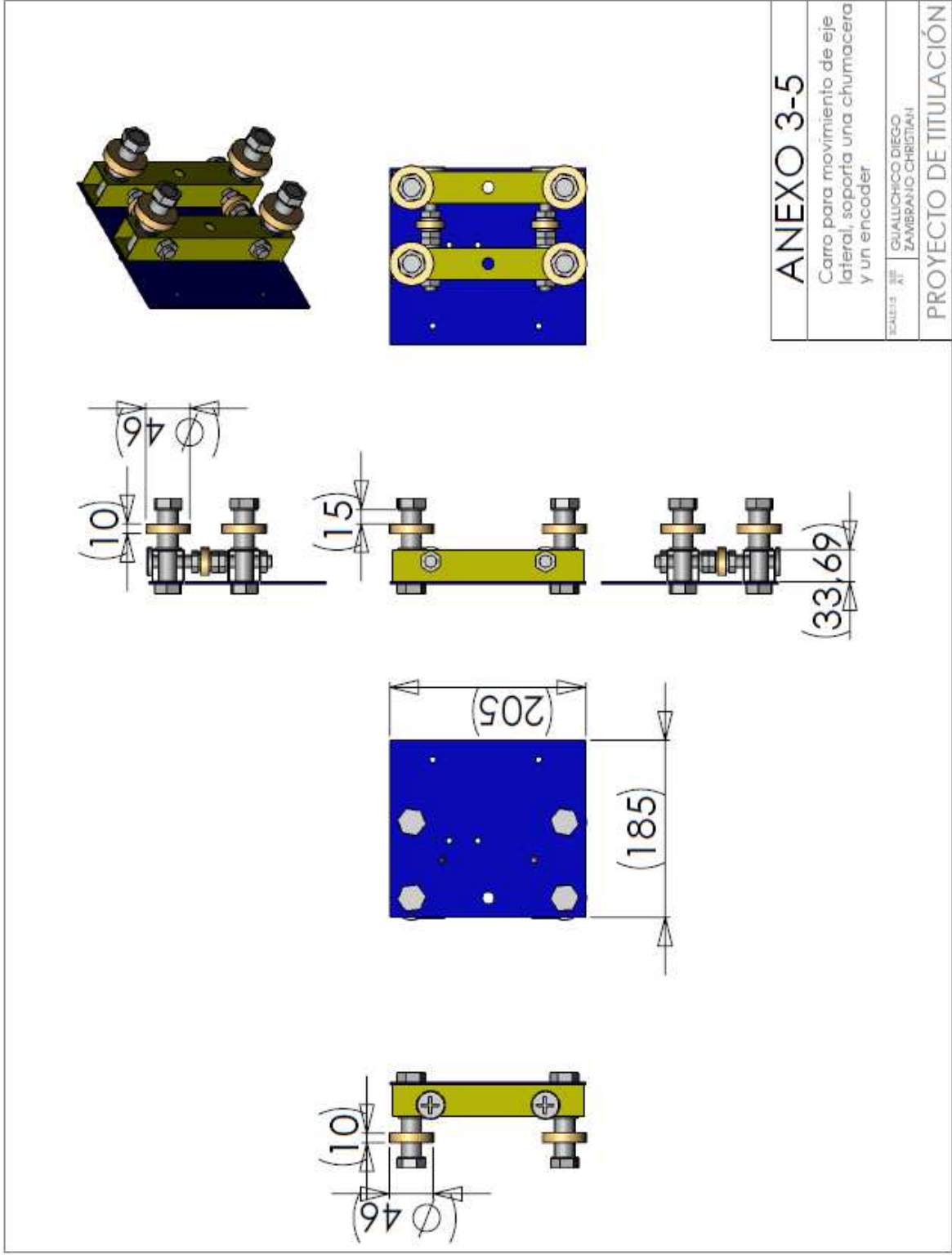


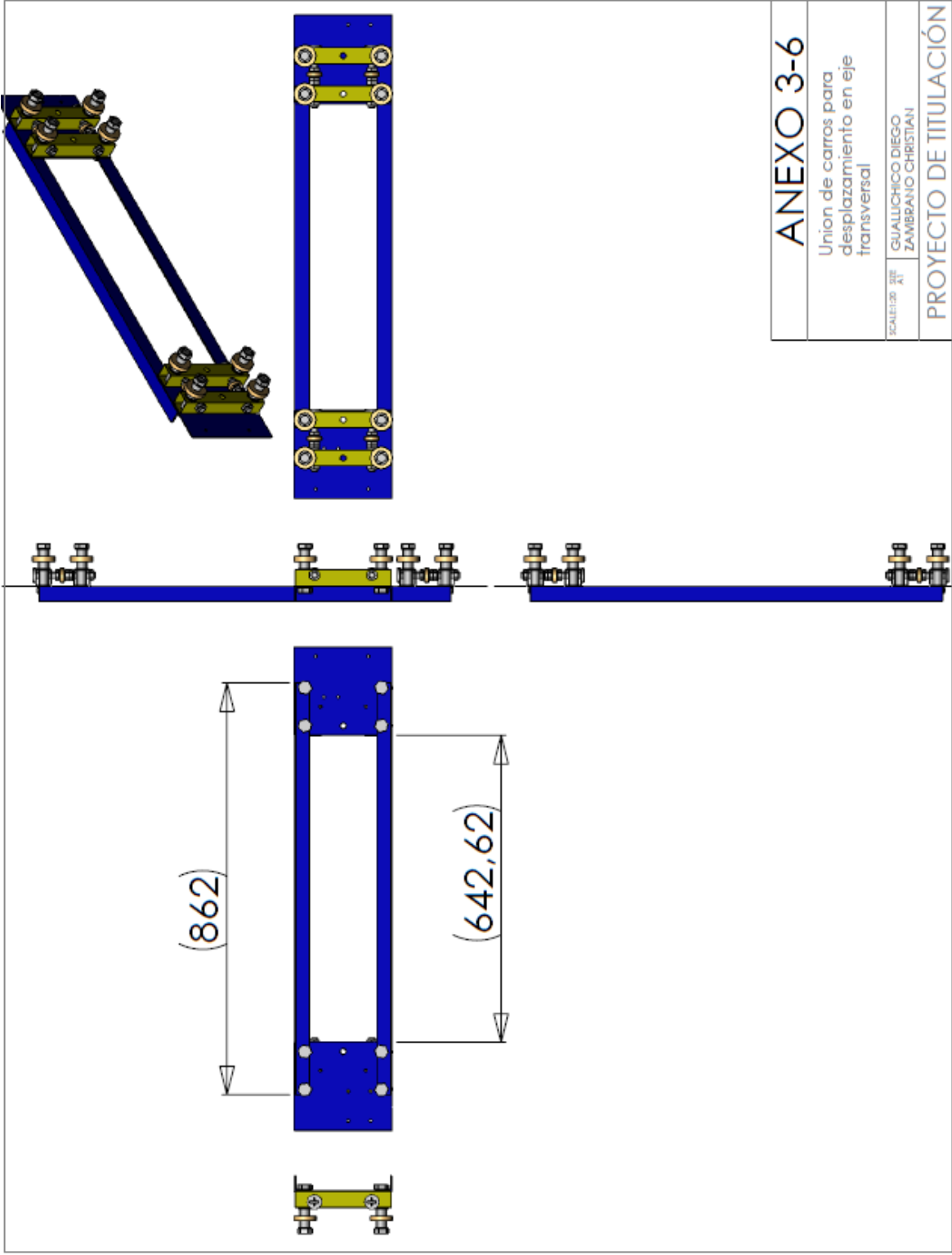


ANEXO 3-3	
Mesa base para soporte de plancha metálica y sistema de control	
SCALE: 1:1	GUILLUCHICO DIEGO ZAMBRANO CHRISTIAN
PROYECTO DE TITULACIÓN	

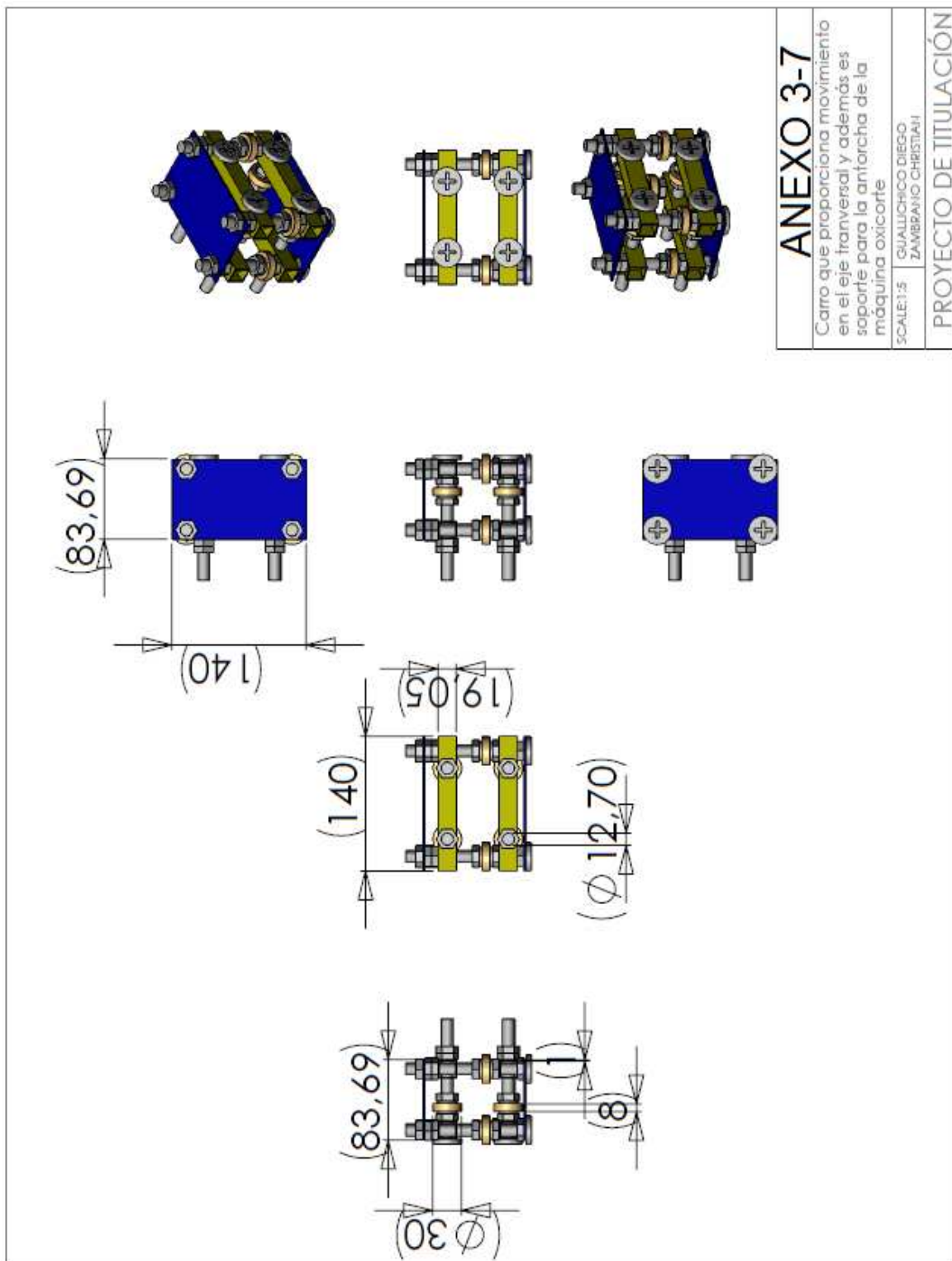


ANEXO 3-4
 Carro para movimiento de eje lateral, que soporta una chumacera y un motor dc
 ESCUELA Nº 31 | GUALICHICO DIEGO ZAMBRANO CHRISTIAN
 PROYECTO DE TITULACIÓN





ANEXO 3-6	
Union de carros para desplazamiento en eje transversal	
SCALE: 3/4" = 1"	GUALLICHICO DIEGO ZAMBRANO CHRISTIAN
PROYECTO DE TITULACIÓN	

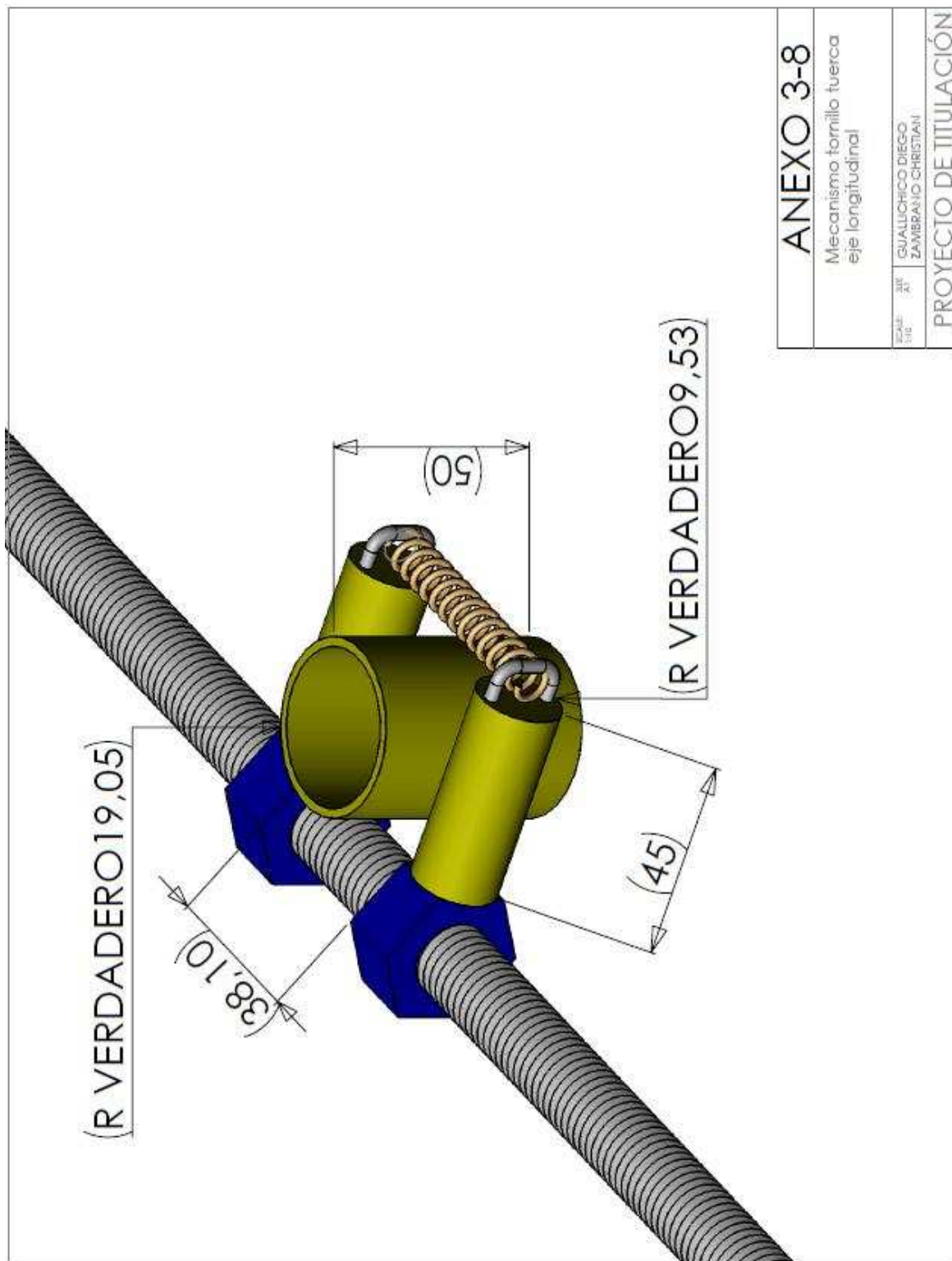


ANEXO 3-7

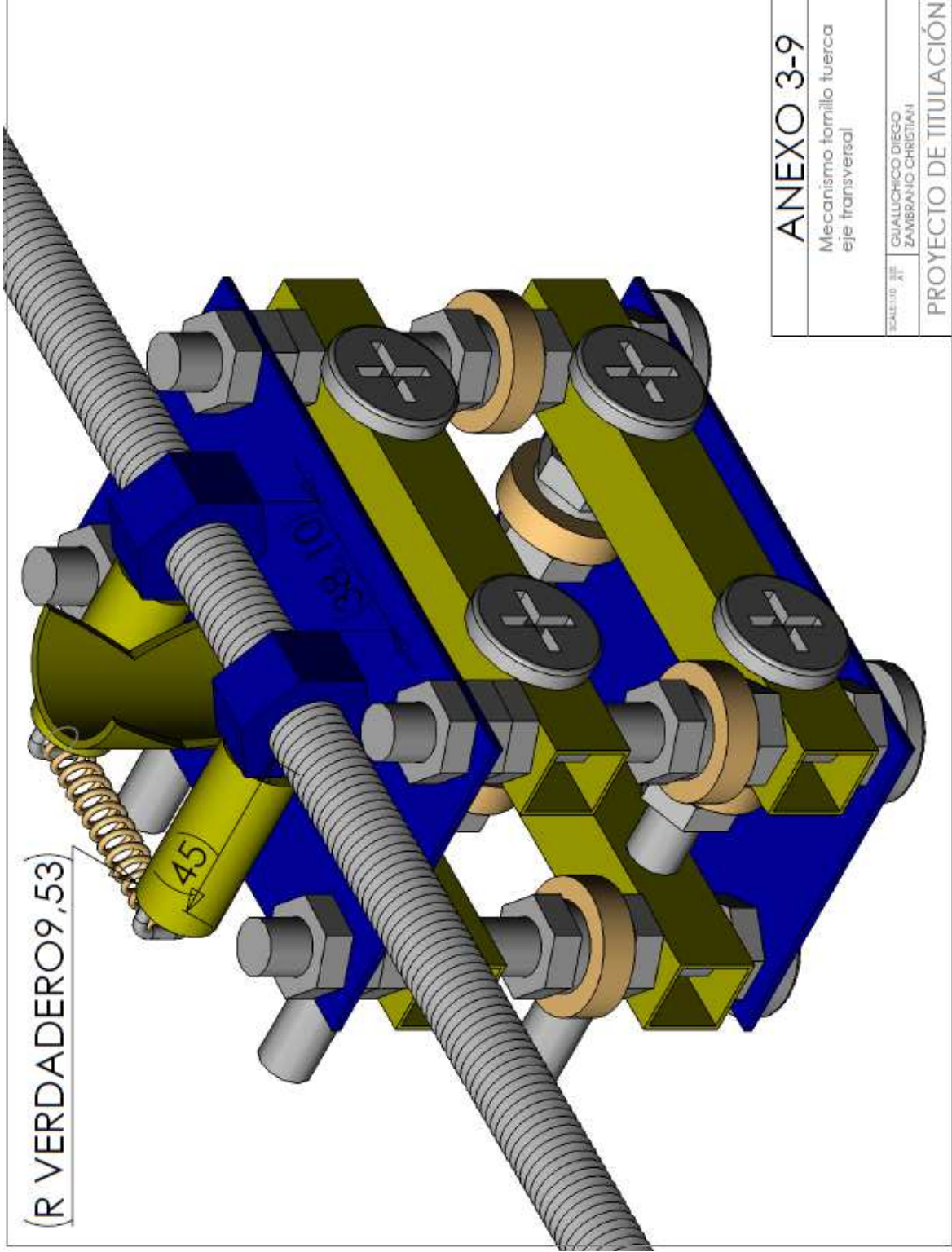
Carro que proporciona movimiento en el eje transversal y además es soporte para la antorcha de la máquina oxiacorte

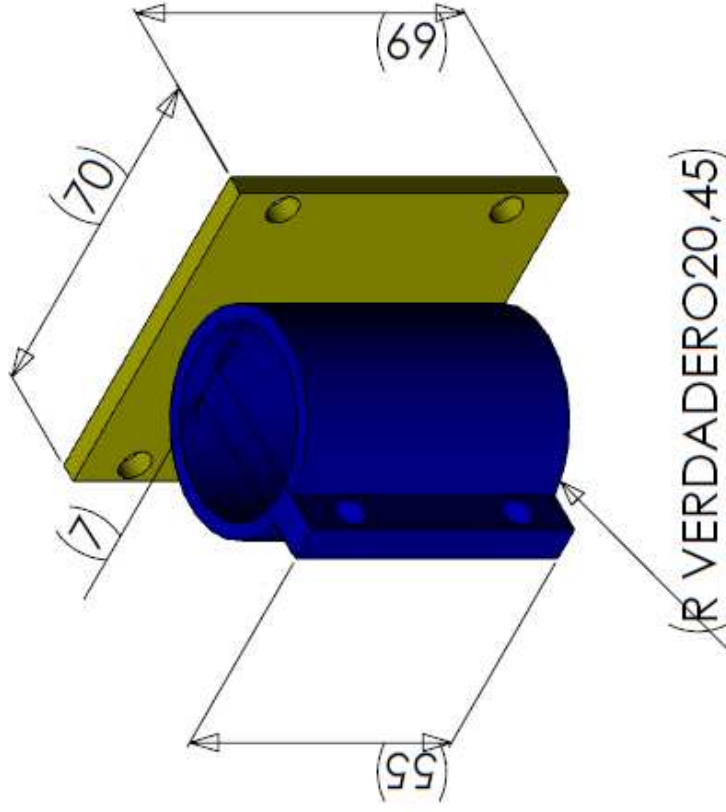
SCALE:1:5
 GUALICHICO DIEGO
 ZAMBRANO CHRISTIAN

PROYECTO DE TITULACIÓN

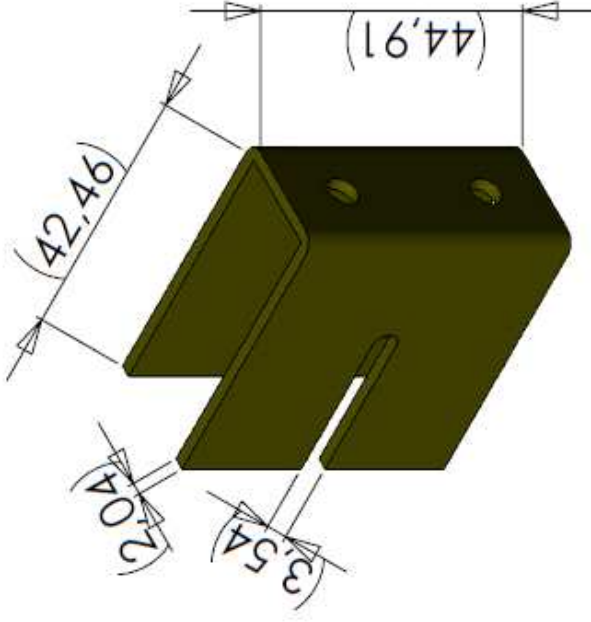


ANEXO 3-8	
Mecanismo tornillo tuerca eje longitudinal	
<small>ELABORADO POR</small> <small>DISEÑADO POR</small>	<small>GUALICHICO DIEGO</small> <small>ZAMBRANO CHRISTIAN</small>
PROYECTO DE TITULACIÓN	





ANEXO 3-10	
Soporte para motor	
ESQUEMA Nº	GUALICHICO DIEGO ZAMBRANO CRISTIAN
PROYECTO DE TITULACIÓN	



ANEXO 3-11

Soporte para encoder

ESCUELA ^{UNIV} _{UNIV} GUAYLICHICO DIEGO
ZAMBRANO CHRISTIAN

PROYECTO DE TITULACIÓN

ANEXO 4 DIAGRAMAS DE CIRCUITOS Y MONTAJE DEL TABLERO PRINCIPAL

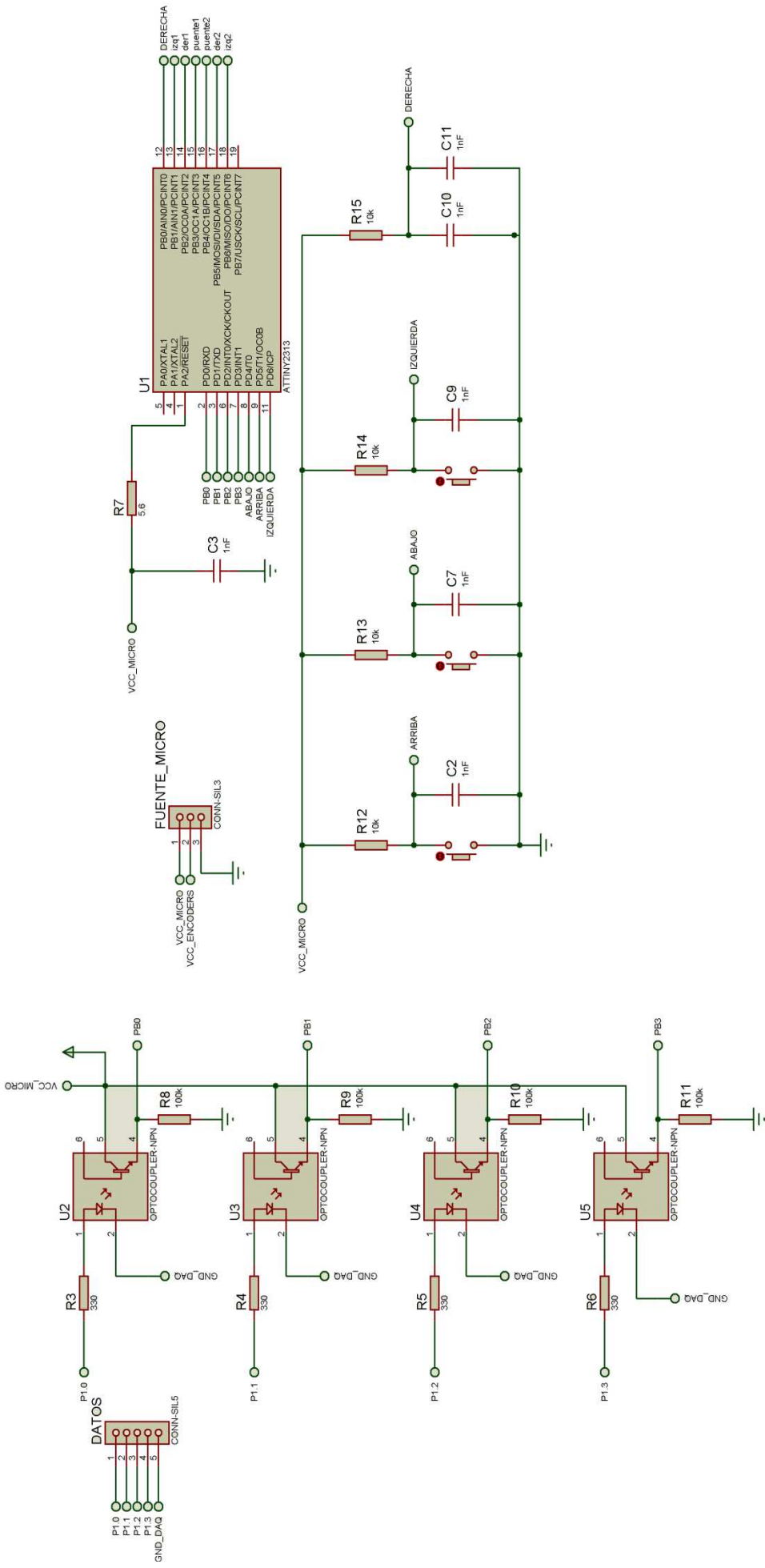


FIGURA ANEXO 4.1

Diagrama esquemático, Circuito de control PARTE1

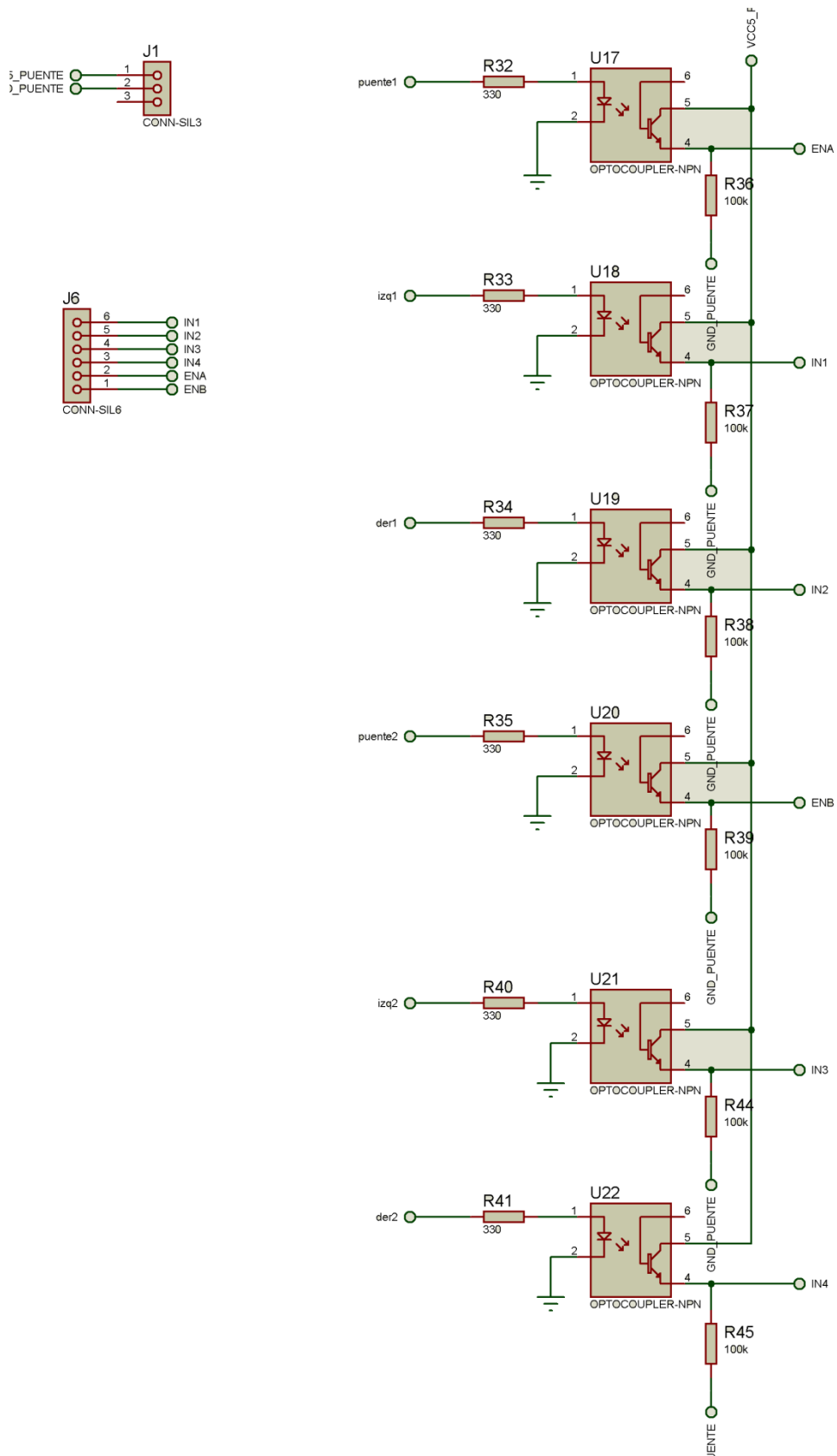


FIGURA ANEXO 4.2

Diagrama esquemático, Circuito de control PARTE 2

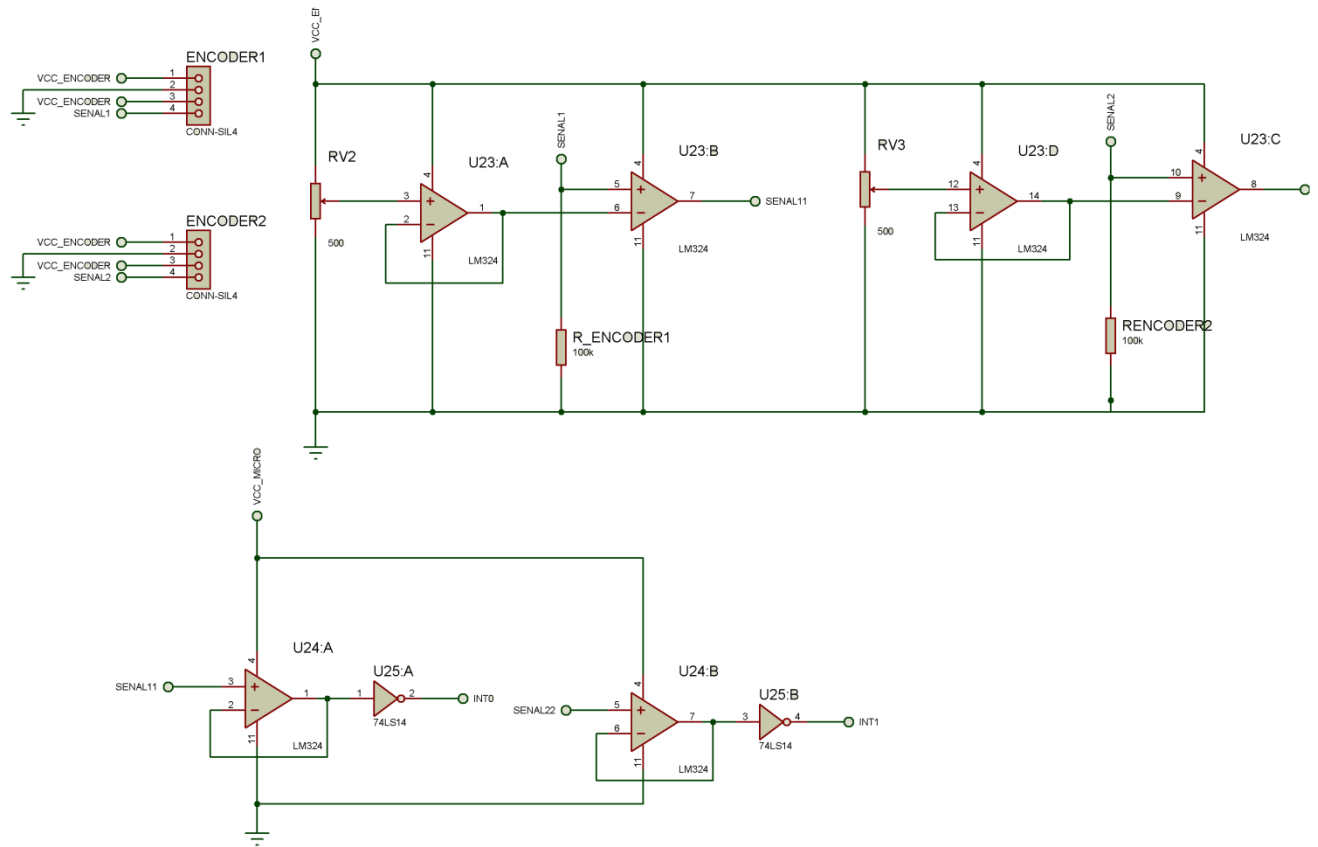


FIGURA ANEXO 4.3

Diagrama esquemático, Circuito de control PARTE 3

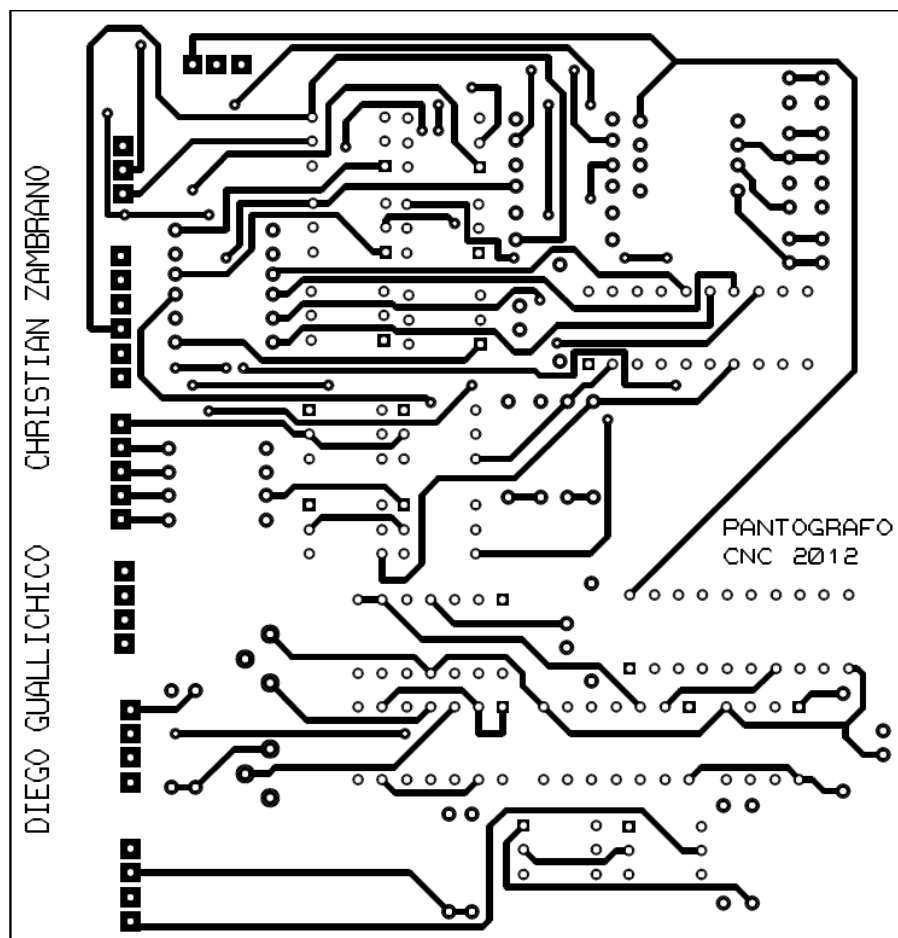


FIGURA ANEXO 4.4 Circuito de control, lado superior

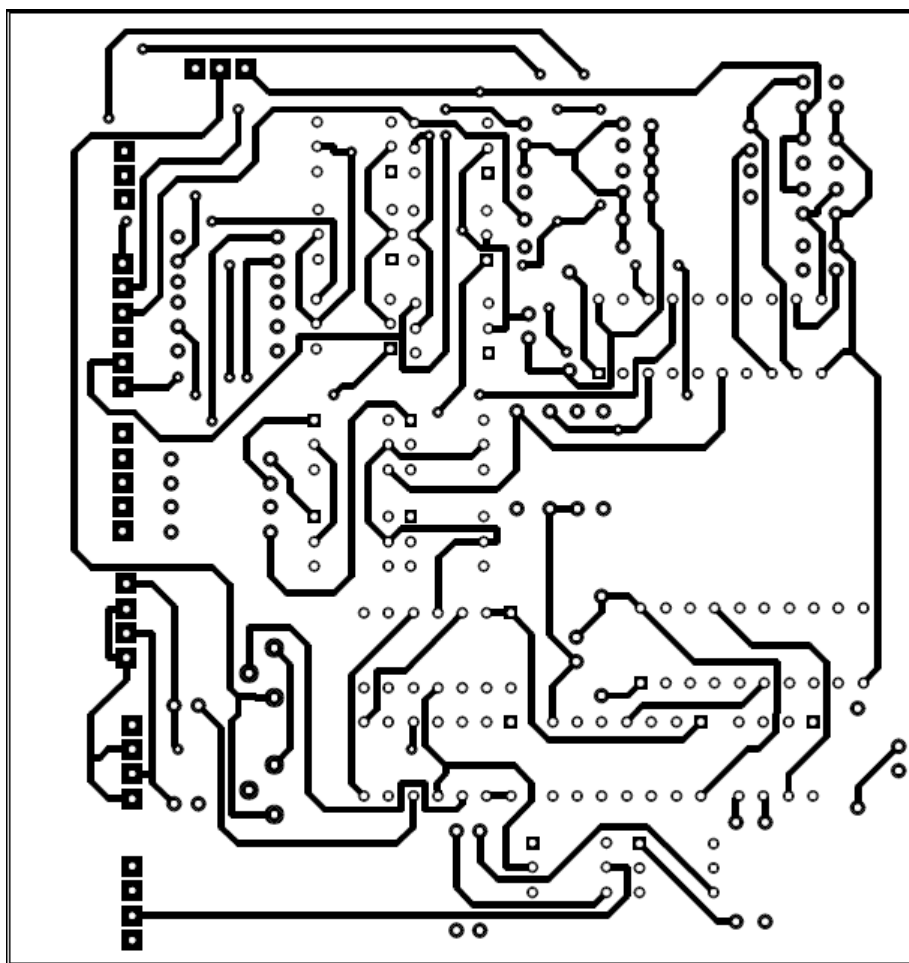


FIGURA ANEXO 4.5 Circuito de control, lado inferior

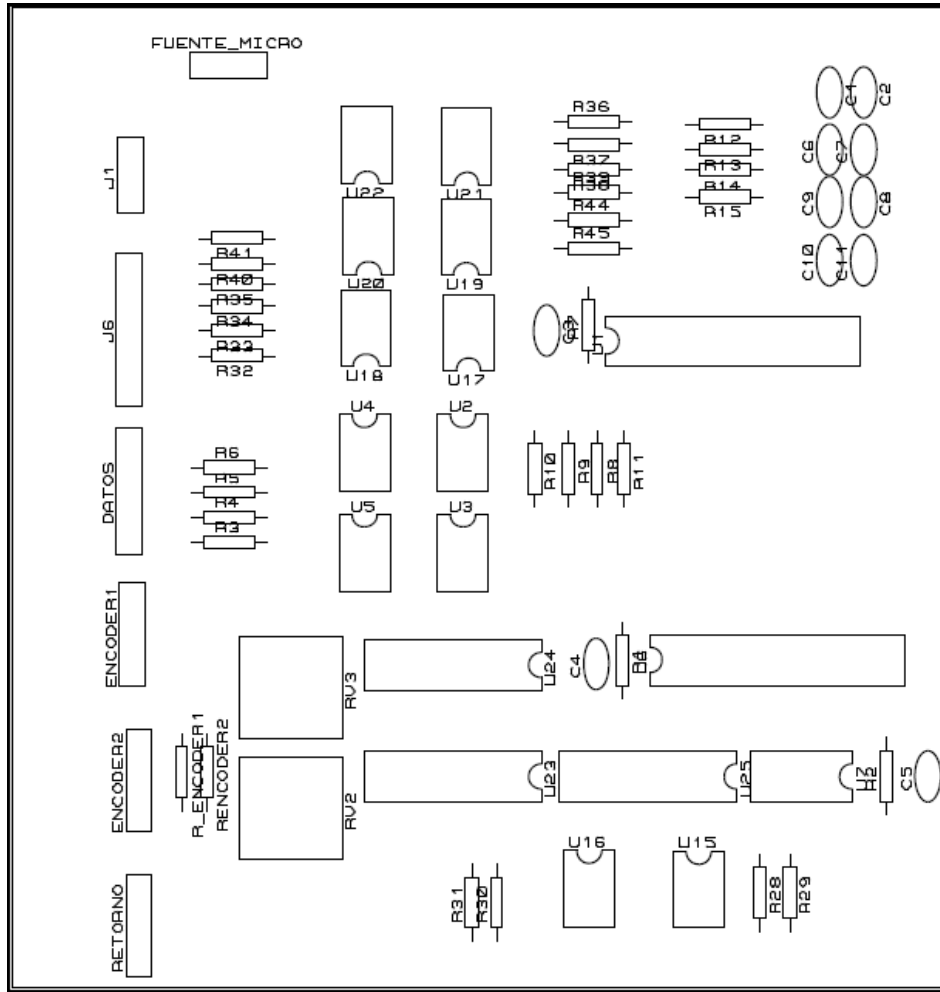


FIGURA ANEXO 4.6 Distribución de elementos circuito de control, lado superior

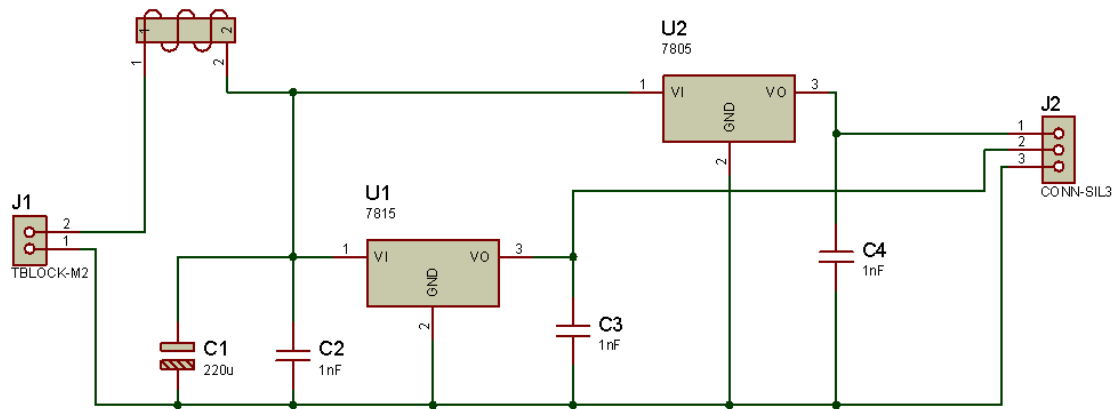


FIGURA ANEXO 4.7 Diagrama esquemático Circuito de Potencia PARTE 1

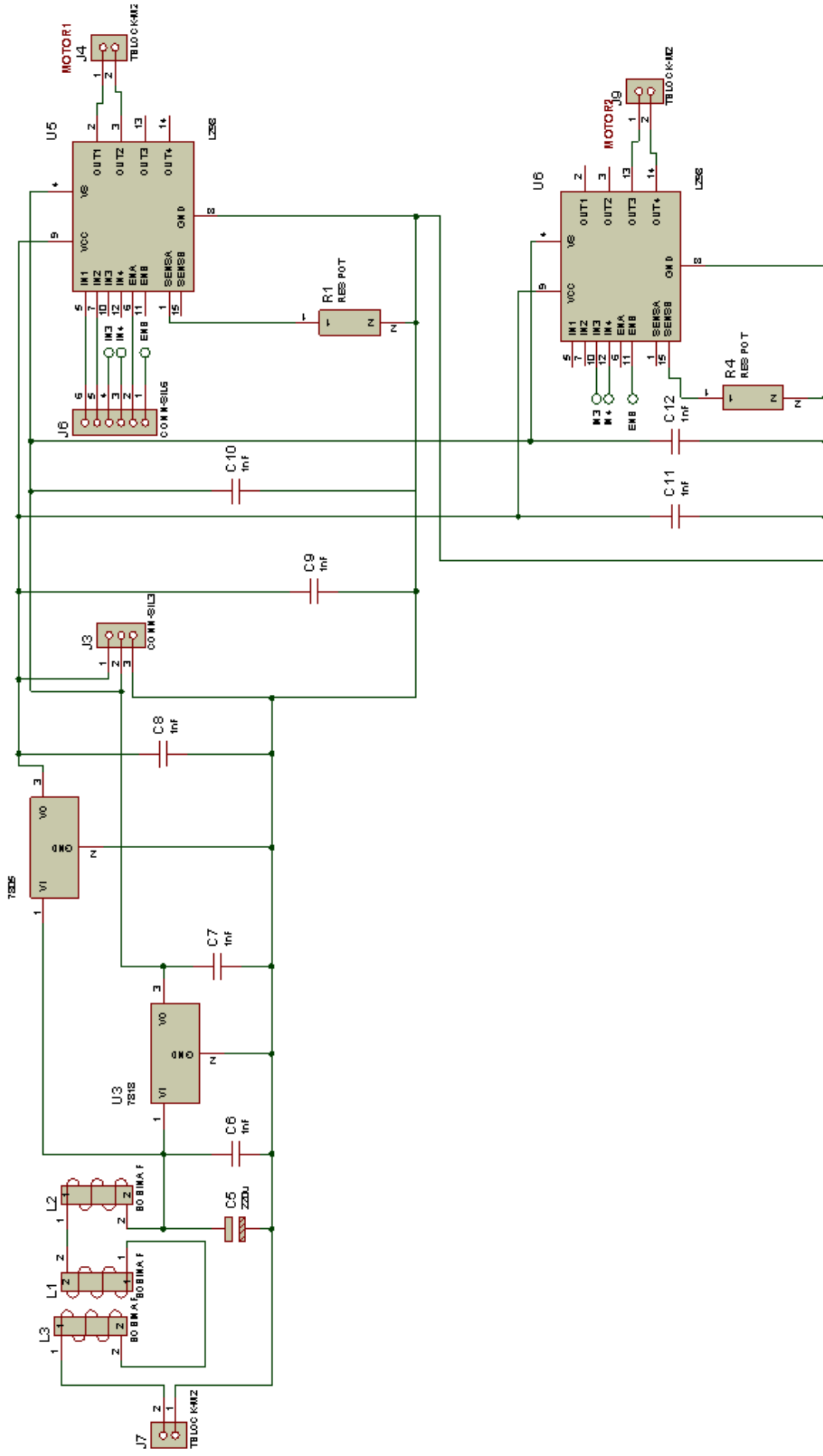


FIGURA ANEXO 4.8 Diagrama esquemático circuito de potencia PARTE 2

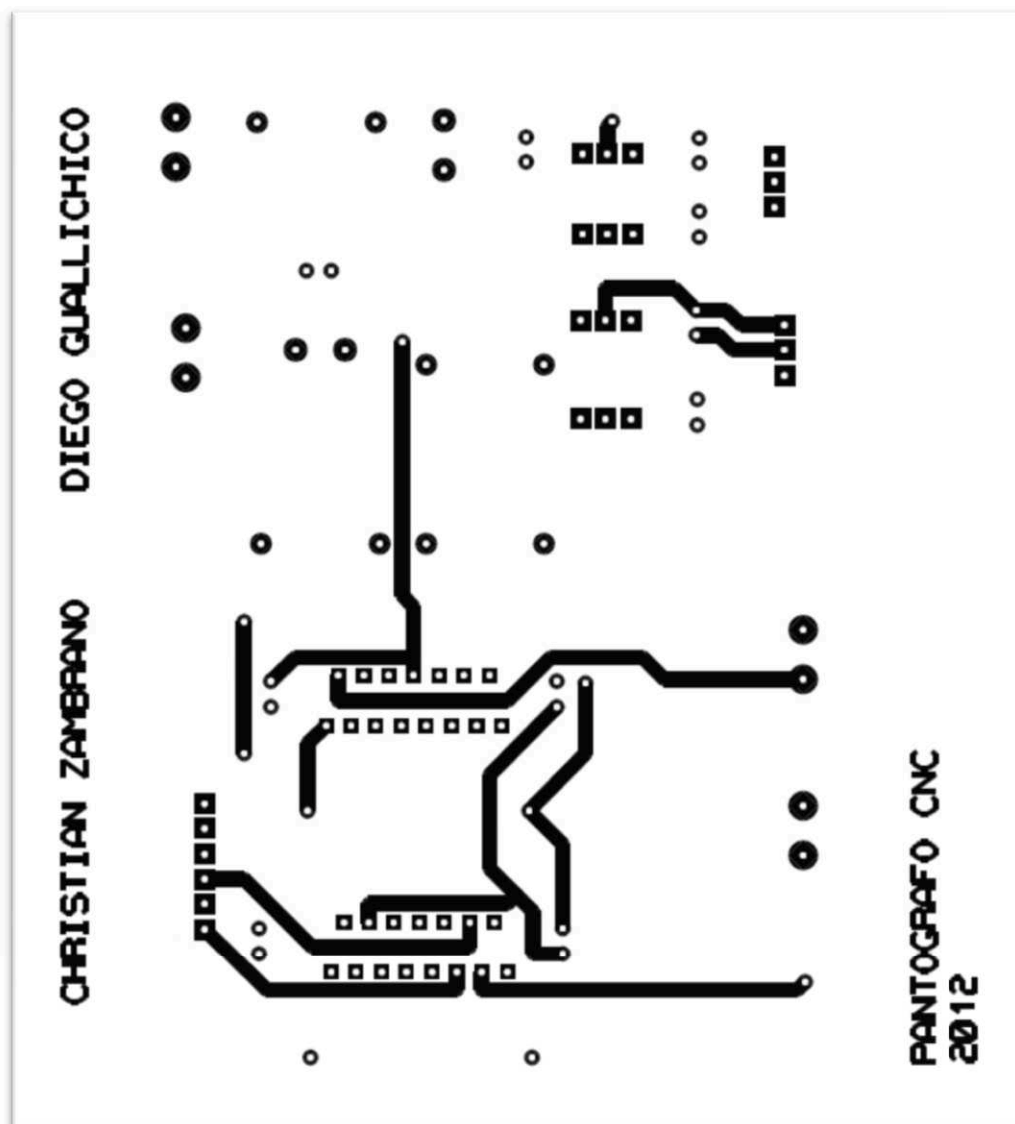


FIGURA ANEXO 4.9 Circuito de potencia, lado superior.

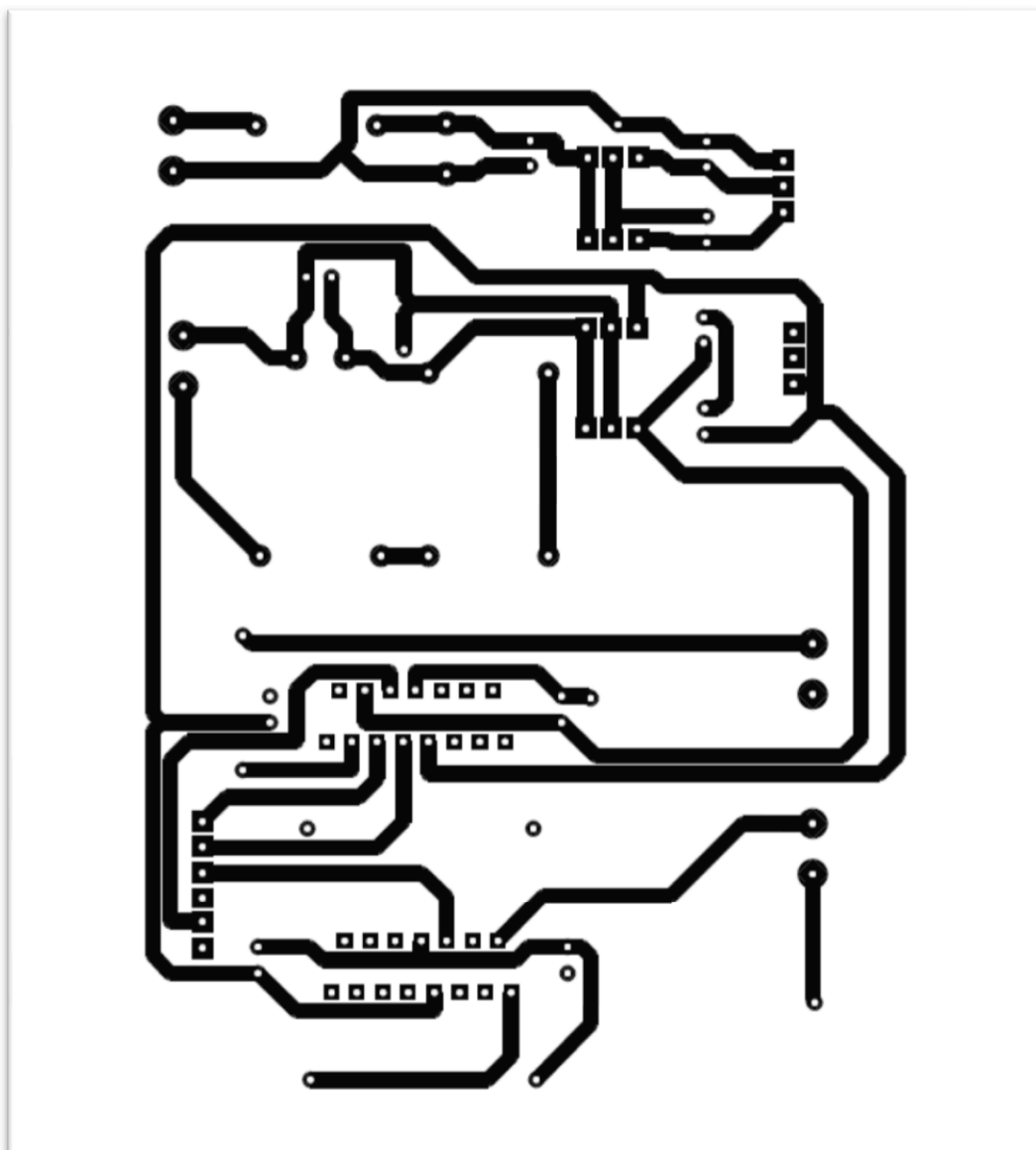


FIGURA ANEXO 4.10 Circuito de potencia, lado inferior

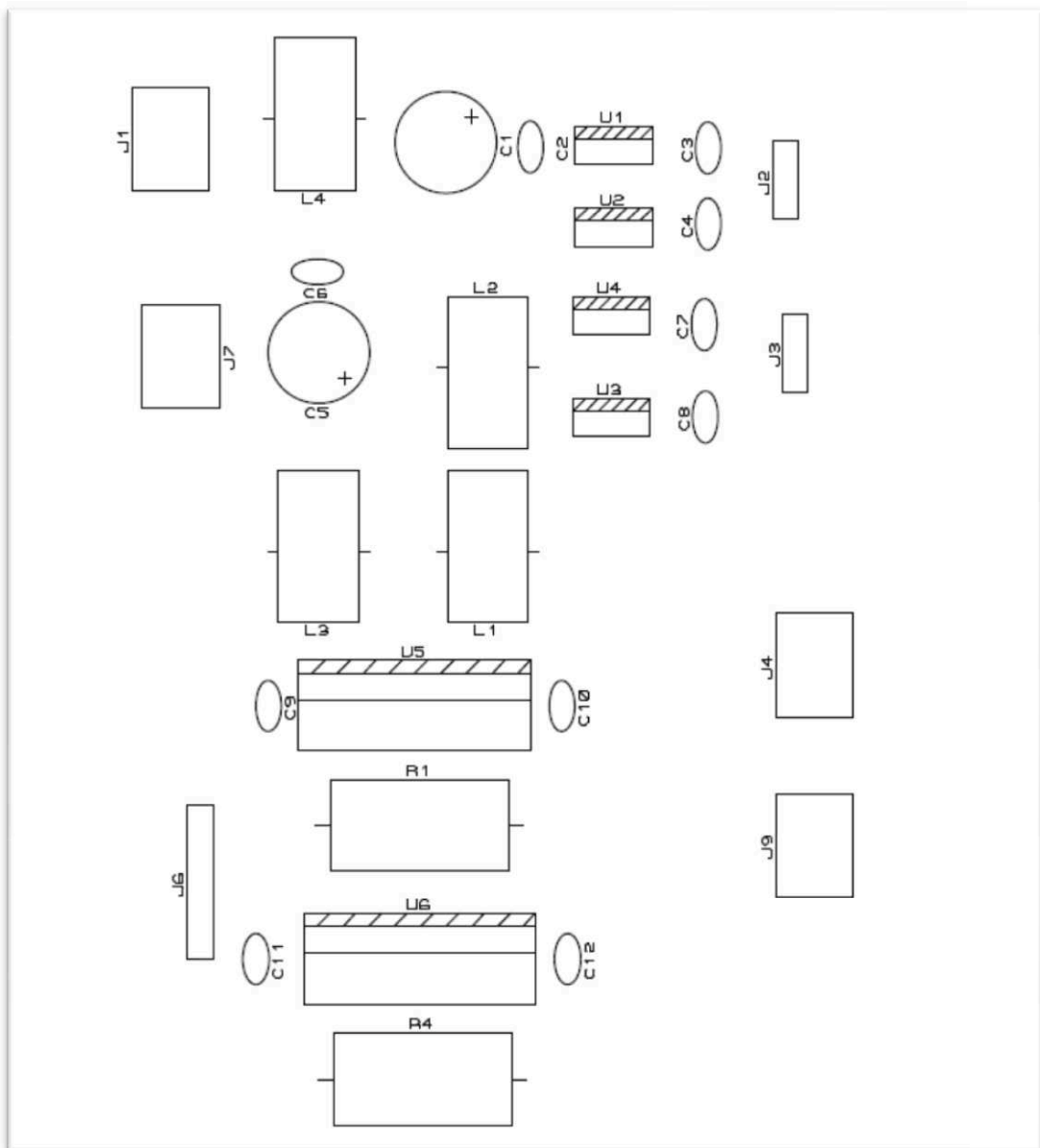
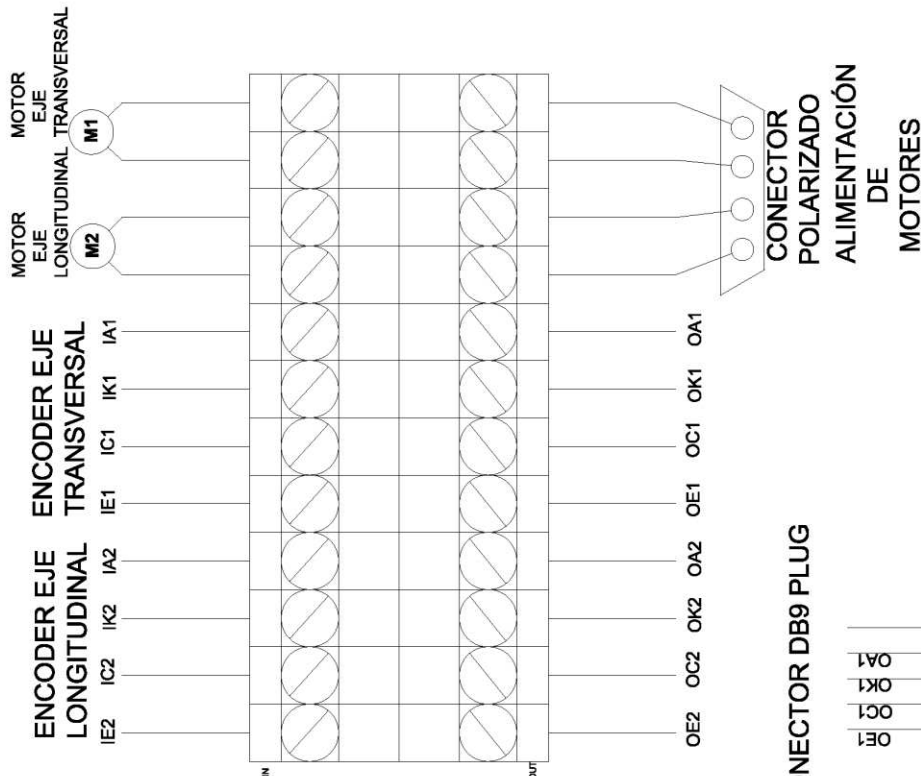


FIGURA ANEXO 4.11 Distribución de elementos Circuito de Potencia, lado superior

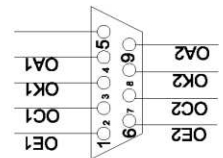
BORNE DIN OTTA2.5



CORRESPONDENCIA

- E1: Emisor ENCODER EJE TRANSVERSAL
- C1: Colector ENCODER EJE TRANSVERSAL
- A1: Ánodo ENCODER EJE TRANSVERSAL
- K1: Cátodo ENCODER EJE TRANSVERSAL
- E2: Emisor ENCODER EJE LONGITUDINAL
- C2: Colector ENCODER EJE LONGITUDINAL
- A2: Ánodo ENCODER EJE LONGITUDINAL
- K2: Cátodo ENCODER EJE LONGITUDINAL

CONECTOR DB9 PLUG



ANEXO 4-12

Diagra de conexiones Mesa - Panel de control. Salidas de Conector DB9 Salida conector polarizado para motores	
SCALE: 1:1	GUALLICHICO DIEGO ZAMBRANO CHRISTIAN
PROYECTO DE TITULACIÓN	