



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL

CARRERA DE INGENIERÍA INDUSTRIAL EN PROCESOS DE AUTOMATIZACIÓN

Tema:

“MÓDULO CONTROLADOR PARA LA MÁQUINA FRESADORA CNC DE LA FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL”

Proyecto de trabajo de graduación modalidad TEMI Trabajo Estructurado de Manera Independiente, presentado previo a la obtención del título de Ingeniero Industrial en Procesos de Automatización.

Sublínea de Investigación: Sistemas de control automatizado e instrumentación virtual para procesos industriales de baja y alta potencia.

AUTOR: Rodríguez Vargas Paúl Jesús.

TUTOR: Ing. Wilian López Gordon

Ambato - Ecuador

Abril - 2015

APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de investigación sobre el tema: Módulo controlador para la máquina fresadora CNC de la Facultad de Ingeniería en Sistemas Electrónica e Industrial, elaborado por el señor Rodríguez Vargas Paúl Jesús, estudiante de la Carrera de Ingeniería Industrial en procesos de Automatización, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el Art. 16 del Capítulo II, del Reglamento de Graduación para Obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato.

Ambato, Abril 2015

EL TUTOR

Ing. Mg. Wilian López Gordon

AUTORÍA

El presente trabajo de investigación titulado: “MÓDULO CONTROLADOR PARA LA MÁQUINA FRESADORA CNC DE LA FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL”. Es absolutamente original, autentico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, Abril 2015

EL AUTOR

Rodríguez Vargas Paúl Jesús
C.I. 180401773-7

APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Jessica López y Ing. Carlos Sánchez, revisó y aprobó el informe Final del trabajo de graduación titulado “Módulo controlador para la máquina fresadora CNC de la Facultad de Ingeniería en Sistemas Electrónica e Industrial”, presentado por el señor Rodríguez Vargas Paúl Jesús de acuerdo al Art. 17 del Reglamento de graduación para obtener el título Terminal de tercer nivel de la Universidad Técnica de Ambato.

Ing. Vicente Morales Lozada, Mg.
PRESIDENTE DEL TRIBUNAL

Ing. Jessica López. Mg.
DOCENTE CALIFICADOR

Ing. Carlos Sánchez. Mg.
DOCENTE CALIFICADOR

DEDICATORIA

Dedico éste trabajo a Dios por darme sus bendiciones y sabiduría en cada meta que me he propuesto.

A mi mamá Magdalena que desde el cielo siempre me cuida y me protege, a mi querido y amado Padre Alfredo por cuidarme pacientemente, por estar conmigo alentándome en todo momento para ser una persona de bien, brindándome sus consejos, su apoyo y el ejemplo de ser el mejor padre.

A mis hermanos Mercedes, Rosario y en especial a mi ñaño David, que me brindaron su apoyo y me alentaron a seguir en mi camino.

A mis queridos amigos Isra, Vivi, Mauricio y en especial a mi querida Lucia por alentarme a que siga a delante con mis metas.

Jesús Rodríguez

Índice de Contenidos

Aprobación del tutor	i
Autoría	ii
Aprobación de la comisión calificadora	iii
Dedicatoria.....	iv
Índice de contenidos	v
Índice de figuras.....	vii
Índice de tablas	x
Resumen.....	xi
Abstract.....	xii
Glosario de términos y acrónimos	xiii
Introducción	xv
CAPITULO 1.....	1
EL PROBLEMA.....	1
1.1. Tema.....	1
1.2. Planteamiento del problema.....	1
1.3. Delimitación.....	2
1.4. Justificación.....	3
1.5. Objetivos	4
CAPITULO 2.....	5
MARCO TEÓRICO	5
2.1. Antecedentes investigativos	5
2.2. Fundamentación teórica	7
2.3. Propuesta de solución.....	23
CAPITULO 3.....	24
METODOLOGÍA.....	24
3.1. Modalidad de la investigación	24
3.2. Población y muestra	25
3.3. Recolección de la información.....	25
3.4. Procesamiento y análisis de datos	25

3.5. Desarrollo del proyecto	25
CAPITULO 4.....	27
DESARROLLO DE LA PROPUESTA	27
4.1. Condiciones físicas en las que se encuentra inicialmente la máquina CNC	27
4.2. Características técnicas y operativas de la máquina CNC:	28
4.3. Componentes que contiene la máquina CNC.....	31
4.3.1. Motores.....	32
4.3.1.1. Motor eje X y eje Y.	32
4.3.1.2. Motor eje Z	33
4.3.1.3. Motor husillo.....	34
4.3.2. Contactos:	34
4.3.3. Cerradura de puerta	36
4.3.4. Potenciómetros	36
4.4. Funcionamiento de partes mecánicas y control de los mecanismos.	37
4.4.1. Tornillo sin fin.....	37
4.4.2. Transmision de polea.....	37
4.5. Descripción y funcionamiento de partes electrónicas para control de los mecanismos	39
4.5.1. Isolator	40
4.5.2. Filtro IRF.....	40
4.5.3. Driver de husillo (Spindle)	41
4.5.4. Transformador CA/CA	42
4.5.5. Controlador de máquina CNC.	43
4.6. Especificaciones de los dispositivos necesarios para habilitar la máquina CNC. 48	
4.7. Elaboración de los planos eléctricos del sistema a automatizarse	49
4.8. Circuitos y diagramas de control para la máquina CNC.....	50
4.8.1. Driver de motores pasos paso.....	51
4.8.2. Tarjeta controladora	55
4.8.3. Tarjeta de comunicaión	58
4.9. Diseño del software de control para la interacción del usuario y la máquina CNC 60	
4.9.1. Interfaz a nivel hardware.	60

4.9.2.	Interfaz a nivel software	60
4.9.3.	Diseño de la interfaz HMI	60
4.9.4.	Partes del sistema HMI diseñado	65
4.10.	Programación en lenguaje G de labview.	71
4.11.	Implementación del controlador y desarrollo del manual de usuario.....	77
4.12.	Resultados del funcionamiento.....	78
CAPITULO 5.....		87
CONCLUSIONES Y RECOMENDACIONES		87
5.1.	Conclusiones	87
5.2.	Recomendaciones.....	88
Anexos		92

Índice de Figuras

Figura 1.	Driver controlador para motor bipolar	8
Figura 2:	Final de carrera mecánico	9
Figura 3:	Diagrama de conexión anti rebote para pulsador.....	10
Figura 4:	Diagrama de conexión para fuente de alimentación	11
Figura 5:	Dispositivo microcontrolador PIC.	12
Figura 6:	Conectores USB	13
Figura 7:	Codificación NRZI para comunicación USB.....	14
Figura 8:	Representación de interpolación lineal	17
Figura 9:	Gráfica de interpolación circular.....	18
Figura 10:	Panel frontal o interfaz HMI de LabVIEW	19
Figura 11:	Diagrama de bloques para la programación en LabVIEW	20
Figura 12:	Secuencia para la adquisición de señales.	22
Figura 13:	Conector macho tipo 357F de máquina CNC	31
Figura 14:	Final de Carrera.....	35
Figura 15:	Paro de emergencia.	35
Figura 16:	Cerradura electromagnética.	36
Figura 17:	Potenciómetros de la maquina CNC.	37

Figura 18: Sistema de transmisión mecánico	38
Figura 19: Caja Eléctrica	39
Figura 20: Isolator.....	40
Figura 21: Filtro RFI.....	41
Figura 22: Driver husillo	42
Figura 23: Transformador.....	43
Figura 24: Tarjeta controladora	43
Figura 25: Conector de finales de carrera.....	44
Figura 26: Conector de salidas digitales.....	45
Figura 27: Conector de entradas análogas	45
Figura 28: Conector de motores	46
Figura 29: Conector salidas análogas	46
Figura 30: Conector entradas digitales	47
Figura 31: Fuente de alimentación	47
Figura 32: Fuente da alimentación de tarjeta.....	48
Figura 33: Diagrama de flujo de control máquina CNC.....	51
Figura 34: Diagrama esquemático de motores	52
Figura 35: Diagrama esquemático de protección de motores.....	52
Figura 36: Diagrama esquemático de control de corriente	53
Figura 37: Diagrama esquemático fijar la corriente	53
Figura 38: Diseño PCB de driver de motores	54
Figura 39: Diagrama esquemático de entradas del controlador.....	55
Figura 40: Diagrama esquemático de entradas análogas	56
Figura 41: Programación de Control de motores.....	56
Figura 42: Diagrama esquemático de fuente de alimentación.....	57
Figura 43: Diagrama esquemático del Arduino uno.....	59
Figura 44: Diagrama esquemático de control de Arduino uno.....	59
Figura 45: Software HMI para el control de la máquina CNC.....	61
Figura 46: Diagrama de Flujo del HMI	62
Figura 47: Diagrama de Flujo del HMI	63
Figura 48: Diagrama de Flujo del HMI	64
Figura 49: Pestaña de configuración.....	65

Figura 50: Pestaña "MAQUINAR"	67
Figura 51: Cargar código de instrucciones CNC	71
Figura 52: Envío de instrucciones	72
Figura 53: Código para incrementar nueva instrucciones de maquinado	72
Figura 54: Envío de datos a la tarjeta controlador	72
Figura 55: Código de monitoreo de señales	73
Figura 56: Código de alertas	74
Figura 57: Decodificado de Trama serial	74
Figura 58: Decodificado de Trama serial	75
Figura 59: Decodificado de Trama serial	75
Figura 60: Visualización de maquinado	76
Figura 61: Envío de señales al controlador.....	76
Figura 62: Diseño Logo UTA.....	78
Figura 63: Maquinado Logo UTA.....	79
Figura 64: Análisis de medidas	79
Figura 65: Consumo de corriente	81
Figura 66: Tendencia de temperaturas.....	83
Figura 67: Diseño "Caballo".....	84
Figura 68: maquinado de "Caballo"	84
Figura 69: Desbaste de superficie.....	85
Figura 70: Desbaste de superficie.....	85
Figura 71: Diseño desbaste multi capa.....	86
Figura 72: Desbaste multi capa	86

Índice de Tablas

Tabla 1: Secuencia de activación de bobinado en motor a pasos bipolar	9
Tabla 2: Configuración estándar de pines usb	14
Tabla 3: Códigos G Generales	15
Tabla 4: Códigos M- Misceláneos	16
Tabla 5: Estructura de un bloque de programa CNC	16
Tabla 6: Análisis de elementos	28
Tabla 7: Descripción del conector tipo 357F.....	31
Tabla 8: Características técnicas de los motores x, y.	32
Tabla 9: Código de Colores de las bobinas motor pasó a paso.....	33
Tabla 10: Características técnicas de los motores z.....	33
Tabla 11: Código de colores de las bobinas motor pasó a paso.....	34
Tabla 12: Descripción de pines de tarjeta	44
Tabla 13: Descripción componentes técnicos.....	48
Tabla 14: Componentes del driver de motores	54
Tabla 15: Componentes de tarjeta controladora	58
Tabla 16: Descripción de elementos del HMI de la pestaña ajustes.....	66
Tabla 17: Descripción de elementos del HMI de la pestaña “MAQUINAR”	67
Tabla 18: Descripción de elementos del HMI de la pestaña ajustes.....	68
Tabla 19: Descripción de elementos del HMI de la pestaña ajustes.....	69
Tabla 20: Descripción de elementos del HMI de la pestaña ajustes.....	70
Tabla 21: Comandos G principales.....	77
Tabla 22: Análisis de medidas	80
Tabla 23: Análisis de corrientes	81
Tabla 24: Análisis de temperatura	82

Índice de ecuaciones

Ecuación 1: Trasmisión de movimiento en engranajes.....	54
Ecuación 2: Calculo para resolución de mecanismos.....	56
Ecuación 3: Control de corriente en motores paso a paso.....	69
Ecuación 4: Cálculo del error en medidas.....	79

RESUMEN

El presente proyecto de graduación tiene como finalidad realizar un módulo o tarjeta controladora para la máquina Fresadora Denford Novamill CNC (Control Numérico Computarizado) de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato, que por fallas en su circuitería de control se encuentra inactiva hace dos años, ésta tarjeta electrónica no se consigue en el mercado local y debido a su alto costo y disponibilidad es complicado adquirir una, es por ello que se ve en la necesidad de realizar un módulo controlador sustituto que realice un similar trabajo al original diseñado con componentes de fácil acceso, con el propósito de habilitar nuevamente el funcionamiento del equipo CNC, en el cual los estudiantes y profesores puedan desarrollar prácticas o trabajos basados en diseños CAD/CAM.

Para alcanzar este propósito fue necesario el diseño de las diversas tarjetas electrónicas que permitirá que éste equipo entre nuevamente en funcionamiento, entre las tarjetas electrónicas se tiene: la placa de comunicaciones que se encarga de comunicar el HMI (Interfaz Hombre Máquina), con la máquina CNC mediante la tarjeta Arduino Uno utilizando protocolo usb, la siguiente tarjeta es la de control de motores a pasos que dispone la máquina, y finalmente la tarjeta de control que se encarga de interactuar con los demás módulos electrónicos, permitiendo así el correcto funcionamiento de los distintos mecanismos del equipo.

La interacción del usuario con la máquina CNC se realiza por medio de la manipulación de botones virtuales en el software HMI desarrollado en Labview, cuyo control permite cargar secuencias o instrucciones de códigos G o comandos CNC creados en software de diseño CAD/CAM como lo es el Master-CAM, éste HMI tiene visualizadores, alarmas, gráficas en tiempo real, comandos que permiten controlar y monitorear y supervisar la máquina CNC en cada momento de trabajo.

Alcanzando de esta manera los objetivos y brindando a los estudiantes una herramienta para su desarrollo profesional dentro de la Universidad Técnica de Ambato.

ABSTRACT

This graduation project aims to make a module or controller board for CNC milling machine, Faculty of Systems Engineering, Electronics and Industrial Technical University of Ambato, in order to enable CNC Router theoretical development practical student of Industrial Engineering.

To achieve this purpose it was necessary to design the various electronic cards that allow the CNC machine into operation , including electronic cards have: communication board that is responsible for communicating the HMI with CNC machine using Arduino board One using usb protocol, the next card is the control of stepper motors available to the machine , and finally the control board that is responsible for interacting with the other cards enabling thus the correct workings of the mechanisms of the machine.

The HMI allows the interaction between the software and the machine in order to load the codes that occur in numerical control programs, the HMI has displays, alarms, graphical real-time commands to control and monitor the CNC machine.

Once the driver and set up the machine presents the new design of the software and controls the mechanisms of CNC.

Thus achieving the goals and giving students a tool for professional development within the Technical University of Ambato.

GLOSARIO DE TÉRMINOS Y ACRÓNIMOS

ABSOLUTA: En la programación absoluta, el cero es el punto desde el cual todas las demás dimensiones son descritas.

CICLO AUTOMÁTICO: A modo de operación de control que se ejecuta continuamente un ciclo o programa almacenado hasta que una parada del programa o al final de la palabra programa es leído por el controlador.

EJES: Los planos de movimiento de la herramienta de corte, generalmente se hace referencia como X (horizontal izquierda y la derecha, paralelo al borde frontal de la mesa), Y (horizontal hacia delante y hacia atrás, paralela al borde lateral de la mesa) y Z (directamente vertical). Las combinaciones de los 3 permitan coordenadas exactas.

BILLET: El material real que se está mecanizando, a veces referido como la "pieza a maquinar".

CARÁCTER: Un número, letra o símbolo que se recoge en un programa CNC.

CIRCULAR: Término código G interpolación para un movimiento de arco programado.

COMANDO: Una señal (o grupo de señales) para instruir a un paso de operación que se debe llevar a cabo.

COORDENADAS: Posiciones o relaciones de puntos o planos. Las coordenadas se describen generalmente usando tres números que se refieren a los ejes (X, Y, Z).

CNC: Control Numérico Computarizado.

CICLO: Una secuencia de eventos o comandos.

CÓDIGO-G: El lenguaje de programación entendido por el controlador de la máquina.

AVANCE: La tasa, en mm / min o in / min en la que la herramienta de corte se hace avanzar la pieza de trabajo.

HOME: Operación para enviar los ejes de la máquina CNC a sus límites extremos de movimiento. Define el sistema de cuadrícula de coordenadas basado de la máquina CNC. Comúnmente como homing la máquina, o enviar la máquina a su posición inicial.

INTERFACE: El medio a través del cual el control / equipo dirige la máquina herramienta.

CÓDIGOS –M: Una función de código de varios en un programa CNC se utiliza para indicar una función auxiliar.

CERO MÁQUINA: A punto de referencia fijo cero fijado por el fabricante de la máquina. El cero de la máquina es utilizado para definir el sistema de la rejilla de coordenadas basado de la máquina CNC.

AVANCE RÁPIDO: Movimiento rápido de la herramienta de corte a través de la máquina en sus tres ejes.

RPM: Revoluciones por minuto (rev / min) - una medida de la velocidad del husillo.

TXT: Texto de Windows estándar solo archivo, la extensión ".txt".

INTRODUCCIÓN

La máquina CNC es una innovación tecnológica en el campo manufacturero y es una herramienta indispensable para la elaboración de piezas mecánicas de gran precisión en forma automática. Las características de este tipo de máquina permiten obtener detalles que con otros maquinados no se podrían alcanzar, conservando los altos estándares de diseño y precisión, así disminuyendo el tiempo de fabricación y elevando la productividad considerablemente. Por ésta gran importancia en la Industria, la carrera de Ingeniería Industrial impulsando los avances tecnológicos y con el propósito de reforzar los conocimientos teóricos de los estudiantes, dispone en el laboratorio de CNC con cinco máquinas de éste tipo, una de las cuales se encuentra inoperativa por fallas técnicas en su tarjeta electrónica, y es por eso que se propone la rehabilitación de éste equipo con el diseño e implementación de una tarjeta controladora he interfaz gráfica, que trabaje similar al circuito de control original.

El presente proyecto tiene como finalidad la habilitación de la maquina fresadora CNC dañada, acondicionando y elaborando los controladores necesarios para este fin, para alcanzar éste propósito se diseñará e implementará un módulo controlador o tarjetas electrónicas y software HMI que permita al estudiante interactuar y controlar directamente y de una forma fácil y sincronizada los distintos mecanismos que permiten realizar movimientos de desplazamiento lineal de la mesa de trabajo y la herramienta de corte en la máquina CNC.

Una vez realizado los objetivos planteados se realizan las pruebas de funcionamiento, verificando que todos los ejes de la máquina se muevan adecuadamente y estén acordes a los parámetros de funcionamiento.

CAPITULO 1

EL PROBLEMA

1.1. Tema

MÓDULO CONTROLADOR PARA LA MÁQUINA FRESADORA CNC DE LA FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL

1.2. Planteamiento del problema

En un principio el hombre para mejorar sus condiciones de vida en su hogar, el trabajo, y demás actividades diarias, requerían el uso de artefactos, como armas para la casa, o de pequeñas herramientas que le ayuden de una forma a reducir su esfuerzo físico, estos instrumentos eran elaborados por métodos manuales como la forja, el moldeado, y otras técnicas de herraje que requerían de mucho tiempo para su elaboración, en sí toda la actividad del hombre y en particular de su tecnología se ha basado en la utilización de herramientas creadas de forma artesanal y con máquinas totalmente manuales [1].

Con el pasar de los años la revolución industrial abrió el camino a nuevas innovaciones tecnológicas que han permitido en la actualidad la inserción al campo manufacturero de nuevas máquinas totalmente automáticas, permitiendo aumentar en número y calidad de la elaboración de piezas mecánicas o demás utensilios que requieren de un proceso para su elaboración, optimizando así eficientemente una cadena producción, dándole más flexibilidad al poder cambiar el diseño a manufacturar sin requerir de nueva reingeniería en el proceso, o tener que parar la operación de la planta por un tiempo prolongando, disminuyendo también la mano de obra directa al no requerir supervisión en cada proceso de manufactura, que para una empresa es una meta gerencial [1][2].

Una de esta clase de máquinas totalmente automáticas y como tendencia moderna de automatización en una cadena de producción manufacturera son las operadas por control numérico o conocidas también como máquinas CNC mismas que están conectadas a un ordenador central que cuenta con un software en el cual se puede programar, almacenar y transmitir información para la operación de las máquinas operadas por control numérico [3][4].

La Universidad Técnica de Ambato impulsando el avance tecnológico en el Ecuador y principalmente promoviendo el desarrollo y progreso de la tecnología por medio de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial imparte sus conocimientos a los estudiantes en materia tecnológica sobre sistemas CAD que es un Diseño Asistido por Computadora y CAM que es una Manufactura Asistida por Computadora, para ello utiliza para practica de los estudiantes una máquina fresadora CNC, la misma que por su fabricación y origen no permite tener un fácil acceso a repuestos, mantenimiento, y eventuales actualizaciones que se le debe suministrar para prolongar su vida útil.

Esta máquina fresadora CNC al momento se encuentra inactiva debido a fallas técnicas en su módulo controlador original provocadas por no haber recibido un buen mantenimiento, mala manipulación del equipo por parte de los estudiantes, o una falla en el sistema de protección de sobrecargas eléctricas, las cuales produjo que esta máquina esté al momento dañada y no pueda ser ocupada por los estudiantes para complementar prácticamente las materias de CAD y CAM que se imparten en la Facultad, es por eso que este estudio permitirá proponer un diseño alternativo de un módulo controlador elaborado con partes y elementos de fácil acceso que permita la correcta operación de la máquina y sustituir así el módulo controlador original logrando poner ésta máquina nuevamente en funcionamiento.

1.3. Delimitación

ÁREA ACADÉMICA: Electrónica

LÍNEA DE INVESTIGACIÓN: Automatización

SUB-LÍNEA: Sistemas de control automatizados e instrumentación virtual para procesos industriales de baja y alta potencia.

DELIMITACIÓN ESPACIAL: Se realizará en el Laboratorio de CNC de la Facultad de Ingeniería en Sistemas Electrónica e Industrial de la Universidad Técnica de Ambato.

DELIMITACIÓN TEMPORAL: El desarrollo del presente proyecto se realiza previo a la aprobación del perfil en un lapso de seis meses.

1.4. Justificación

La Facultad de Ingeniería en Sistemas Electrónica e Industrial requiere la habilitación de la máquina fresadora CNC para impartir conocimientos en materias de CAD y CAM aplicadas a la manufactura para el maquinado de partes y piezas de mediana complejidad, evitando así el deterioro de la misma y posterior dada de baja del inventario de la Facultad por la falta de uso.

Mediante la puesta en marcha de éste equipo los estudiantes podrían manufacturar partes y piezas de mediana complejidad para sus proyectos de diseño e ingeniería y además puedan aplicar los nuevos conocimientos en avances tecnológicos.

Esta investigación brindara un estudio completo y detallado a la persona encargada del área de automatización en la Facultad de Ingeniería en Sistemas Electrónica e Industrial, el mismo que describirá partes y componentes utilizados, diagramas de circuitos de control, manual de operaciones para mantenimiento y operaciones.

Es importante realizar este trabajo puesto que los costos de adquirir una nueva máquina rodea como mínimo los \$10000, dependiendo de las características de mecanizado, tamaño, precisión y velocidad con las que trabaja, así como de sus repuestos que no se los logra encontrar en el mercado local, teniendo que recurrir a importaciones y demás gastos que conlleva esto, además se debe aprovechar que los estudiantes de esta Facultad están capacitados y facultados para realizar una automatización completa o parcial de muchos equipos mecánicos.

Con la realización de un módulo controlador fabricado con partes y piezas de fácil acceso se podrá en un futuro realizar reparaciones en el caso de necesitar, así mismo de poder proveerle de un mantenimiento, ajustes de afinación, calibración y posibles actualizaciones que se le puedan dar para la correcta operación de la máquina fresadora CNC.

1.5. Objetivos

Objetivo general:

- Implementar un módulo controlador para la máquina Fresadora CNC de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

Objetivos Específicos:

- Identificar las causas y fallos que ocasionan la inactividad de la máquina Fresadora CNC.
- Estudiar el funcionamiento de partes electrónicas y piezas mecánicas, en el equipo.
- Diseñar circuitos electrónicos que trabajen coordinadamente con la interfaz gráfica para el control de motores los cuales permiten el desplazamiento de los distintos mecanismos de la máquina.
- Fabricar, verificar e implementar los circuitos de control en la caja eléctrica de la máquina CNC.
- Diseño y desarrollo del software del HMI para el control y supervisión del equipo CNC.
- Realizar pruebas de funcionamiento del sistema mecánico, electrónico y del HMI en la Fresadora CNC.

CAPITULO 2

MARCO TEÓRICO

2.1. Antecedentes Investigativos

En la actualidad el avance tecnológico y la gran demanda por parte de los consumidores hacen que las empresas que brindan su servicio mejoren su capacidad de producción, y para ello utilizan especialmente el mundo manufacturero las denominadas máquinas herramientas CNC es decir operadas por control numérico que es un mecanismo rápido y eficiente de manipular la flexibilidad en una cadena de producción y donde las paras de planta no se pueden dar ya que representa pérdidas en dinero y tiempo, y es por su gran importancia que deben estar disponibles y operativas en todo momento [5].

Existen empresas en el mundo que se dedican al diseño de tales equipos teniendo la característica común de que ninguna se encuentra en el Ecuador y lo más cercano que se puede estar de ellas es en Norteamérica, y en los países europeos, es por eso que de contar con una de éstas máquinas CNC el manejo y mantenimiento debe ser puntual y correcto porque de llegar a ocurrir un incidente donde el resultado final sea la avería parcial o total de la maquinaria los costos de los repuestos, importación, asistencia técnica profesional representaría un costo considerable [2][5].

El control numérico por computadora es básicamente adaptable a cualquier máquina herramienta en los cuales todos los distintos movimientos para generar trayectorias específicas de los ejes es controlado por motores, el costo y diseño de una de estas máquinas dependerán esencialmente de características como su velocidad, precisión,

tamaño, tipo de operación, capacidad de almacenamiento, flexibilidad como las más principales [5].

En nuestro país la situación tecnológica ha mejorado considerablemente pero el alto costo de estas no permite que las empresas las adquieran de manera fácil, como para agregarlas a la cadena de producción, siendo pocas empresas que las adquirido y han vinculado exitosamente a sus métodos de trabajo; una solución la promueven estudiantes de las distintas universidades del Ecuador en donde como proyectos de investigación desarrollan desde pequeños controles hasta complejos sistemas de mecanismos contralados por computadora a equipos cuya operación se la realizaba manualmente, como por ejemplo:

Para la fabricación de puertas en MDF, se diseña un Router CNC para lo cual se utiliza un mecanismo con movimientos en sus tres ejes X, Y, Z el cual ésta controlado por motores a pasos dimensionados a la potencia y fuerzas, el software que utilizan para la decodificación de los códigos G es el MACH3MILL cuyos pulsos electrónicos son transmitidos por el puerto paralelo de la PC [6].

Para el diseño de circuitos impresos en baquelita se diseña una máquina CNC controlada por motores paso a paso, con movimiento en sus tres ejes X, Y, Z, utilizando como herramienta de corte un taladro de mediana potencia y brocas de desbaste fino, para su operación y control utiliza el software LabVIEW en el cual realiza procesamiento digital de una imagen y extrae la secuencia de códigos G necesarios para el movimiento de la máquina CNC [7].

Para grabar diseños gráficos en materiales como vidrio, metal, y otros elementos que resisten el calor, se diseña una máquina CNC de corte por láser, sus movimientos están basados en motores a pasos con movimiento en sus tres ejes, utiliza como método de comunicación por puerto paralelo, y como software de control utiliza el Sapus-CNC [8].

Como este hay otros proyectos que se encaminan a la automatización de máquinas convencionales dando un grado de funcionabilidad automático capaz de trabajar sin mucho control y supervisión humana, obteniendo muchas ventajas como lo son:

- Mayor precisión y mejor calidad de productos.
- Mayor uniformidad en los productos producidos.
- Ahorro de energía y fácil mantenimiento.
- Accesibilidad a repuestos y actualizaciones.
- Fácil Manejo y control.

2.2. Fundamentación teórica

Las máquinas CNC son equipos mecánicos que para el control de sus distintos movimientos utilizan en la mayor parte motores de alto torque, sensores como finales de carrera, dispositivos encoder para determinar su posicionamiento botones de control entre otros; a continuación se describirá el principio de funcionamiento, características técnicas y forma de control de algunos componentes necesarios que permitan el desarrollo de éste proyecto.

2.2.1. SISTEMA ELECTROMECAÁNICO

Los sistemas electromecánicos consisten en la transformación de la energía eléctrica en movimiento mecánico o viceversa transformar un movimiento mecánico rotacional o lineal en señales eléctricas a continuación se describe los elementos más usados en las máquinas CNC.

2.2.1.1. Motores pasó a paso

Son componentes electromecánicos que trasforman una serie de pulsos eléctricos en pequeños desplazamientos angulares, éste motor es capaz por grados llamados denominados pasos, su giro por paso puede ser de 0.9 grados, de 1.8 grados, 2.7 grados,

éste avance depende de las características de fabricación, éste motor presenta las ventajas de tener alta precisión y repetitividad en cuanto al posicionamiento.

2.2.1.2. Tipos de motores pasó a paso

Motor a pasos Bipolar. Tienen generalmente cuatro cables de salida, este motor a pasos necesita de un driver de manejo debido a que necesita conmutar la dirección de circulación de corriente para controlar el accionamiento de las bobinas, este tipo de motor utiliza un arreglo de transistores en configuración puente H o un integrado denominado puente H. En la siguiente figura 1, indica su forma de conexión [4].

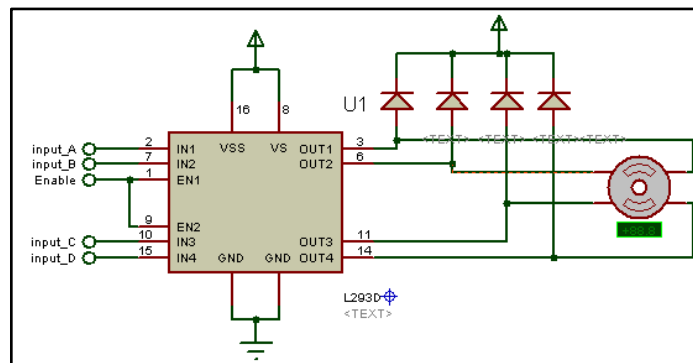


Figura 1. Driver controlador para motor bipolar
Fuente: Jesús Rodríguez

2.2.1.3. Secuencias para manejar motores paso a paso Bipolares

Como estos motores necesitan la inversión de la corriente que circula en sus bobinas en una secuencia determinada. Cada inversión de la polaridad provoca el movimiento del eje en un paso, cuyo sentido de giro está determinado por la secuencia seguida. A continuación se puede ver en la tabla 1, con la secuencia necesaria para controlar motores paso a paso del tipo Bipolar.

Tabla 1: Secuencia de activación de bobinado en motor a pasos bipolar

PASO	SECUENCIA NORMAL			
	A	B	C	D
1	+V	-V	+V	-V
2	+V	45-V	-V	+V
3	-V	+V	-V	+V
4	-V	+V	+V	-V

Fuente: Jesús Rodríguez

2.2.2. Finales de carrera

Final de carrera o sensor de contacto (también conocido como "interruptor de límite"), son dispositivos eléctricos, neumáticos o mecánicos situados al final del recorrido de un elemento móvil, como por ejemplo una cinta transportadora, con el objetivo de enviar señales que puedan modificar el estado de un circuito. Internamente pueden contener interruptores normalmente abiertos, cerrados o conmutadores dependiendo de la operación que cumplan al ser accionados, en la figura 2, se muestra un ejemplo. Generalmente estos sensores están compuestos por dos partes: un cuerpo donde se encuentran los contactos y una cabeza que detecta el movimiento. Su uso es muy diverso, empleándose, en general, en todas las máquinas que tengan un movimiento rectilíneo de ida y vuelta o sigan una trayectoria fija [4][5].

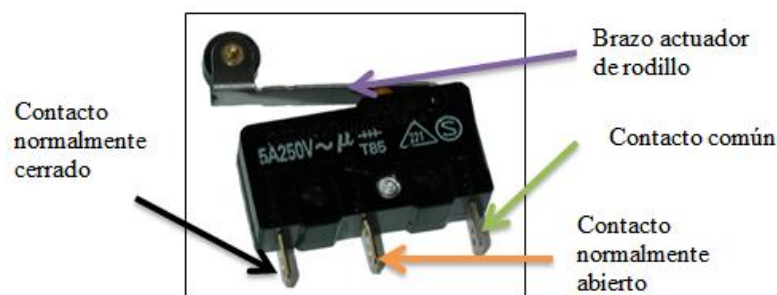


Figura 2: Final de carrera mecánico

Fuente: Jesús Rodríguez

2.2.3. SISTEMA ELECTRÓNICO

La electrónica es parte fundamental para el control de componentes mecánicos y procesos digitales, en este caso la electrónica al enviar y recibir señales electrónicas de tipo digital o análogas, permite determinar el estado en que se encuentran los sensores,

así como de controlar aplicaciones sistemas actuadores como motores, luces y otros dispositivos eléctricos.

2.2.3.1. Filtros

Los filtros anti rebote son utilizados generalmente para eliminar ruidos en las señales de entradas de los interruptores en circuitos electrónicos y ayudan absorbiendo las transiciones rápidas de los interruptores y generando señales más confiables por lo que a veces se pueden generar detecciones incorrectas de la activación de los mismos. En la siguiente figura 3, se representa el circuito básico por medio de una resistencia pull-up a 5V y un filtro pasa bajos, que consiste de la una resistencia y el capacitor. En estado estable cuando el capacitor está cargado y el interruptor no está activado se obtiene en la señal de salida 5V. Cuando se presiona el interruptor el capacitor se descarga a tierra, obteniendo en la salida un valor cercano a los 0V que es igual a baja.

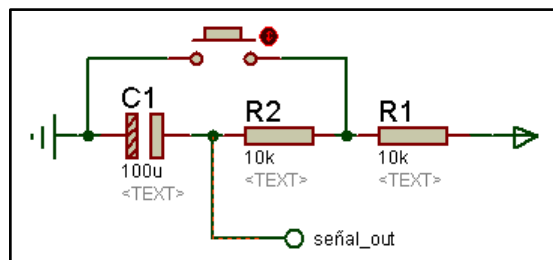


Figura 3: Diagrama de conexión anti rebote para pulsador.

Fuente: Jesús Rodríguez

2.2.3.2. Eliminación de ruidos eléctricos en micro controladores

Uno de los grandes inconvenientes de los microcontroladores como los PIC, es la sensibilidad a los ruidos eléctricos al accionar o activar cargas de potencia y recibir señales del mundo real, su funcionamiento empezará a desestabilizarse.

Cuando el microcontroladores debe accionar etapas de potencia, como relés, contactores, motores cuyo consumo de corriente sea alto, el funcionamiento de éste puede verse seriamente afectado.

2.2.3.3. Forma de reducir el ruido eléctrico

- Utilizar un condensador de Bypass, (0.1uF) entre los pines de alimentación VCC y GND del microcontrolador lo más cerca posible.
- No dejar pines sin conexión, configurarlos como salidas y conectarlos a GND.
- Si se utiliza un cristal, conectarlo a tierra por medio de condensadores que usualmente están entre los 10 y 33pf.
- Utilizar el Reset por Hardware (Resistencia y condensador).
- Para leer pines digitales como pulsadores o interruptores es necesario conectar un condensador de 0.1uF entre el pin de entrada y tierra (GND).

Para mejorar el funcionamiento de circuitos electrónicos es necesario contar con una buena fuente de alimentación bien filtrada, una forma básica de una fuente de alimentación se representa en la siguiente figura 4.

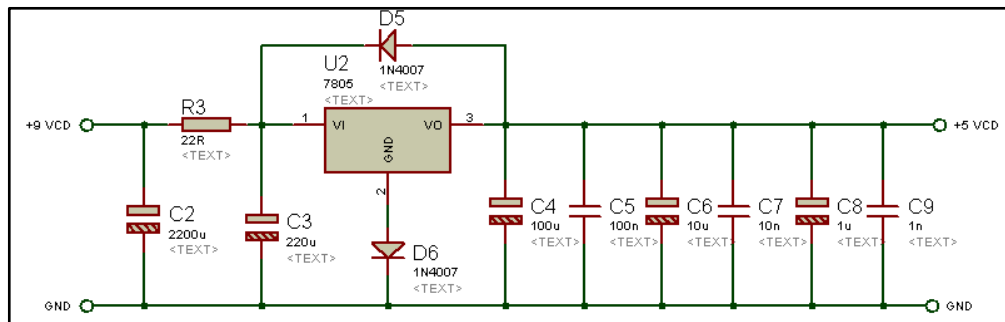


Figura 4: Diagrama de conexión para fuente de alimentación
Fuente: Jesús Rodríguez

En la entrada el condensador de 2200uF filtra el voltaje DC proveniente de la rectificación del voltaje AC. El filtro pasa-bajos RC, formado por la resistencia de 2.2 Ohmios y el condensador de 220uF, esta sintonizado a una frecuencia de 32Hz, todas las frecuencias por encima de esta, serán eliminadas, con lo cual queda el circuito protegido contra ruidos eléctrico de: Licuadoras, taladros, lámparas y de más artefactos de uso común.

El regulador de voltaje 7805 fija el voltaje de salida a un nivel de 5V, el diodo entre la salida y la entrada protege al circuito de corrientes inversas, provocadas por cargas

inductivas. El diodo en la terminal GND del 7805 provee 0.5 Vdc adicionales, en caso de que una caída súbita de voltaje, finalmente, se tiene un arreglo de condensadores electrolíticos y cerámicos con filtros RC, para asegurar un filtrado mejor y eliminar oscilaciones parasitas.

2.2.4. Microcontroladores

Es un pequeño dispositivo electrónico que permite por medio de sus puertos de entrada y salida controlar dispositivos de baja y alta potencia previamente acondicionados, internamente posee una memoria RAM y una memoria ROM, estos internamente están controlados por una unidad central de procesamiento (CPU), este dispositivo es flexible en cuanto a su aplicación pues basta con controlar por medio de un algoritmo de programación para decir que pin va a realizar determinada función, existen también dispositivos que permite adquirir señales análogas, en la siguiente figura 5, se representa la estructura de un microcontrolador.

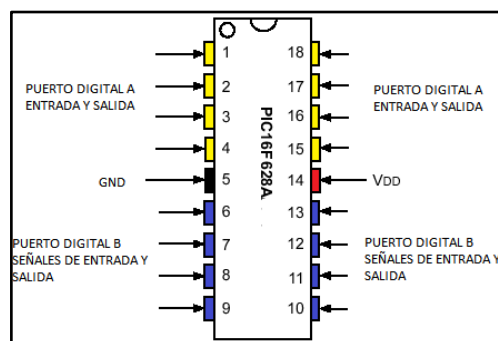


Figura 5: Dispositivo microcontrolador PIC.
Fuente: Jesús Rodríguez

Las formas de programar este microcontrolador se lo realiza por medio de software el PIC C Compiler y Microcode en los cuales se programa una serie de instrucciones que permiten gestionar los puertos de entrada y salida del microcontrolador, que a su vez puede controlar Led, Display, motores y recibir señales que proviene de sensores, pulsadores , interruptores y demás dispositivos electrónicos cuyas señales operan con niveles de 0 a 5 voltios, además permite de manera eficaz la comunicación con otros dispositivos utilizando puertos de comunicación RS232, RS485, USB, Ethernet y más protocolos.

2.2.5. COMUNICACIÓN ENTRE DISPOSITIVOS Y PC

La comunicación es el medio por el cual se intercambia información entre tarjetas electrónicas y PC con la finalidad de controlar y mantener constantemente el estado en el que se encuentran cada uno de los componentes, existen varias formas para comunicar dos dispositivos como conexiones seriales RS232, puerto paralelo, USB y Ethernet, de estos se analiza la comunicación USB por ser rápida y segura, éste puerto está presente en toda computadora.

2.2.5.1. Comunicación por puerto USB (Bus de serie universal)

El USB (Bus de serie universal), se basa en una arquitectura de tipo serial, sin embargo, es una interfaz de entrada/salida mucho más rápida que los puertos seriales estándar. La arquitectura serial se utilizó para este tipo de puerto por dos razones principales:

La arquitectura serial le brinda al usuario una velocidad de reloj mucho más alta que la interfaz paralela debido a que este tipo de interfaz no admite frecuencias demasiado altas (en la arquitectura de alta velocidad, los bits que circulan por cada hilo llegan con retraso y esto produce errores); Los cables seriales resultan mucho más económicos que los cables paralelos [9].

Existen dos tipos de conectores USB ver Figura 6, y Tabla 3:

- Los conectores conocidos como **tipo A**, cuya forma es rectangular y se utilizan, generalmente, para dispositivos que no requieren demasiado ancho de banda (como el teclado, el ratón, las cámaras Web, etc.)
- Los conectores conocidos como **tipo B** poseen una forma cuadrada y se utilizan principalmente para dispositivos de alta velocidad (discos duros externos, etc.).

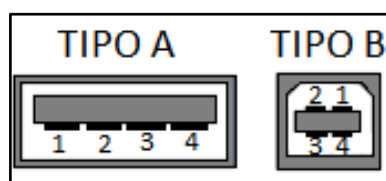


Figura 6: Conectores USB
Fuente: Jesús Rodríguez

Tabla 2: Configuración estándar de pines usb

1	Fuente de alimentación de +5 V (VBUS)
2	Datos (D-)
3	Datos (D+)
4	Conexión a tierra (GND)

Fuente: Jesús Rodríguez

La comunicación entre el host (equipo) y los dispositivos se lleva a cabo según un protocolo (lenguaje de comunicación) basado en el principio de red en anillo. Esto significa que el ancho de banda se comparte temporalmente entre todos los dispositivos conectados. El host (equipo) emite una señal para comenzar la secuencia cada un milisegundo (ms), el intervalo de tiempo durante el cual le ofrecerá simultáneamente a cada dispositivo la oportunidad de "hablar". Cuando el host desea comunicarse con un dispositivo, transmite (un paquete de datos que contiene la dirección del dispositivo cifrada en 7 bits) que designa un dispositivo, de manera tal que es el host el que decide "hablar" con los dispositivos. Si el dispositivo reconoce su dirección en la red, envía un paquete de datos (entre 8 y 255 bytes) como respuesta. De lo contrario, le pasa el paquete a los otros dispositivos conectados. Los datos que se intercambian de esta manera están cifrados conforme a la codificación NRZI (No retorno a cero invertido), ver Figura 7 [9].

Con este tipo de codificación, cuando el valor del bit es 1, la señal cambia de estado luego de que el reloj lo indica. Cuando el valor del bit es 0, la señal no cambia de estado.

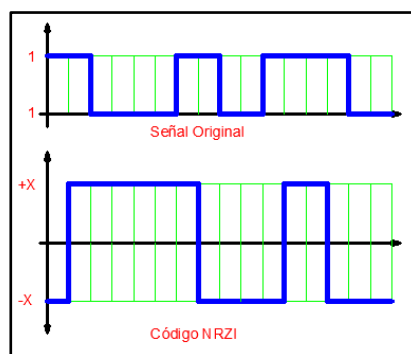


Figura 7: Codificación NRZI para comunicación USB

Fuente: Jesús Rodríguez

La codificación NRZI posee numerosas ventajas que incluyen:

- La detección de una señal o la ausencia de la misma
- La necesidad de una corriente de transmisión de baja señal

2.2.6. CÓDIGOS CNC

Los códigos CNC son instrucciones de texto conformada por letras y números éstas son secuencias numéricas estándar, y la mayor parte de máquinas CNC las utilizan. Estas instrucciones determinan una acción específica como parámetros de configuración, tipo de desplazamiento, instrucciones de movimiento, y para el control de componentes externos y otros accesorios que se deben controlar en una máquina CNC, a continuación se describe los aspectos más relevantes para un control numérico.

2.2.6.1. Códigos G

La programación para la mayoría de las máquinas de Control Numérico Computarizado se efectúa mediante un lenguaje de bajo nivel llamado G & M o (funciones generales y misceláneas), éste código se trata de un lenguaje de programación vectorial mediante el que se describen acciones simples como son segmentos de recta y arcos de circunferencia además de instrucciones de velocidades de husillo y de avance de herramienta, éste código es estándar para la mayor parte de máquinas CNC, en la siguiente tabla 4 y 5 se muestra los códigos G & M más utilizados.

Tabla 3: Códigos G Generales

Códigos G Generales	
G00	Posicionamiento rápido (sin maquinar)
G01	Interpolación lineal (maquinando)
G02	Interpolación circular (horaria)
G03	Interpolación circular (anti-horaria)
G20	Comienzo de uso de unidades imperiales (pulgadas)
G21	Comienzo de uso de unidades métricas
G28	Volver al home de la máquina
G40:	Cancelar compensación de radio de curvatura de herramienta

Fuente: Jesús Rodríguez

Tabla 4: Códigos M- Misceláneos

Códigos M- Misceláneos	
M03	Hacer girar el husillo en sentido horario
M04	Hacer girar el husillo en sentido anti-horario
M05	Frenar el husillo
M06	Cambiar de herramienta
M07	Abrir el paso del refrigerante B
M08	Abrir el paso del refrigerante A
M30	Finalizar programa y poner el puntero de ejecución en su inicio
M38	Abrir la guarda
M39	Cerrar la guarda

Fuente: Jesús Rodríguez

Para enviar un comando de código G o programa CNC, los comandos enviados deben tener un orden, sin embargo no es necesario que estén todos los ítems presentes, la estructura se presenta en la siguiente tabla 6.

Tabla 5: Estructura de un bloque de programa CNC

N	G	X	Y	Z	F	S	T	M	
									Funciones auxiliares
									Número de herramienta
									Velocidad de usillo
									Velocidad de avance
									Coordenada Z
									Coordenada Y
									Coordenada X
									Instrucción de movimiento
									Número de bloque

Fuente: Jesús Rodríguez

2.2.6.2. Movimientos de interpolación

La máquina CNC de tres ejes solo puede realizar tres movimientos, desplazamientos lineales y circulares en las dos direcciones para lograr estos desplazamientos es necesario conocer cómo y trabajan estas secuencia numéricas, a continuación se describe los movimientos principales y básicos de una CNC.

2.2.6.3. Interpolación lineal (G01)

La instrucción después de un comando G01, describe a un desplazamiento lineal de un punto inicial a otro final, la máquina se desplaza según dicha trayectoria al avance F programado. El controlador calcula los avances de cada eje para que el avance de la trayectoria resultante sea la F programada como se muestra en la siguiente figura 8.

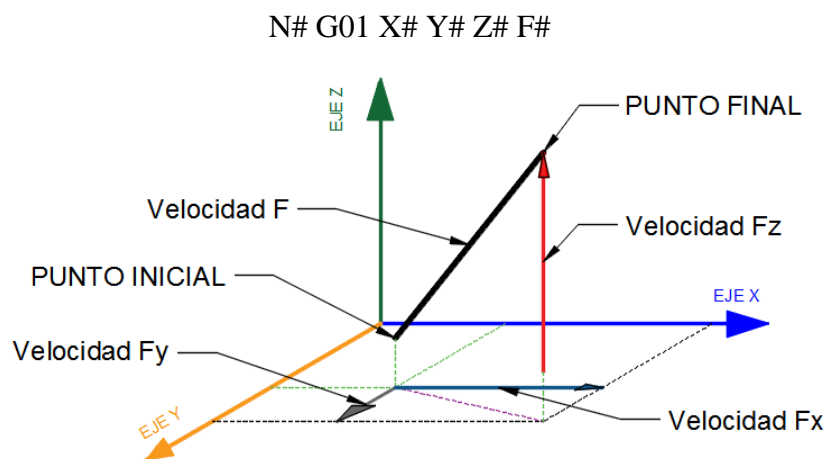


Figura 8: Representación de interpolación lineal
Fuente: Jesús Rodríguez

2.2.6.4. Movimientos circulares Instrucciones G02 y G03

Para programar una curva, lo primero a realizar es situar la máquina en el punto de inicio del arco.

N# G02/G03 X# Y# I# J#

A continuación se detalla lo que es cada cosa:

- ◆ G02/G03 Interpolación circular a derechas o a izquierdas, respectivamente.
- ◆ X Coordenada "X" del punto final del arco.
- ◆ Y Coordenada "Y" del punto final del arco.
- ◆ I Distancia desde el punto inicial del arco hasta el centro del arco en sentido del eje "X".

- ◆ J Distancia desde el punto inicial del arco hasta el centro del arco en sentido del eje "Y".

En la siguiente figura 9, se representa un ejemplo típico de interpolación circular.

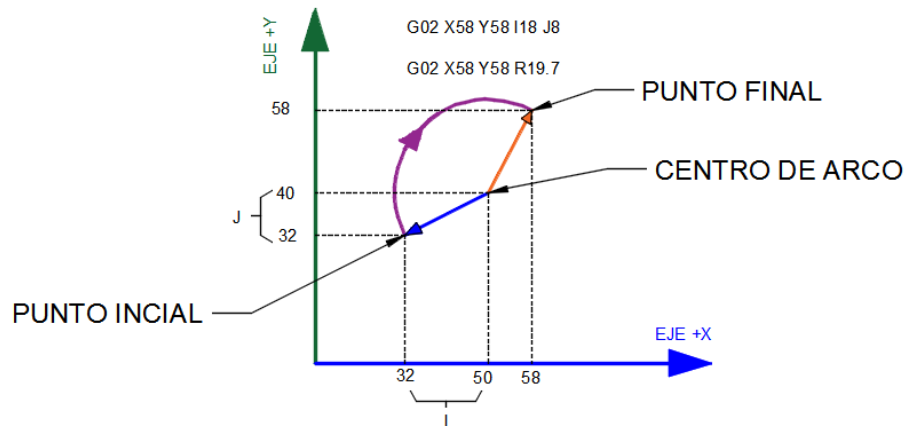


Figura 9: Gráfica de interpolación circular

Fuente: Jesús Rodríguez

2.2.7. SOFTWARE DE PROGRAMACIÓN

Para controlar toda máquina CNC es necesario un equipo de control, éste equipo de control puede ser una computadora, un touch panel o una interfaz gráfica que permita de algún modo por medio de botones físicos o virtuales controlar y supervisar las acciones del equipo en todo momento para ello nos sirve el desarrollo de la interfaz gráfica desarrollada en alguna plataforma de programación.

2.2.7.1. LabVIEW

(Acrónimo de Laboratory Virtual Instrumentation Engineering Workbench) es una plataforma y entorno de desarrollo para diseñar sistemas, con un lenguaje de programación visual gráfico. Recomendado para sistemas hardware y software de pruebas, control y diseño, simulado o real y embebido, pues acelera la productividad. El lenguaje que usa se llama lenguaje G, donde la G simboliza que es lenguaje Gráfico.

Con LabVIEW, no se programa el software a escribir líneas de códigos con una sintaxis compleja. La programación se hace con iconos los cuales representan funcionalidades, ligados entre ellos por cables quienes representan los flujos de datos (un poco a la manera de una tarjeta electrónica con sus componentes y circuitos integrados).

Esta representación muy adornada del código es cerca de la concepción como uno hace: con esquemas; eso, por supuesto, facilita mucho el trabajo que necesita la programación del concepto. Esta abstracción del lenguaje gráfico no requiere de ser un experto en programación para desarrollar software sencillo.

Los programas se dividen en dos partes bien diferenciadas, una llamada Panel Frontal, y otro Diagrama de Bloques [10].

Panel Frontal: es la interfaz con el usuario, la utilizamos para interactuar con el usuario cuando el programa se está ejecutando Ver Figura 10. En esta interfaz se definen los controles (los usamos como entradas, pueden ser botones, marcadores etc.) e indicadores (los usamos como salidas, pueden ser gráficas, etc.)

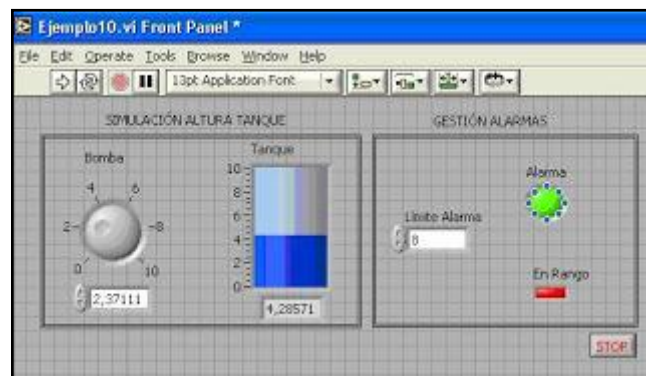


Figura 10: Panel frontal o interfaz HMI de LabVIEW
Fuente: LabVIEW entorno grafico de programación [10].

Diagrama de Bloques: es el programa propiamente dicho, donde se define su funcionalidad, aquí se colocan íconos que realizan una determinada función y se interconectan (el código que controla el programa), Ver Figura 11. Suele haber una tercera parte icono/conector que son los medios utilizados para conectar un VI con otros VIs.

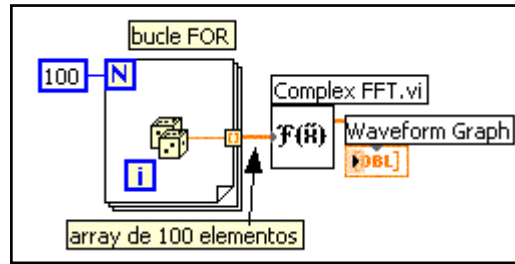


Figura 11: Diagrama de bloques para la programación en LabVIEW
Fuente: LabVIEW entorno gráfico de programación [10].

2.2.7.2. Interfaz HMI

La sigla HMI es la abreviación en inglés de Interfaz Hombre Máquina. Los sistemas HMI podemos pensarlos como una “ventana” de un proceso. Esta ventana puede estar en dispositivos especiales como paneles de operador o en una computadora. Los sistemas HMI en computadoras se los conoce también como software HMI o de monitoreo y control de supervisión. Las señales del procesos son conducidas al HMI por medio de dispositivos como tarjetas de entrada/salida en la computadora, también pueden ser de un PLC’s (Controladores lógicos programables) o de RTU (Unidades remotas de I/O) [10].

2.2.7.3. Funciones de un Software HMI

- **Monitoreo.-** Es la habilidad de obtener y mostrar datos de la planta en tiempo real. Estos datos se pueden mostrar como números, texto o gráficos que permitan una lectura más fácil de interpretar.
- **Supervisión.-** Esta función permite junto con el monitoreo la posibilidad de ajustar las condiciones de trabajo del proceso directamente desde la computadora.
- **Alarmas.-** Es la capacidad de reconocer eventos excepcionales dentro del proceso y reportarlo estos eventos. Las alarmas son reportadas basadas en límites de control pre- establecidos.

- Control.- Es la capacidad de aplicar algoritmos que ajustan los valores del proceso y así mantener estos valores dentro de ciertos límites. Control va más allá del control de supervisión removiendo la necesidad de la interacción humana.- Sin embargo la aplicación de esta función desde un software corriendo en una PC puede quedar limitada por la confiabilidad que quiera obtenerse del sistema.
- Históricos.- Es la capacidad de mostrar y almacenar en archivos, datos del proceso a una determinada frecuencia. Este almacenamiento de datos es una poderosa herramienta para la optimización y corrección de procesos.

Tareas de un Software de Supervisión y Control

- Permitir una comunicación con dispositivos de campo.
- Actualizar una base de datos “dinámica” con las variables del proceso.
- Visualizar las variables mediante pantallas con objetos animados (mímicos).
- Permitir que el operador pueda enviar señales al proceso, mediante botones, controles ON/OFF, ajustes continuos con el mouse o teclado.
- Supervisar niveles de alarma y alertar/actuar en caso de que las variables excedan los límites normales.
- Almacenar los valores de las variables para análisis estadístico y/o control.

Para complementar la interfaz gráfica o software de control es necesario conocer los dispositivos que permitan adquirir datos o señales electrónicas, para ello es necesario conocer cómo funcionan estas tarjetas de control y supervisión de señales eléctricas.

2.2.7.4. Dispositivos de entrada y salida DAQ (adquisición de datos)

Los sistemas de adquisición de datos nos ayudan a medir información presentada en forma digital o analógica, estas señales digitales pueden venir de una variedad de fuentes tales como: interruptores, relevadores, interfaces compatibles con niveles TTL,

etc.[11],[10].Las señales analógicas vienen de diferentes instrumentos, como pueden ser sensores o transductores que convierten energía en forma de presión, posición o temperatura en voltaje. La adquisición de datos o adquisición de señales, consiste en la toma de muestras del mundo real (sistema analógico) para generar datos que puedan ser manipulados por un ordenador, consiste, en tomar un conjunto de señales físicas, convertirlas en tensiones eléctricas y digitalizarlas de manera que se puedan procesar en una computadora ver Figura 12.

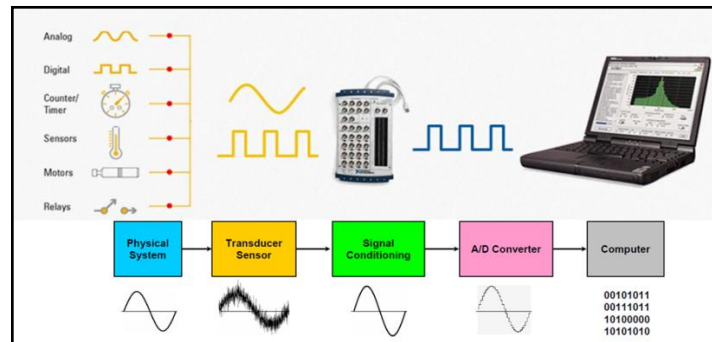


Figura 12: Secuencia para la adquisición de señales.

Fuente: LabVIEW entorno grafico de programación [10].

Las tarjetas de adquisición de datos (DAQ) pueden ser como las siguientes tarjetas independientemente o contar con combinaciones de estas:

Convertidor Analógico-Digital (A/C).- Convierte señales analógicas a una forma de señal digital equivalente. Los rangos de tensión de entrada comúnmente utilizados son: 10V, -5V y 5V, 0 a 5V, 0 a 10V, también hay tarjetas A/D que miden corrientes entre 4 a 20 mA.

Convertidor Digital-Analógico (D/A).- Este tipo de sistema convierten una señal digital dada por la computadora en una señal analógica. Los rangos normales de salida de tensión que otorgan estas tarjetas son de 5V, 10V, 0 a 5V y de 0 a 10V, también hay tarjetas que generan corriente de 4 a 20 mA.

Digital Input/Output (D/IO).-Tarjetas de entradas y salidas digitales. Mediante estas tarjetas se pueden accionar todo lo que implique cambio entre dos estados. Por lo general se tiene un nivel bajo de 0 a 0.8 V, y un nivel alto de 2 a 5 V, dependiendo de cada fabricante

2.3. Propuesta de Solución

Con la implementación del módulo controlador se conseguirá la habilitación y funcionamiento de la máquina Fresadora CNC que posee la Facultad de Ingeniería en Sistemas Electrónica e Industrial, la misma que estará elaborada con componentes electrónicos genéricos de bajo costo, y accesibles en el mercado local, éste equipo podrá ser operado en baja y alta tensión, se implementara circuitos de protección para garantizar daños por fallos eléctricos; con esta máquina nuevamente funcionando los estudiantes podrán complementar sus clases teórico prácticas en materias de CAD y CAM.

CAPITULO 3

METODOLOGÍA

El presente proyecto es una investigación de tipo aplicada, por lo que se requiere de información de libros con temas de control de mecanismo, funcionamiento de motores, desplazamientos de mecanismo móviles, tipos de maquinado, instrucciones de control numérico y otros tipo de catálogos de dispositivos electrónicos y mecánicos que permitan llegar a los objetivos planteados; así como de sugerencias tomadas del personal responsable de la máquina CNC.

3.1. Modalidad de la investigación

De campo

El proyecto se desarrolló directamente en el Laboratorio CNC, mediante la supervisión de los encargados del área, pudiendo así observar las instalaciones y las condiciones físicas de la máquina fresadora CNC, así como también el estado actual del módulo controlador que impide la operación de dicho equipo. De esta forma se pudo visualizar directamente el problema que presenta la Fresadora CNC.

Bibliográfica

Se sustentó la base teórica en la investigación en fuentes bibliográficas de tipo técnico, catálogos de dispositivos, revistas de innovaciones tecnológicas, apuntes de operario, documentos varios originales de la misma máquina CNC, y en artículos de investigación que permitieron buscar alternativas de solución válidas para nuestro propósito.

Aplicada

La investigación tubo como característica el diseño y la implementación, brindando así a la máquina CNC un funcionamiento estable, que cumpla con los requerimientos y estándares establecidos originalmente en éste equipo, lo que complementara la enseñanza teórico práctico dentro de la Facultad por parte de los estudiantes.

3.2. Población y muestra

Por las características de inoperatividad que presenta la máquina CNC no se requiere de un análisis de población y muestra.

3.3. Recolección de la información

Se recepto los datos técnicos como características de motores, voltajes de operación, tipos de mecanismos que la conforman, dispositivos de control mecánico y eléctrico mediante el análisis a la Fresadora CNC, los mismos que se anotarán en una matriz de características y parámetros de funcionamiento que fueron esenciales para diseñar los circuitos de control.

3.4. Procesamiento y análisis de datos

Con la matriz de características y parámetros de funcionamiento se determinó el tipo de control que los mecanismos requieren para su funcionamiento, permitiendo realizar circuitos para proteger tanto el sistema mecánico como el sistema eléctrico, y demás seguridades que ayudan tanto al equipo como al usuario al momento de que estén en operación.

3.5. Desarrollo del proyecto

- Análisis de la máquina fresador CNC mediante la supervisión del personal a cargo del área de Automatización.

- Recolección de especificaciones de partes y componentes que conforman la máquina CNC.
- Estudio del funcionamiento de partes electrónicas y piezas mecánicas, necesarias para el control de los mecanismos.
- Especificaciones de los dispositivos necesarios para habilitar la máquina CNC.
- Selección del dispositivo apropiados para el controlador.
- Elaboración de los planos eléctricos del sistema a automatizarse.
- Realización de circuitos y diagramas de control para la máquina CNC.
- Selección del protocolo de comunicación para el intercambio de datos entre la máquina y el controlador.
- Diseño del software de control que permita la interacción del usuario y la máquina en cuestión.
- Evaluación de funcionamiento del sistema previo a su implementación.
- Implementación del controlador y desarrollo del manual de usuario.

CAPITULO 4

DESARROLLO DE LA PROPUESTA

4.1. CONDICIONES FÍSICAS EN LAS QUE SE ENCUENTRA INICIALMENTE LA MÁQUINA CNC

La máquina Fresadora CNC es tipo NOVAMILL de la serie E00284, con caja eléctrica modelo 5101BR000262 de la Marca Denford, que tiene un total de tres ejes, ésta es una máquina de entrenamiento, capacitación y educación, el cual simula un ambiente de trabajo industrial real en pequeña escala de forma que el operario pueda trabajar de forma segura y fácil.

Mesa de trabajo

La mesa de trabajo está fabricada en hierro las dimensiones que tiene es de 360 mm X 130 mm (14" x 5"), la misma que tiene canales ranurados en los cuales se coloca los soportes de sujeción para la pieza a mecanizar o tallar.

Desplazamiento

La mesa de trabajo puede desplazarse desde el cero máquina a su límite mecánico, 225 mm para el eje X, 150 mm para el eje Y, y para el eje Z una distancia de 115 mm.

Husillo

El husillo es de ajuste manual en él se puede utilizar fresas de carácter estándar según la norma ISO 30, las fresas de vástago cilíndricas o cónicas posibles a utilizar son de 1 mm a 12 mm efectivo de maquinado o 1/16" a 1/2".

Objeto de trabajo

Las dimensiones máximas permisibles de trabajo en el plano X, Y, Z, están limitadas a 225 mm en el eje X, 150 mm en el eje Y, y para el eje Z dispone de un rango de trabajo entre 1 mm a 100 mm. De la cual se recomienda utilizar una dimensión menor a los límites permisibles de trabajo por seguridad.

Los materiales con los cuales la máquina trabaja pueden ser cera, plásticos, acrílicos, cobre, aluminio, latón y acero suave, madera. Se recomienda utilizar bloques de madera MDF de 120 mm x 120 mm.



Datos eléctricos



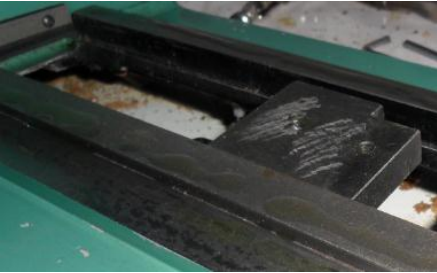


La máquina CNC trabaja a 220 voltios monofásica de CA, con una frecuencia de 50/60 Hz, y un mínimo de 8 amperios.

En la tabla 7, se enuncia las características actuales de la Fresadora CNC.

4.2. CARACTERÍSTICAS TÉCNICAS Y OPERATIVAS DE LA MÁQUINA CNC:

Tabla 6: Análisis de elementos

Descripción	Muestra	Especificaciones
La interfaz Gráfica en la PC cumple con las condiciones normales para el correcto funcionamiento.		Pc de escritorio. Memoria RAM 512 Disco Duro 1Gb
Los motores a pasos se encuentran en óptimas condiciones, responden favorablemente a los pulsos de control.		Unipolar Voltaje: 30VCD Amperaje: 1-2A Pulsos por revolución:200

<p>El motor principal para giro de herramienta responde a la señal de control</p>		<p>Velocidad variable Potencia: 1 Hp Voltaje: 180 VCD RPM: 0 a 3000</p>
<p>Tarjeta de control no transmite información ni controla el desplazamiento de los mecanismos de la CNC.</p>		<p>Voltaje de alimentación: 28 VCA Input: 24 digitales, 2 análogos Output: 24 digitales y 1 análoga Comunicación: RS232 Alimentación para motores: 30 VCD</p>
<p>Los rieles de desplazamiento se encuentran en perfectas condiciones para el recorrido de los diferentes mecanismos.</p>		<p>Material: hierro Lubricación: por inyección Riel para cada movimiento.</p>
<p>La fuente de alimentación no entrega el voltaje necesario para el funcionamiento de los actuadores.</p>		<p>Voltajes de funcionamiento: +5 VCD +12 VCD -12 VCD</p>
<p>Comunicación realiza comunicación con la PC.</p>		<p>Comunicación: RS232 Memoria eeprom Display de errores</p>

<p>Los finales de carrera se encuentran operativos.</p>		<p>Normalmente cerrados. Tipo: vástago pulsante</p>
<p>Paro de Emergencia se encuentra en óptimas condiciones.</p>		<p>Normalmente cerrado</p>
<p>Potenciómetros de avance regula correctamente el avance de maquinado.</p>		<p>Entrega: 0 a 5 VCD variable</p>
<p>Potenciómetro de control de velocidad del husillo</p>		<p>Entrega: 0 a 5 VCD variable</p>
<p>Guarda puerta recibe perfectamente el pulso de apertura o cierre de la puerta.</p>		<p>Solenoides de 110 VCA Contactos: 2 normalmente abiertos. 2 normalmente cerrados.</p>
<p>Sistema de transmisión de movimiento para los tres ejes X, Y, Z. Transmiten correctamente el movimiento a su respectivo eje.</p>		<p>Sistema de transmisión para reducción de velocidad y multiplicación de torque. Banda dentada sincrónica</p>
<p>Tornillo sin fin del eje X, Y, Z, desplaza correctamente los mecanismos</p>		<p>Este tornillo sin fin tiene un diámetro efectivo de 16 mm con 5 mm (0,2") de paso, con una</p>

correspondientes de cada eje como mesa y husillo	resolución 0.01 mm. Debe estar lubricado constantemente.
El conector se encuentra transmitiendo información correctamente desde la máquina hasta el control eléctrico.	Tipo macho de 42 pines de potencia tipo 357F Con seguro de conexión.



Fuente: Jesús Rodríguez

4.3. COMPONENTES QUE CONTIENE LA MÁQUINA CNC

La máquina tiene un conector tipo macho de 42 pines como se muestra en la figura 13, por inspección se detalla cada uno de los pines corresponde a un componente interno de la máquina CNC la tabla 8, ver Anexo 6, se describe a continuación.



Figura 13: Conector macho tipo 357F de máquina CNC

Fuente: Jesús Rodríguez

Tabla 7: Descripción del conector tipo 357F

Número de PIN	Descripción
1,2,3,4	Motor eje X
5,6,7,8	Motor eje Y
9,10,11,12	Motor eje Z
13,14	Final de carrera eje X
15,16	Final de carrera eje Y

17,18	Final de carrera eje Z
19,20,21	No se utilizan
22,23,24	Motor del husillo
25,26,27	No se utiliza
28	Entrada de 24V
29,30	Stop
31,32,33,34	Contacto de chapa
35,36	Solenoide de chapa
37	Entrada de 5V
38	0V - GND
39,40	No se utilizan
41	Potenciómetro de avance rápido
42	Potenciómetro de velocidad husillo

Fuente: Jesús Rodríguez

4.3.1. MOTORES

4.3.1.1. Motor eje X y eje Y.

El motor paso a paso marca SANYO DENKI que controla los ejes X, Y cuenta con las siguientes características que se detallan en la siguiente tabla 9.

Tabla 8: Características técnicas de los motores x, y.

Descripción	Características
Angulo básico de paso	$1.8^{\circ} \pm 0.9^{\circ}$
Voltaje de bobina	3.6 VCD
Amperaje de bobina	1.4 A
Resistencia de bobina	0.28Ω
Torque de retención	62 Ncm
Aceleración teórica	$17900 \text{ rad} \cdot \text{sec}^2$
Masa	0.54 Kg

Fuente: Jesús Rodríguez

Cada uno de estos motores a paso a paso tiene cables de colores, los cuales por medición de la resistencia eléctrica en sus bobinas se determina su configuración en la siguiente tabla 10.

Tabla 9: Código de Colores de las bobinas motor paso a paso

Bobina	Colores de cable
Bobina A	azul, naranja
Bobina B	rojo, amarillo

Fuente: Jesús Rodríguez

La secuencia de energizado para estos motores a pasos es azul, naranja, rojo, amarillo las cuales servirá para conectarlos al driver de potencia.

4.3.1.2. Motor eje Z

Este es un motor paso a paso marca SANYO DENKI cuenta con las siguientes características que se detallan en la siguiente tabla 11.

Tabla 10: Características técnicas de los motores z

Descripción	Características
Angulo básico de paso	$1.8^{\circ} \pm 0.9^{\circ}$
Voltaje de bobina	2.9 VCD
Amperaje de bobina	3.05 A
Resistencia de bobina	0.28Ω
Torque de retención	170 Ncm
Aceleración teórica	$30000 \text{ rad} \cdot \text{sec}^2$
Masa	1.4 Kg

Fuente: Jesús Rodríguez

Este motor paso a paso tiene cables de colores, los cuales por medición de la resistencia eléctrica en sus bobinas se detalla en la siguiente tabla 12.

Tabla 11: Código de colores de las bobinas motor paso a paso.

Bobina	Colores de cable
Bobina A	azul, naranja
Bobina B	rojo, amarillo

Fuente: Jesús Rodríguez

La secuencia de energizado para estos motores a pasos es azul, naranja, rojo, amarillo las cuales servirá para conectarlos al driver de potencia.

4.3.1.3. Motor Husillo

El motor del husillo es de la marca BALDOR INDUSTRIAL, con una potencia de ½ HP, voltaje de alimentación 180 VCD, amperaje de 2.4 A, revoluciones por minuto de 5000 RPM, temperatura ambiente 40°C. Éste motor cuenta en la caja eléctrica un driver que trabaja a 220V de corriente alterna éste la rectifica y controla por PWM la velocidad de giro a la cual se desea que trabaje el husillo, usando para ello el driver una entrada de voltaje variable de 0 a 10 V de corriente directa.

4.3.2. CONTACTOS:

- **Finales de carrera**

La máquina CNC cuenta con tres sensores de contacto (también conocido como "interruptor de límite"), estos están ubicados en los extremos de los ejes (-X), (+Y), (+Z) de acuerdo a la nomenclatura del equipo, se los utiliza como finales de carrera y para cuando la máquina realice un ciclo home y pueda coger nuevamente con precisión el cero máquina. Estos tienen el objetivo de enviar señales que puedan modificar el estado de un circuito e indicar y detener un ciclo de maquinado para evitar que la mesa de trabajo se desplace más allá de lo permitido y ocasione daños. Este dispositivo internamente tiene configuraciones de contactos normalmente abiertos y cerrados o conmutadores dependiendo de la operación que cumplan al ser accionados. Por defecto en éste equipo los contactos están normalmente cerrados y únicamente se conmutan a abiertos

cuando la mesa de trabajo llega al cero máquina, en la figura 14, muestra el sensor instalado en la máquina.



Figura 14: Final de Carrera
Fuente: Jesús Rodríguez

- **Pulsador paro de emergencia**

El pulsador de emergencia al ser presionado abre sus contactos normalmente cerrados lo cual permite cortar el flujo de energía al contactor que se ubica en la caja eléctrica y controla que se encienda o no el driver del que hace que gire el husillo y el apaga los driver de los motores a pasos, en la figura 15, ver Anexo 8, se muestra el pulsador de emergencia.

Este normalmente se activa para cuando se produzca una emergencia como:

- Desplazamiento de la herramienta de corte no admitido en el código.
- Posibles colisiones entre el husillo y la pieza o mesa de trabajo.
- Falla en el código.
- Cancelar una operación.
- Para abrir la puerta en una emergencia.

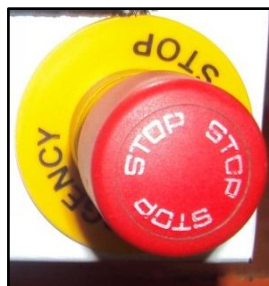


Figura 15: Paro de emergencia.
Fuente: Jesús Rodríguez

4.3.3. CERRADURA DE PUERTA

El mecanismo de cierre de la chapa eléctrica comandada por la caja eléctrica, abre y cierra el seguro de la puerta, en un nivel industrial es obligatorio debido a que el operario puede realizar una operación incorrecta cuando la máquina este encendida o trabajando y ocasionar daños físicos tanto al operador como personas ajenas al puesto de trabajo y a la misma máquina.



Figura 16: Cerradura electromagnética.

Fuente: Jesús Rodríguez

El mecanismo de cierre de la puerta tiene una solenoide que trabaja a un voltaje de 110 Voltios de corriente alterna, adicionalmente tiene contactos normalmente abiertos y cerrados cuyo estado dependen de la posición de la puerta es decir si la puerta está abierta o cerrada, con la cual permite maquinar solo si la puerta está cerrada como medida de seguridad, en la figura 16, se observa el seguro colocado en la parte lateral de la máquina CNC.

4.3.4. POTENCIÓMETROS

Los potenciómetros se energizan con 5 Voltios de corriente directa y funciona como divisores de voltaje logrando tener un voltaje variable de 0 a 5 voltios, en la figura 17, se detalla los potenciómetros que dispone la máquina.

El Spindle override permite aumentar la velocidad de avance de maquinado de 0 a 100%. El Feedrate override similar al anterior permite aumentar la velocidad de la RPM del husillo.



Figura 17: Potenciómetros de la maquina CNC.
Fuente: Jesús Rodríguez

4.4. FUNCIONAMIENTO DE PARTES MECÁNICAS Y CONTROL DE LOS MECANISMOS.

4.4.1. Tornillo sin fin

El tornillo sin fin es de tipo Ballcrews es decir cuenta con un sistema de rodamiento el cual ayuda a disminuir la fricción entre el tornillo y la mesa de trabajo al realizar un desplazamiento en el caso de los ejes X, Y, y para el eje Z al desplazar el husillo.

Este tronillo sin fin tiene un diámetro efectivo de 16 mm con 5 mm (0,2") de paso, con una resolución 0.01 mm. Éste debe estar lubricado constantemente.

4.4.2. Transmision de polea

- **Transmisición para el eje X, Y, Z:**La transmision de movimiento en los tres ejes es la misma, el sistema mecánico esta formado por un generador de movimiento en este caso es un motor paso a paso, una banda sincrónica y dos poleas dentadas en una relacion de multiplicacion de fuerza y reducion de velocidad. El tornillo sin fin tiene 5mm de paso de rosca por cada revolución lo que permite desplasar en los ejes X,Y, la mesa de trabajo y en ele eje Z el husillo la misma distancia. El motor a pasos tiene acoplado una polea dentada de 12 dientes, y en el eje del tornillo ésta acoplado la polea dentada de 30 dientes, el sistema esta formado como se representa en la siguiente figura 18.

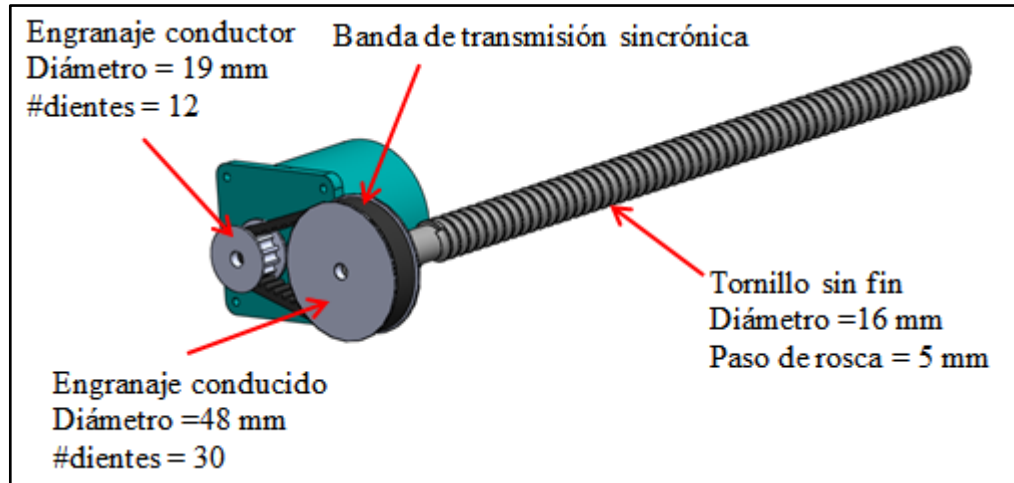


Figura 18: Sistema de transmisión mecánico
Fuente: Jesús Rodríguez

Fórmula de transmisión para los ejes X, Y, Z

Datos:

1 Revolución_tornillo = 5 (mm)

1 Revolución_motor = 200 (pasos)

$$Rev_{tor} = \frac{1}{5} Rev/mm \quad ; \quad z1 = 12 \quad ; \quad z2 = 30$$

Ecuación de transmisión de movimiento.

$$Rev_{mot} * z1 = Rev_{tor} * z2 \quad (1)$$

$$Rev_{mot} = \frac{Rev_{tor} * z2}{z1}$$

$$Rev_{mot} = \frac{\frac{1}{5} Rev/mm * 30}{12} = 0.5 Rev/mm$$

$$Rev_{mot} = \frac{0.5 Rev}{mm} * \frac{200 pasos}{1 Rev} = 100 pasos/mm$$

Fórmula para la resolución de la máquina

El cálculo de la resolución de la máquina fresadora CNC por cada eje se calcula con la siguiente fórmula.

$$\text{Resolución} = \frac{p}{ppu} \quad (2)$$

Donde:

p= desplazamiento metrico del tornillo

ppu=paso por unidad metrica de desplazamiento

$$\text{Resolución} = \frac{1mm}{100 \text{ pasos}} = 0.01 \text{ mm/pasos}$$

Con éstos cálculos se determina que para desplazar la mesa de trabajo o el husillo 1 mm en cualquier dirección, el motor debe dar 100 pasos, con eso se tiene una resolución de 0.01 mm/paso lo que es muy bueno para obener trayectorias precisas.

4.5. DESCRIPCIÓN Y FUNCIONAMIENTO DE PARTES ELECTRÓNICAS PARA CONTROL DE LOS MECANISMOS

En la figura 19, muestra la caja eléctrica la cual permite el funcionamiento de la máquina fresadora CNC, en esta caja se encuentran breakers, borneras, conectores, fuente de alimentación, cajas fusibles, drivers de motores, circuito controlador del sistema, transformadores, y demas sistemas de protección que funcionan coordinadamente con su interfaz grafica en la PC.

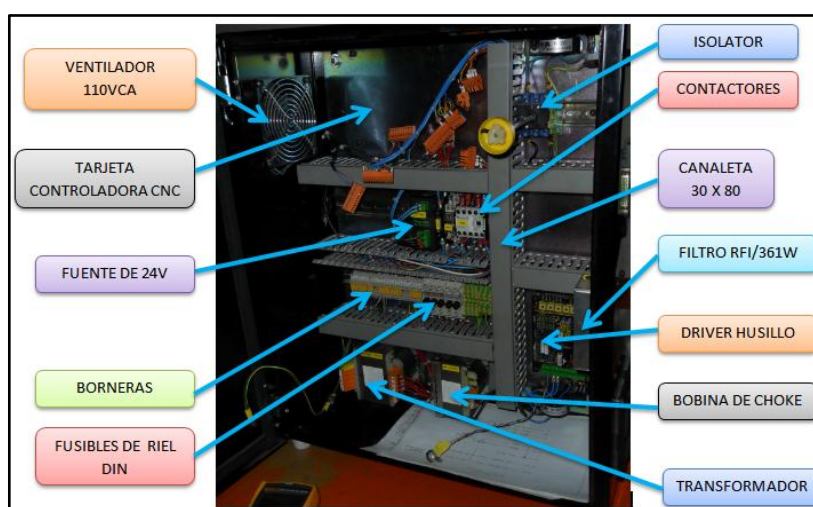


Figura 19: Caja Eléctrica
Fuente: Jesús Rodríguez

4.5.1. Isolator

Conocido también como interruptor de aislamiento de circuitos eléctricos, es un conmutador de perilla utilizado para aislar una sección o todo el sistema de un circuito. En la máquina CNC éste isolator no está diseñado para romper las corrientes de carga o para romper las corrientes de falla. Al accionar la perilla permite energizar con 220 VCA monofásica a todo el sistema eléctrico.

Los terminales NN1 y LL1 son los cables de conexión que viene de la red eléctrica principal y pasan a través de un anillo de ferrita con el cual ayuda a disminuir el ruido eléctrico de la red, en la siguiente figura 20, se muestra el diagrama eléctrico y la disposición real del dispositivo.

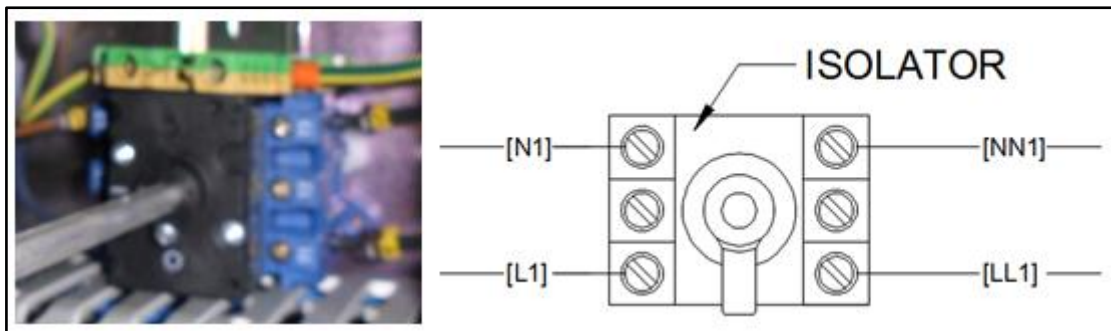


Figura 20: Isolator
Fuente: Jesús Rodríguez

4.5.2. Filtro IRF

Controla la interferencia electromagnética que es una perturbación que ocurre en cualquier circuito, componente o sistema electrónico causada por una fuente de radiación electromagnética externa al mismo, la cual puede interrumpir, degradar o limitar el rendimiento del sistema, en la siguiente figura 21, se muestra el diagrama eléctrico y la disposición real del dispositivo.

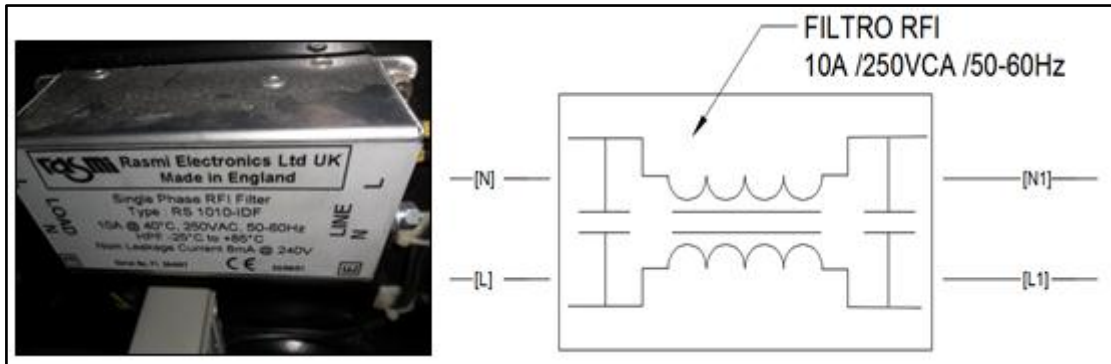


Figura 21: Filtro RFI
Fuente: Jesús Rodríguez

A continuación del aislador, el voltaje de alimentación procedente de las líneas N1, L1 debe ser filtrada de la mayor parte de ruido eléctrica por parte del filtro IRF obteniendo una tensión de bajo ruido electromagnético en las terminales N y L.

4.5.3. Driver de husillo (Spindle)

La tarjeta activa y desactiva la rotación del husillo, trabaja con un voltaje de alimentación de 220 VCA siempre y cuando el rele ESR este activado este voltaje se conecta a las borneras del driver 12 y 13.

En las terminales 8 y 9 corresponden a la bobina del motor del SPINDLE en esta línea cuenta con un anillo de ferrita para disminuir el ruido eléctrico y una bobina para que resista el choque eléctrico producido por la corriente que se podrían generar. Con los pines 4 y 5 le corresponden un contacto normalmente abierto, que al ser activado por la conmutación del relay SGR enciende el driver del husillo.

La velocidad de revoluciones por minuto, los pines 2 y 3 reciben una señal analógica de 0 a 10 VCD, el voltaje de ajuste es directamente proporcional a las revoluciones del husillo. La tarjeta dispone de un relay que se acciona cuando el driver está encendido y dicha señal indicará al controlador que está en uso, por medio del relay GIR, en la siguiente figura 22, y el Anexo 5, se muestra el diagrama eléctrico y la disposición real del dispositivo.

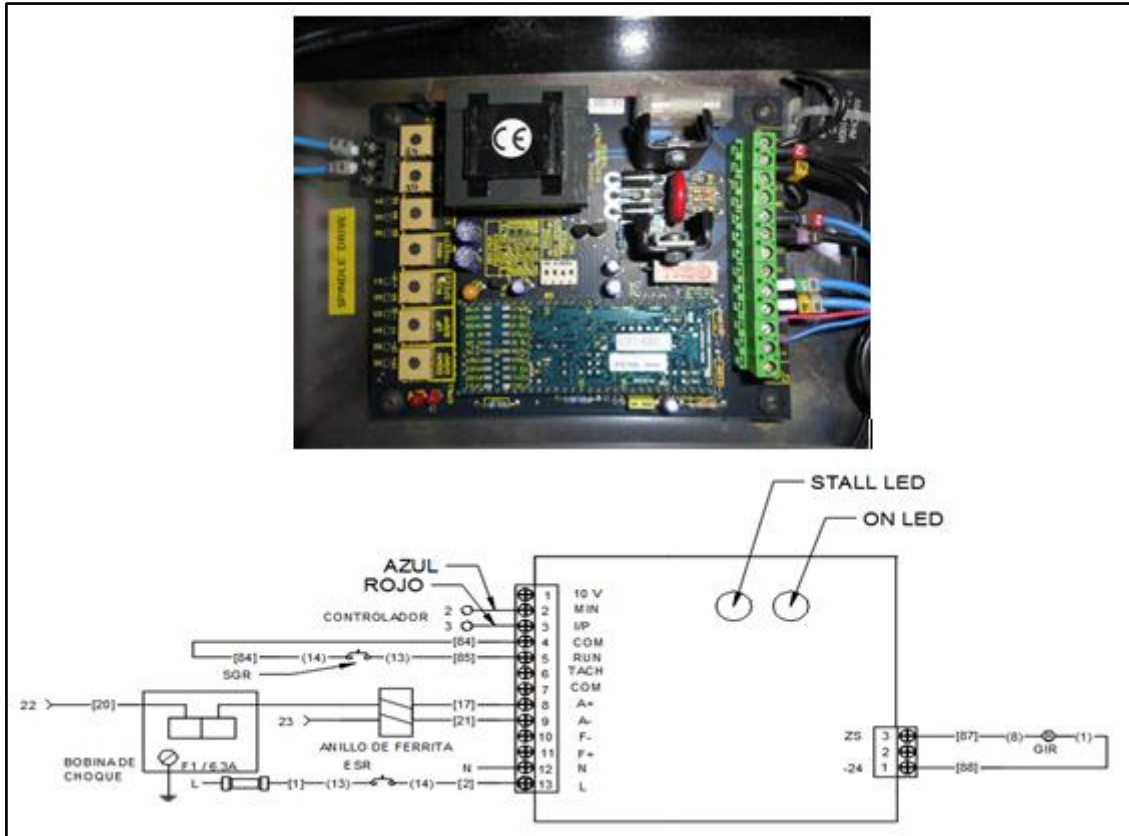


Figura 22: Driver husillo
Fuente: Jesús Rodríguez

4.5.4. Transformador CA/CA

El transformador reduce el voltaje alterno de la red principal de 220 VCA a tres voltajes menores dos de ellos son de 28 VCA y uno de 110 VCA. Las terminales 5 y 6 es el voltaje de 110 VCA para ativar la selenoide de apertura del guarda puerta, el ventilador y una fuente de poder de 24 VCD de 2.5 A de salida.

Las terminales de salida 1 y 2 van hacia la tarjeta controladora y alimentan a los dirver de los motores paso a paso de los ejes X, Y, Z.

Las terminales de salida 3 y 4 de la misma forma alimentan a los circuitos lógicos. En la figura 23, anterior se muestra el diagrama eléctrico del dispositivo.

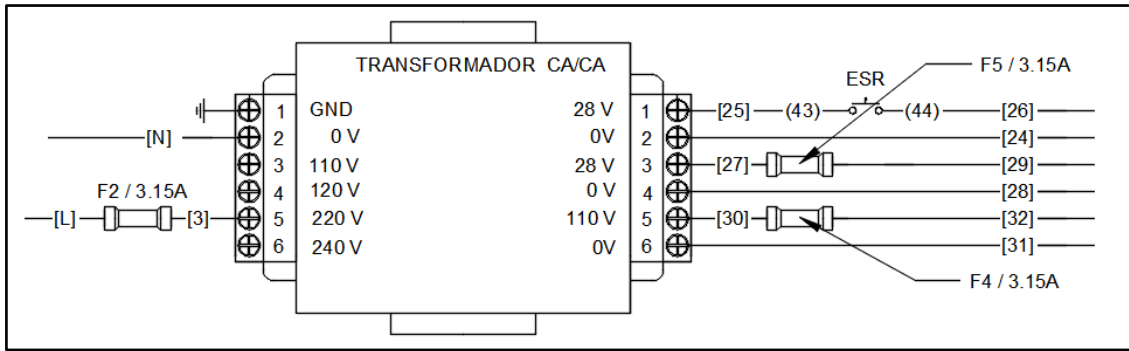


Figura 23: Transformador.
Fuente: Jesús Rodríguez

4.5.5. Controlador de máquina CNC.

La placa o driver original tiene 12 socalos de conexión enumerados como TB1 a TB12 cada uno de ellos controla los distintos componentes de la máquina CNC como motores, contactores, señales de entrada y salida, a ésta tarjeta le llega información del software original, las señales son codigos transmitidos por comunicación serial los cuales son comandos e instrucciones que gestionan los distintos movimientos de la máquina CNC, la figura 24, muestra la placa original la misma que se encuentra fuera de funcionamiento.



Figura 24: Tarjeta controladora
Fuente: Jesús Rodríguez

Socalo 1 (TB1): Tiene 12 terminales enumerados del 1 al 12 la tabla 13, indica la descripción de pines, por análisis e inspección eléctrica se determina que son pines

digitales de tipo entrada, que en condiciones normales estos interruptores tanto de finales de carrera como de emergencia estan cerrados con GND.

Tabla 12: Descripción de pines de tarjeta

TB1 Socalo de Limites			
# Bornera	Nombre	Conectado	Descripción
1	Chasis	Si	Gnd del cable blindado
2	Gnd	Si	Señal de tierra o 0V
3	Limit 0	No	XXXXXXXXXXXXXXXXXX
4	Limite 1	No	XXXXXXXXXXXXXXXXXX
5	Limite 2	No	XXXXXXXXXXXXXXXXXX
6	Stop	Si	Señal de stop normalmente esta cerrado
7	Home 0	Si	Interruptor NC del limite X
8	Home 1	Si	Interruptor NC del limite Y
9	Home 2	Si	Interruptor NC del limite Z
10	Fast INT	No	XXXXXXXXXXXXXXXXXX
11	GND	Si	Señal de tierra o 0V
12	Chasis		Gnd del cable blindado

Fuente: Jesús Rodríguez

El la figura 25, ver Anexo 3, se muestra el diagrama donde se detalla la forma de conectar los dispositivos en el socalo, los pines 7, 8, 9 cierra el circuito con tierra por medio de los finales de carrera ubicados en la máquina CNC, mientras que el pin 6 correspondiente al STOP muestra si ocurrio un paro de emergencia al presionar el boton del mismo nombre de la máquina CNC, cortando la energia del contactor denominado ESR, logrando conmutar los contactos desconectando los driver de los motores a pasos de los ejes X, Y, Z y el driver del husillo.

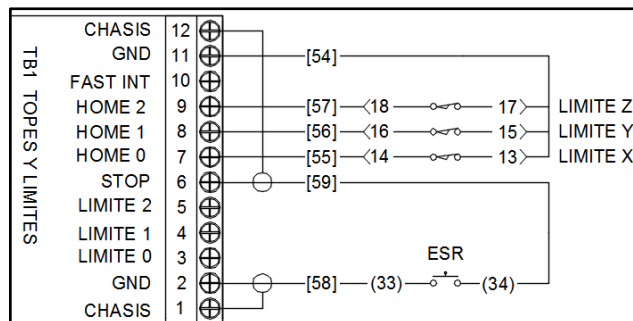


Figura 25: Conector de finales de carrera.

Fuente: Jesús Rodríguez

Socalo 2 (TB2): Éste es un conector para salidas digitales en la máquina no se utiliza ninguno de los pines ya que no contienen componentes extras para el control de aire presurizado, sistema de enfriamiento, carruseles de cambio de herramienta y ningún otro componente, ver Anexo 3.

Socalo 3 (TB3): De este socalo solo se utiliza el pin 2 el cual cierra el circuito para energizar la bobina del SGR, el cual habilita el driver del husillo siempre y cuando el seguro de la puerta este cerrado por medio del contacto 2, en la figura 26, ver Anexo 3, se muestra el diagrama de conexión.

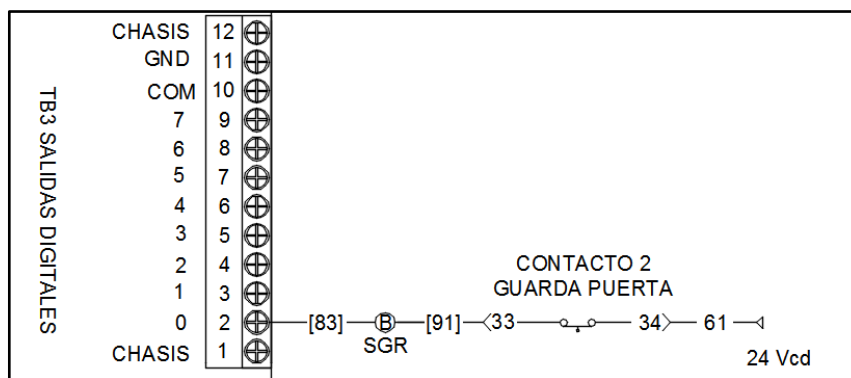


Figura 26: Conector de salidas digitales.

Fuente: Jesús Rodríguez

Socalo 4 (TB4): Es el conector para entradas análogas, para recibir la señal variable de voltaje proveniente del divisor de voltaje, generado por los potenciómetros del FEEDRATE y SPINDLE; por la nomenclatura de los pines 2 y 4 este recibe señales de tipo positivo en un rango de 0 a 5 Voltios de corriente directa, tal como se indica en la figura 27, ver Anexo 3.

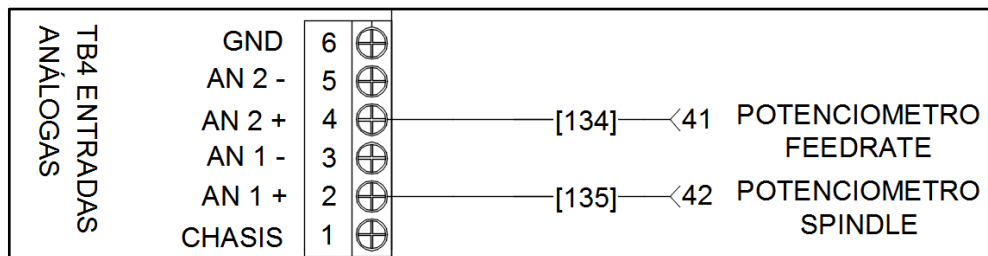


Figura 27: Conector de entradas análogas

Fuente: Jesús Rodríguez

Socalo 5,6,12 (TB5, TB6, TB12): Corresponden a los conectores para los motores paso a paso, los colores indican que deben instalarse en correspondencia y no variar por ningún motivo los colores el azul y el tomate corresponden a una bobina del motor, y los colores amarillo y rojo son la otra bobina, estos motores son bipolares y tienen cuatro cables, como lo indica la siguiente figura 28, ver Anexo 3.

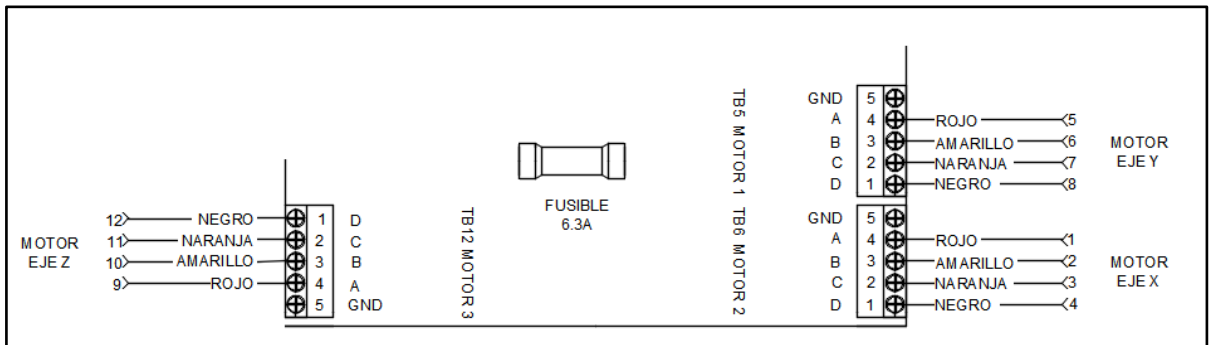


Figura 28: Conector de motores
Fuente: Jesús Rodríguez

Socalo 7 (TB7): Es una salida análoga que sirve para variar la revoluciones del motor SPINDLE, recibe una señal variable de voltaje de 0 a 10 voltios, por tanto el controlador envía un voltaje correspondiente con esos parametros mediante los pines 1 y 2, en la siguiente figura 29, ver Anexo 3, muestra la forma de conexión.

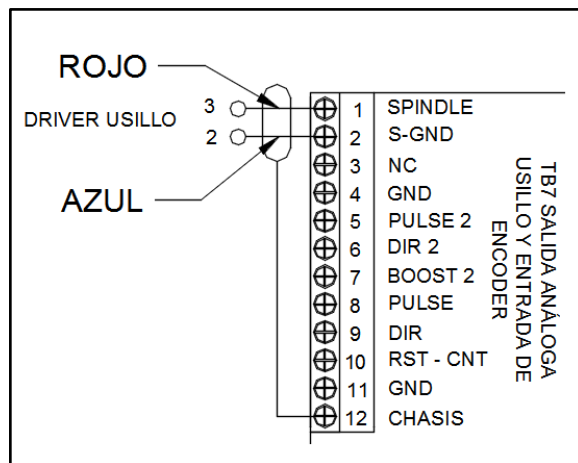


Figura 29: Conector salidas análogas
Fuente: Jesús Rodríguez

Socalo 8 (TB8): Este conector no es utilizado en la máquina CNC por lo tanto no se describe, ver Anexo 3.

Socalo 9 (TB9): Activan al cerrar el circuito con tierra mediante los pines 9, 10, 11. De los cuales 9 y 11 corresponden a un contacto normalmente cerrado denominado guarda contacto 1 del seguro de la puerta. En el pin 10 recibe una señal del rele GIR el cual indica al controlador que el driver del husillo esta encendido, en la figura 30, ver Anexo 3, muestra la forma de conexión.

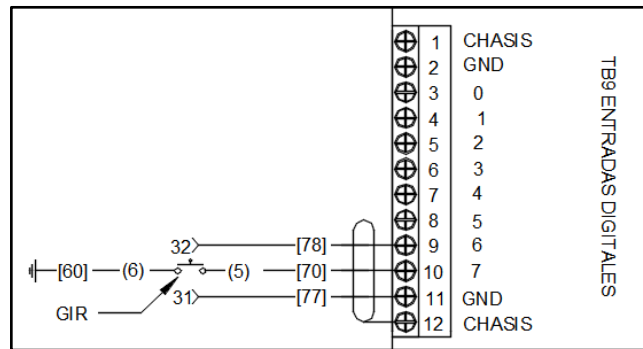


Figura 30: Conector entradas digitales
Fuente: Jesús Rodríguez

Socalo 10 (TB10): Éste conector es la fuente de alimentación, los pines describen los voltajes de salida que la tarjeta puede entregar, en este caso utiliza la fuente de 5 voltios de corriente directa, en la figura 31, ver Anexo 3, muestra la forma de conexión.

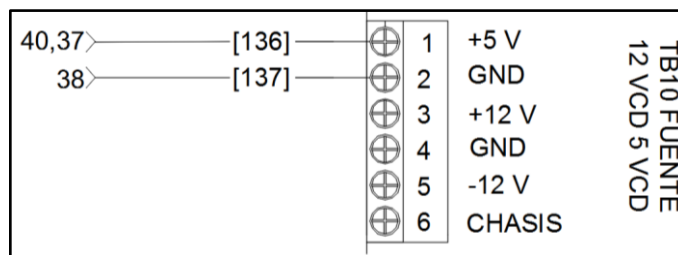


Figura 31: Fuente de alimentación
Fuente: Jesús Rodríguez

Socalo 11 (TB11): La caja eléctrica del controlador funciona a 220 voltios CA por lo cual tiene un transformador que entrega tres voltajes alternos: 28VCA para los drivers, 28VCA para los circuitos lógicos y un voltaje de 110 VCA para energiza el ventilador y

otros componentes. Los dos voltajes de 28 VCA se conectan en este zócalo, como se muestra en la siguiente figura 32, ver Anexo 3.

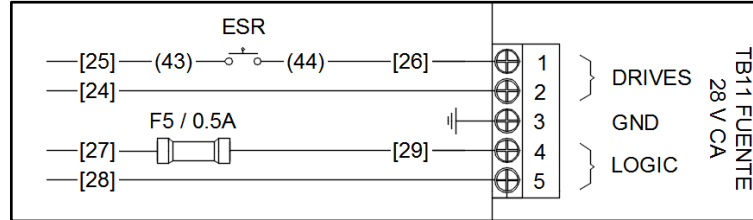


Figura 32: Fuente da alimentación de tarjeta
Fuente: Jesús Rodríguez

4.6. ESPECIFICACIONES DE LOS DISPOSITIVOS NECESARIOS PARA HABILITAR LA MÁQUINA CNC.

En la siguiente tabla 14, se muestra los elementos necesarios para rehabilitar la máquina CNC.

Tabla 13: Descripción componentes técnicos

Elemento	Diagrama	Especificaciones
<i>Driver de motores paso a paso</i>		Corriente mínima de 2A de corriente de trabajo, voltaje de alimentación de 30 VCD, sistema de chopeo o troceador de señal.
<i>Contactos</i>		manejar señales de lógica digital, con una configuración pull-up, la señal en la salida tomara un estado de nivel bajo de acuerdo a la lógica TTL.

Potenciómetros		<p>Señal variable de voltaje directo de 5V, en el anexo 11, se muestra la relación de voltaje de entrada y salida.</p>
$R1 = 4.7k\Omega \quad R2 = 4.7k\Omega \quad V_{out} = V_{in} \left(1 + \frac{R2}{R1} \right) \quad out$ $= V_{in} \left(1 + \frac{4.7k\Omega}{4.7k\Omega} \right)$ $out = V_{in} * (2)$		
Motor del husillo		<p>El relay funciona cuando el transistor recibe un nivel lógico alto, activando el contactor SGR.</p>

Fuente: Jesús Rodríguez

4.7. ELABORACIÓN DE LOS PLANOS ELÉCTRICOS DEL SISTEMA A AUTOMATIZARSE

El diseño de los planos eléctricos va relacionado con la distribución de los elementos eléctricos y electrónicos dentro de un tablero de control, los mismos que deben cumplir las condiciones técnicas para almacenamiento de estos dispositivos.

Los dispositivos, circuitos y demás elementos deben estar distribuidos de tal manera que eviten los ruidos eléctricos que se presentan entre los dispositivos eléctricos y electrónicos y de esta manera asegurando el correcto funcionamiento de la máquina CNC.

Los cables de alimentación así como los de comunicación, y demás conectores que dispone el sistema se encuentra separado y ubicado por canaletas de dimensión 30 x 80, ver Anexo 1.

La máquina dispone de relay o contactores los mismos tienen por función de protección de los circuitos de control y los elementos que intervienen directa e indirectamente dentro del equipo de maquinado, ver Anexo 2, 5, 7,10 donde se observa el diagrama de conexión y disposición de los contactos, y el de los contactores y en el anexo 4, se encuentre detallada el diagrama eléctrico.

4.8. CIRCUITOS Y DIAGRAMAS DE CONTROL PARA LA MÁQUINA CNC.

La máquina CNC debe responder a una serie de seguridades eléctricas para precautelar los componentes eléctricos, electrónicos y mecánicos, la tarjeta controladora recibe por parte de la maquina CNC señales como finales de carrera, voltajes variables de los potenciómetros, estado de la puerta y envía como medio de control señales eléctricas para habilitar los motores paso a paso, encendido o pagado del husillo siempre y cuando el sistema cumpla con ciertas condiciones para funcionar, éstas condiciones se representan en la siguiente figura 33, que es el diagrama de flujo de control de dispositivos en la máquina CNC, para ver a detalle la estructura de control y funcionamiento vea el Anexo 12.

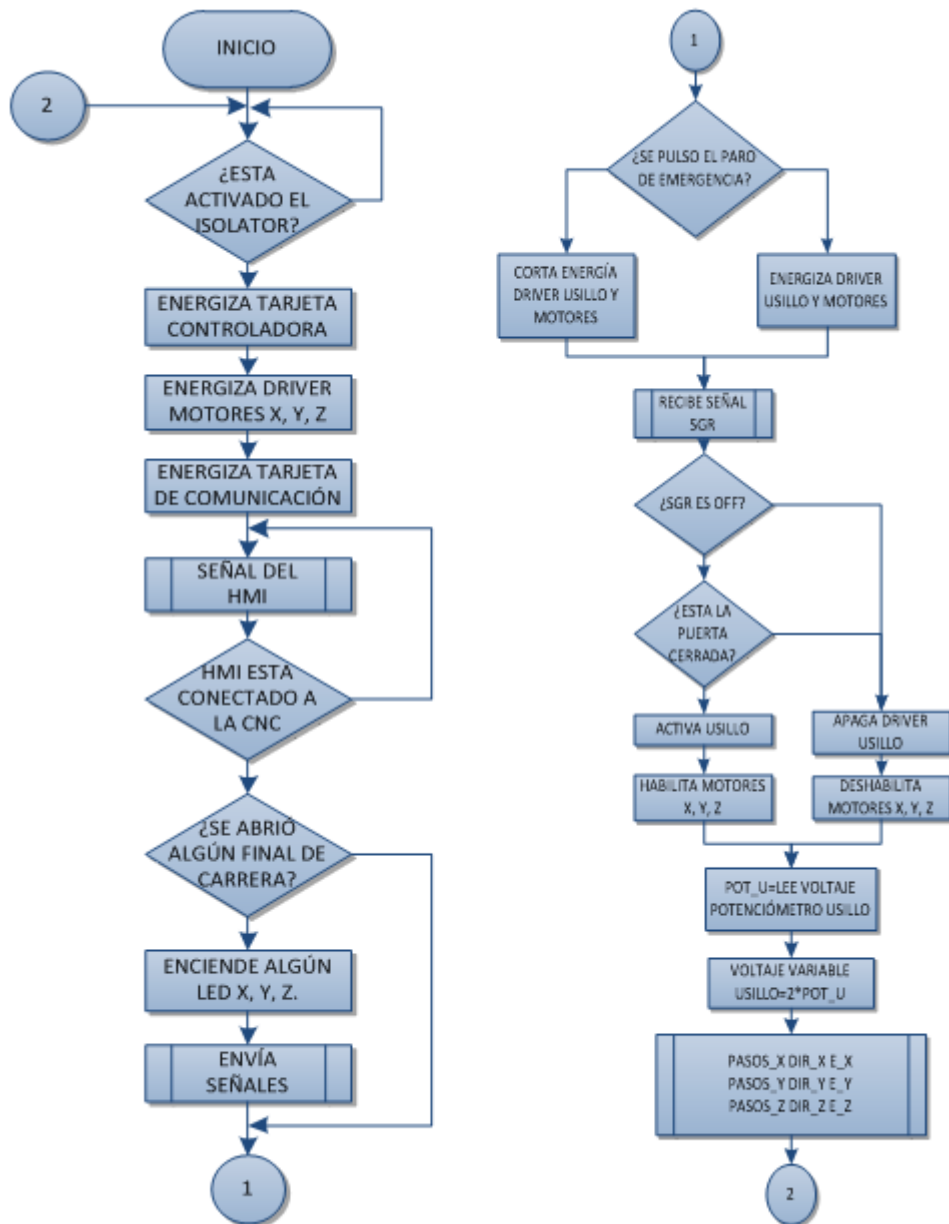


Figura 33: Diagrama de flujo de control máquina CNC
Fuente: Jesús Rodríguez

4.8.1. Driver de motores pasos paso

El driver del motor paso a paso debe manejar una corriente de mínimo 2A y para aumentar la eficiencia en torque y velocidad se debe utilizar un sistema de chopper. Normalmente para activar un motor paso a paso se le debe aplicar un voltaje nominal por bobina relativamente bajo lo cual por secuencia de activación genera un torque bajo y no se puede utilizar a altas tasas de pulsos puesto que el motor no respondera a altas velocidades.

Los motores son de tipo bipolar es decir cuentan con solo dos bobinas lo cual para generar secuencia de pasos es necesario invertir en cada paso la polaridad de sus bobinas por lo cual se utiliza el circuito integrado l298 que es un puente H de potencia, en la siguiente figura 34, muestra el diagrama esquemático del driver de motor paso a paso, y en el Anexo14, se detalla el plano esquemático del diseño electrónico.

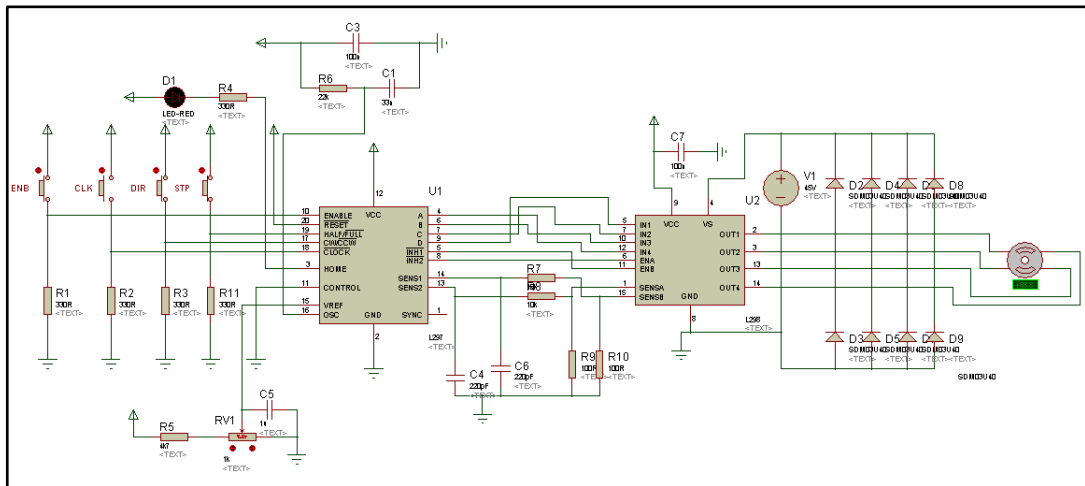


Figura 34: Diagrama esquemático de motores
Fuente: Jesús Rodríguez

Como sistema de protección para el puente H de potencia se utiliza 8 diodos rápidos configurados en doble puente de diodos, como se ilustra en la siguiente figura 35.

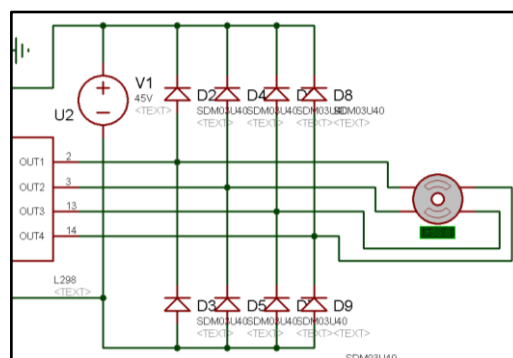


Figura 35: Diagrama esquemático de protección de motores
Fuente: Jesús Rodríguez

Por medio de la configuración de resistencias tómanos el valor de corriente expresado por la caída de voltaje en la resistencias R9 y R10 que pasan por las bobinas del motor y las enviamos al integrado L297 para que realice el chopeo y desactive los enable del

comparador 1 y 2 del otro integrado L298, como se muestra en la siguiente figura 36, y en el Anexo 15 se muestra los datasheet de los integrados utilizados.

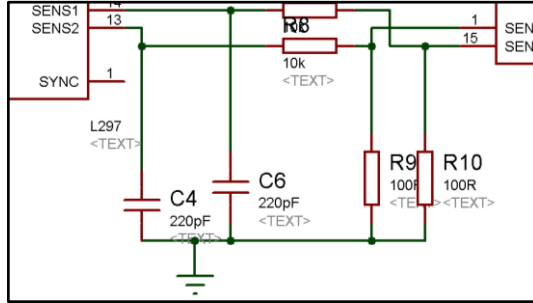


Figura 36: Diagrama esquemático de control de corriente
Fuente: Jesús Rodríguez

Por medio del potenciómetro generamos una tensión variable para el comparador de voltaje de chopeo, con ello podemos tener un control de Corriente, en la figura 37, se muestra el esquema.

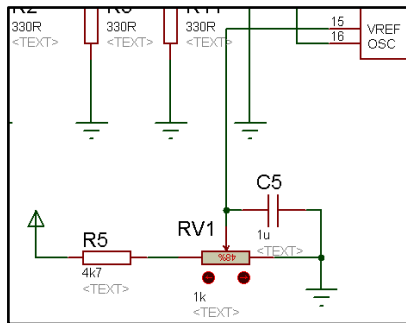


Figura 37: Diagrama esquemático fijar la corriente
Fuente: Jesús Rodríguez

Para regular la corriente a la que debe trabajar el driver hay que calcular la tensión de referencia según la siguiente fórmula:

$$V_{ref} = 0.5 * corriente\ nominal\ del\ motor \quad (3)$$

Con éste valor de voltaje calculado y la ayuda de un multímetro medimos la tensión de referencia y ajustamos el potenciómetro hasta obtener el valor deseado, en el pin 15 del circuito integrado L297.

Diseño final de la tarjeta de drivers de los motores cuenta con los siguientes dispositivos que se detallan en la tabla 15.

Tabla 14: Componentes del driver de motores

Materiales para un driver de motor a pasos	
Cantidad	Material
1	Conector IDC de 10 pines
1	Circuito integrado L297
1	Circuito integrado L298
8	Diodos rapidos de 3A
2	Capacitores ceramicos de 100 nF
1	Capacitor 33 nF
1	Resistencia de 22K
2	Resistencias de 0.5 ohm de 5W
2	Resistencias de 10K
1	Trimmer de 1k ohm
1	Capacitor ceramico de 1uF
2	Capacitor ceramicos de 220pF
3	Borneras dobles

Fuente: Jesús Rodríguez

A continuación se muestra la figura 38, del diagrama pictorico del controlador y en el anexo 13, se detalla el pictorico y el tridimensional.

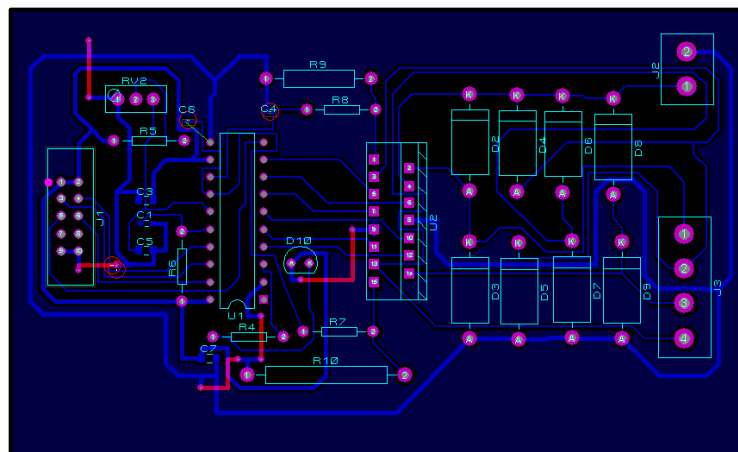


Figura 38: Diseño PCB de driver de motores

Fuente: Jesús Rodríguez

4.8.2. Tarjeta controladora

La tarjeta controladora dispone las mismas características de la tarjeta original, las misma que emite señales a la tarjeta de comunicación.

La tarjeta controladora dispone de cuatro circuitos los cuales se encargan de diferentes funciones cada una, éstas se encuentra en configuración pull – up controladas por un transistor. Los tres circuitos funcionan de la misma forma, la fuente de alimentación para que funcione proviene de la propia tarjeta que estamos diseñando ya que también contiene una fuente de alimentación de 5 voltios y 12 voltios, como se muestra en la siguiente figura 39.

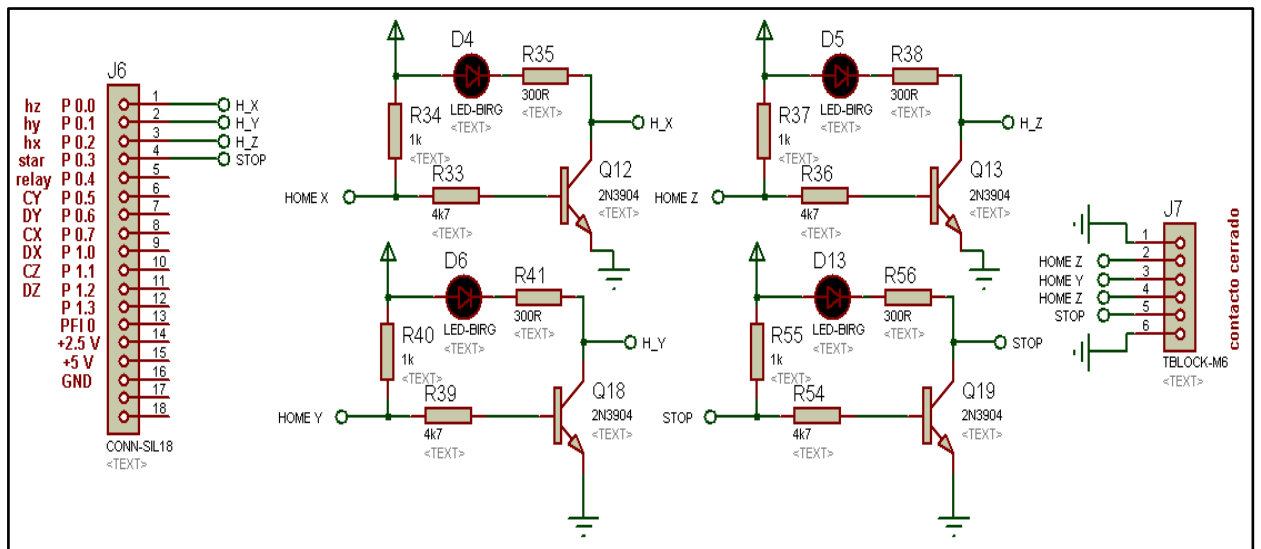


Figura 39: Diagrama esquemático de entradas del controlador
Fuente: Jesús Rodríguez

La tarjeta controladora adquiere señales análogas de velocidad de avance y RPM del husillo que vienen de la máquina CNC por medio de dos potenciómetros. Estas dos señales análogas van directamente a la terminal de la tarjeta de comunicación, solo el voltaje de entrada del SPINDLE_IN pasa por un amplificador operacional multiplicando su voltaje por dos, éste voltaje variable SPINDLE_OUT saldrá de la tarjeta controladora hacia el driver del husillo, en la figura 40, se muestra el diagrama para la amplificación de voltaje.

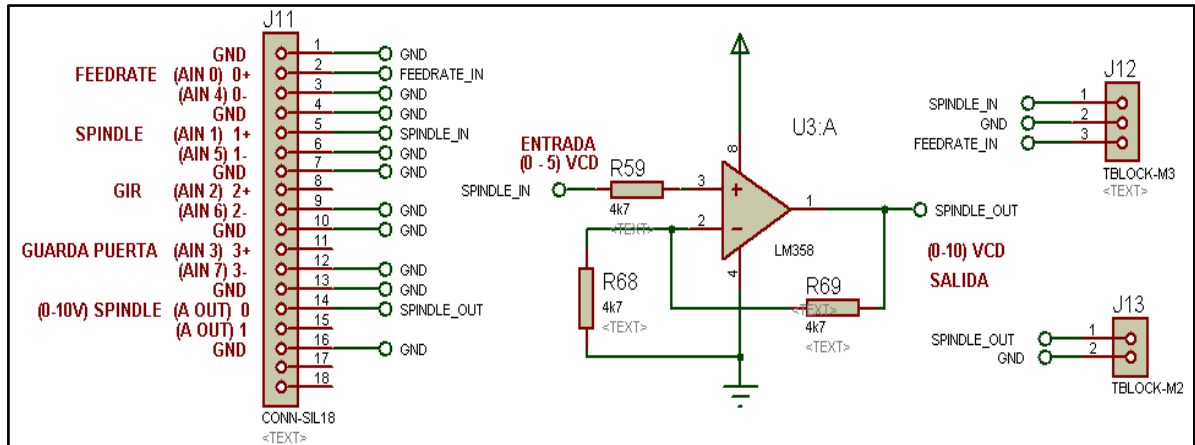


Figura 40: Diagrama esquemático de entradas analógicas
Fuente: Jesús Rodríguez

La tarjeta dispone de un microcontrolador pic 16f628a el cual detecta cualquier actividad de secuencia de reloj de cualquiera de los ejes pondra en un nivel de estado lógico alto al ENABLE activando los tres motores. Luego de activado el ENABLE si ya no existiera actividad en ninguno de los ejes intermamente un contador de tiempo esperara un periodo de cinco segundos y el ENABLE lo pondra en un estado bajo, este control se realiza para que no se sobrecamienten los motores al estar constantemente energizados, en la figura 41, se observa parte del codificado del microcontrolador, ver Anexo 21.

```

#include <main.h>
long tiempo=0;
#int_RB
void RB_isr(void)
{
disable_interruptions(INT_RB);
output_high(pin_a1); //activar enable
tiempo=0;
}
void main()
{
while(true){
set_tris_a(0x00);
set_tris_b(0xff);
enable_interruptions(INT_RB);
enable_interruptions(GLOBAL);
if(tiempo>=5){
output_low(pin_a1);
}
}
}

```

Figura 41: Programación de Control de motores
Fuente: Jesús Rodríguez

La bornera J21 recibe el voltaje alterno de una de las bobinas del transformador cuyo voltaje es de 28VCA, pasa por el puente rectificador de 4A, y para eliminar el rizado se usa un capacitor C17, cuando por la resistencia R76 de 1 ohmio pasa una determinada corriente se creara una caida de voltaje de 0,7VCD logrado activar la base del transistor haciendo que pase toda la carga de corriente a través del transistor Q25, y para regular el voltaje a 12VCD se utiliza el regulador 7812, para filtrar el ruido se aplica una serie de filtro en paralelo que eliminara la mayor parte de ruido eléctrico propio de sistemas industriales.

El voltaje de 12VCD se conectara a un bornera de salida para alimentar dispositivos externos a la tarjeta controladora, en la figura 42, muestra el diagrama de la fuente de alimentación.

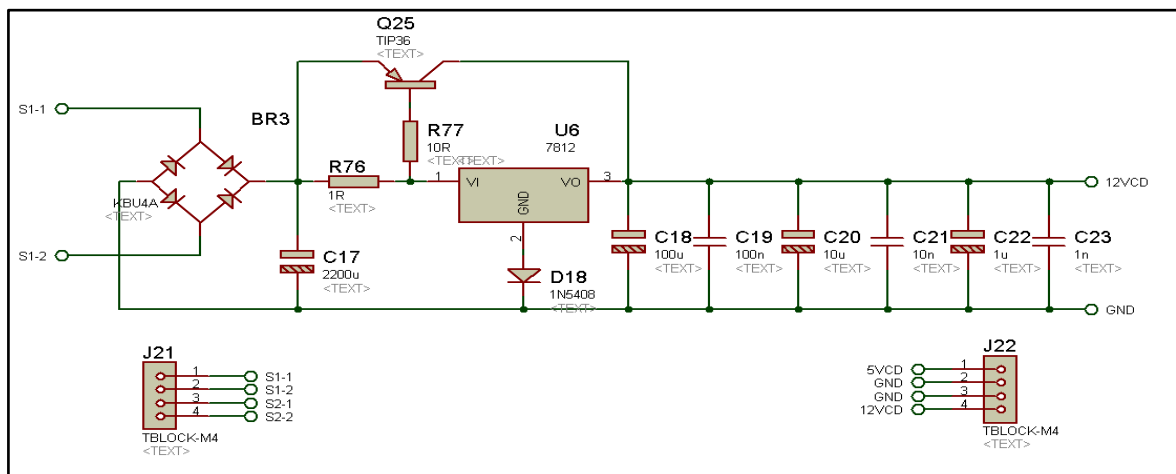


Figura 42: Diagrama esquemático de fuente de alimentación
Fuente: Jesús Rodríguez

La siguiente tabla 16, se muestra la lista de todos los componentes necesarios y utilizados para la elaboración, en los Anexo 16, 17, se adjunta a detalle los componentes, ubicación de la placa.

Tabla 15: Componentes de tarjeta controladora

Materiales para la tarjeta controladora			
Cantidad	Material	Cantidad	Material
6	Borneras dobles	1	Relay de 12V
6	Borneras triples	1	Circuito integrado LM358
7	Led	2	Capacitores 2200uF/60V
2	Puentes rectificadores	2	Capacitor 100uF/100V
5	Transistores 2n3904	2	Capacitores 10uf/50V
11	Resistencias de 4.7Kohm	2	Capacitores 1uf/25V
3	Conectores ICD 10 pines	4	Resistencias de 1K
7	Resistencias de 220 ohm	2	Resistencias de 1ohm/5W
2	Espadines macho de 16 pines	2	Resistencias de 10ohm/5W
1	Diodo 1n4007	2	Capacitores ceramicos 100nF
2	Capacitores 10nF	2	Capacitores 1nF
1	Placa doble capa 20x10 cm		

Fuente: Jesús Rodríguez

4.8.3. Tarjeta de comunicaión

La tarjeta de comunicaión se encuentra realizado mediante una tarjeta Arduino Uno, la cual recibe las señales eléctricas de la tarjeta controladora diseñada y reenvia por otras terminales señales de control para motores, puerta y husillo, este es el componente más importante, el arduino gestiona la comunicaión entre la interfaz gráfica de la PC y la máquina CNC por medio de comunicaión USB, por su diseño, arquitectura, capacidad, y tecnología el arduino controla la mayor parte de dispositivos y componentes de la máquina CNC los pines digitales 0,1 no se utilizan pues se destina para comunicaión, los pines 2,3,4, son de salida y corresponden a las señales de pulsos para rotar los motores X, Y, Z respectivamente, en al figura 43, podemos observar el diagrama esquemático de la tarjeta de comunicaión.

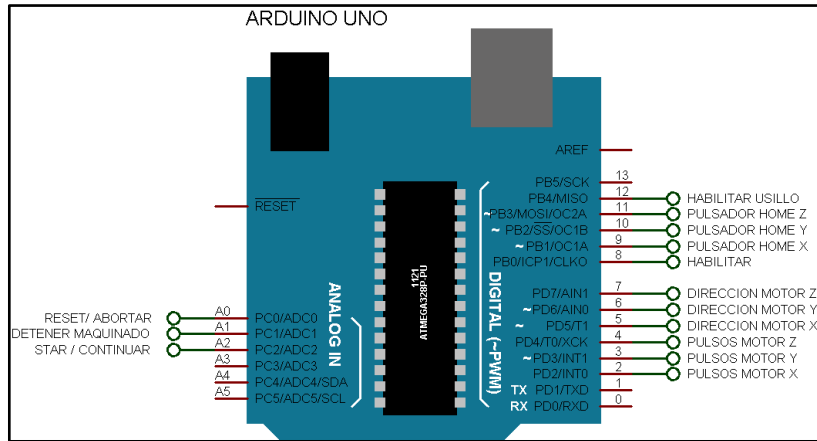


Figura 43: Diagrama esquemático del Arduino uno
Fuente: Jesús Rodríguez

En los pines analógicos del arduino se encuentra configurados como entradas digitales:

- A0: Paro de emergencia
- A1: Pause
- A2: STAR

Para poder utilizar botones físicos como virtuales y controlarlos desde la interfaz gráfica se utiliza otro PIC18F2550 con comunicación USB, éste microcontrolador trabaja a una frecuencia de 20MHz con cristal externo lo cual por programación permite trabajar con una velocidad de transferencia de datos de hasta 48Mhz. En este microcontrolador se receptaran señales de los finales de carrera, en la figura 44, se observa el diagrama para realizar estas tareas, Ver Anexo 16.

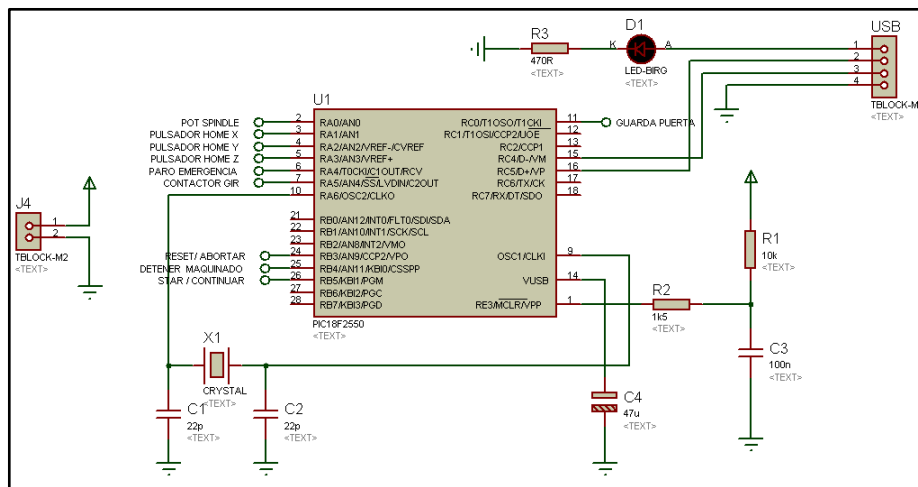


Figura 44: Diagrama esquemático de control de Arduino uno
Fuente: Jesús Rodríguez

4.9. DISEÑO DEL SOFTWARE DE CONTROL PARA LA INTERACCIÓN DEL USUARIO Y LA MÁQUINA CNC

4.9.1. Interfaz a nivel hardware.

La comunicación entre el operador de la máquina CNC y el software, se realiza a nivel de hardware a través de dos potenciómetros tanto de selección de velocidad de avance, selección de velocidad en RPM del husillo y el botón de paro de emergencia ubicada en la máquina CNC.

4.9.2. Interfaz a nivel Software

La comunicación entre el operador y la máquina a nivel software también se plantea con base en la asignación de botones virtuales cada uno con su respectiva función, ésta interfaz cuenta con 3 ventanas donde se agrupan todas las funciones básicas y avanzadas que requiere el sistema de control, cuya programación de código y animación está desarrollado en el software LabVIEW.

4.9.3. Diseño de la interfaz HMI

La interfaz de control de máquina CNC es diseñada utilizando software LabVIEW, la misma que envía información mediante comunicación USB a la tarjeta Arduino. La comunicación USB la recibe en forma de caracteres ASCII.

La interfaz permite controlar a la máquina de manera eficiente entre esas funciones esta:

- Cargar códigos G del Mastercam X, o cualquier tipo de software para control numérico.
- Posicionamiento de las máquinas, y posicionamiento a home.
- Detección, paro de emergencia, puesta en marcha de la máquina CNC.
- Recibe información en tiempo real del maquinado.

La siguiente figura 45, muestra la apariencia del software de control de la máquina CNC, donde permite cargar un código G y enviar a maquinar, muestra display

numéricos en los cuales se encuentra las coordenadas tanto de posición real de la máquina como de la posición de trabajo según el código. Con un indicador gráfico se visualiza en dos dimensiones los trazados con las coordenadas que envía a la tarjeta de comunicación cuando se encuentra en un ciclo de maquinado.



Figura 45: Software HMI para el control de la máquina CNC
Fuente: Jesús Rodríguez

La interfaz de control está diseñada en base a funciones necesarias para que la máquina CNC se comunique correctamente con el HMI y realice su proceso de maquinado correctamente.

Para diseñar el HMI hay que tener en cuenta dos aspectos importantes, el primero como va a realizar el proceso de maquinado el operario y que necesita para realizarlo, de este análisis se tiene dos parámetros como el usuario va a interactuar con la máquina CNC. El proceso inicia abriendo las comunicaciones, posteriormente se desactiva la alarma que se produce para que el operario realice home a la máquina y se ubiquen los ejes en su posición idónea para el maquinado, luego de realizado este proceso el sistema guarda automáticamente las coordenadas, permitiendo encontrar el cero pieza, moviendo cada eje a voluntad del operador de similar forma esta coordenada es guardada, y se queda esperando las líneas de instrucción, que se puede enviar línea por línea o el código completo, en las siguientes figuras 46, 47, 48, muestra el diagrama de flujo de cómo se encuentra realizado el HMI del controlador.

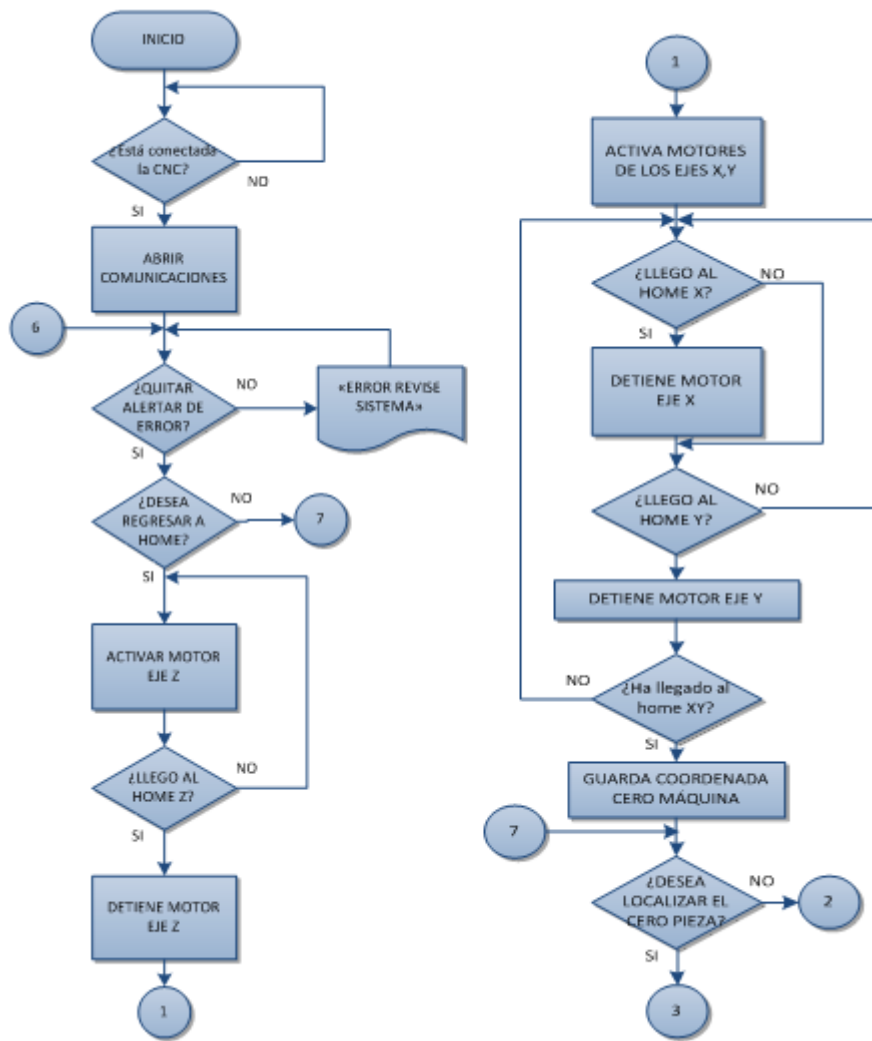


Figura 46: Diagrama de Flujo del HMI
Fuente: Jesús Rodríguez

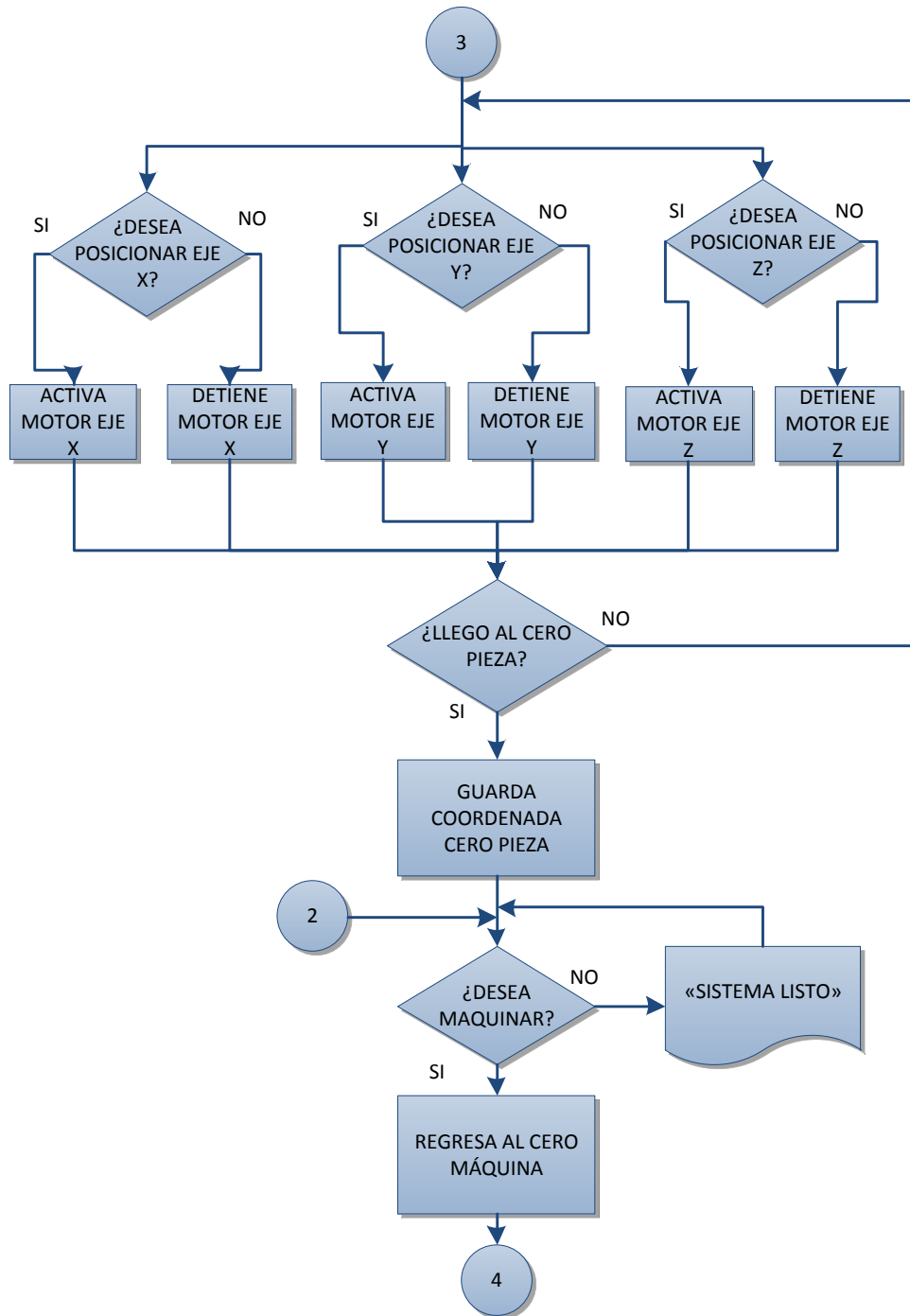


Figura 47: Diagrama de Flujo del HMI
Fuente: Jesús Rodríguez

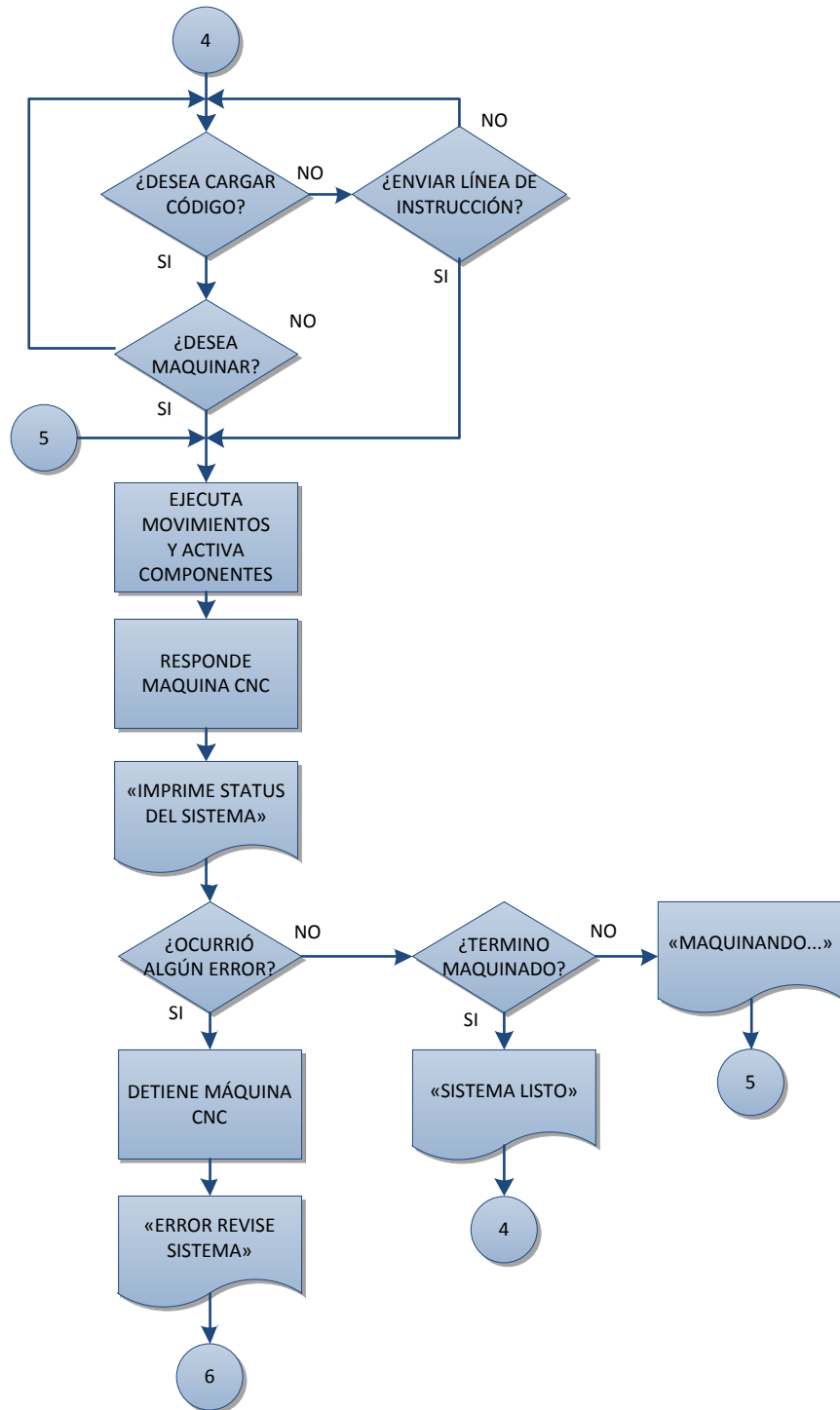


Figura 48: Diagrama de Flujo del HMI
Fuente: Jesús Rodríguez

4.9.4. Partes del sistema HMI diseñado

Al ejecutar el HMI, éste tiene tres pestañas como lo indica la figura 49, la primera “Ajustar máquina” en esta pestaña se tiene botones virtuales que permiten hacer movimientos controlados para posicionamiento de una nueva coordenada para un nuevo cero pieza, esta tarea se la realiza una sola ocasión.

También permite gestionar alarmas, reiniciar el Arduino en caso de inconveniente, indica también los estados en los que se encuentran los finales de carrera, si la puerta está abierta o cerrada, si el husillo está activo o no.

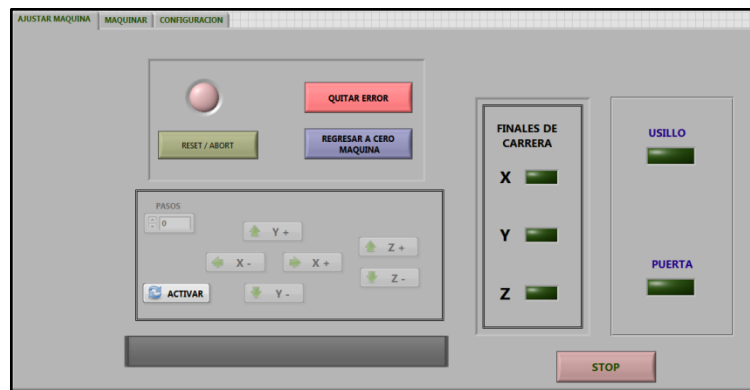


Figura 49: Pestaña de configuración
Fuente: Jesús Rodríguez

La siguiente tabla 17, describe la actividad que realiza cada uno de los botones virtuales e indicadores presentes en esta pestaña de configuraciones iniciales y de puesta a punto de la máquina antes de proceder a realizar una proceso de maquinado con instrucciones de código G.

Tabla 16: Descripción de elementos del HMI de la pestaña ajustes

Elemento	Componente	Descripción
Quitar error		Indica que la máquina a sido activada y debe quitarse el error.
Reset /Abort		Reinicia el Arduino y el proceso estará apto para entrar en funcionamiento.
Regresar a cero máquina.		Si ocurre un corte de energía, la máquina al reiniciar debe hacer un ciclo de home para que tenga un punto de referencia conocido, del cual parta para maquinar.
Botones de desplazamiento		Se utiliza para mover cada uno de los ejes por separado y a una tasa variable. Se utiliza este proceso para ubicar el cero pieza.
Finales de carrera		Son indicadores gráficos y se encienden cuando los respectivos ejes de movimiento están en el cero máquina.
Indicador de Husillo	USILLO 	Indica si el husillo esta encendido o no.
Indicador de puerta	PUERTA 	Indica si esta la puerta abierta o cerrada.

Fuente: Jesús Rodríguez

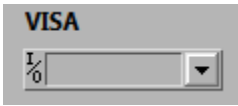



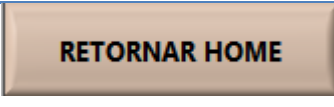

La siguiente pestaña “maquinar”, como se muestra en la figura 50, En ella se encuentran todas las operaciones de control supervisión, alertas, configuraciones de posicionamiento, indicadores de estado, controles que permiten gestionar instrucciones


sobre la máquina, además de un set de botones para cargar el código con el que se desea trabajar, en la siguiente tabla 18, muestra los controles e indicadores virtuales utilizados y su descripción.



Figura 50: Pestaña "MAQUINAR"
Fuente: Jesús Rodríguez

Tabla 17: Descripción de elementos del HMI de la pestaña “MAQUINAR”


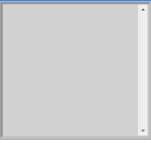

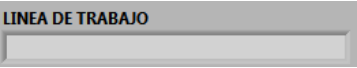
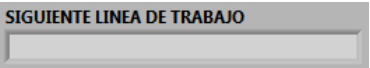

Elemento	Componente	Descripción
Puerto de comunicación RS232		Puerto donde se asigna el COM odirección del arduino.
Puerto de comunicación USB		Puerto donde se asigna el COM o dirección del PIC 18F2550 supervisor
Nuevo cero pieza		Guardar en la memoria eeprom del Arduino esta coordenada como un código G92.
Retronar cero		Luego de un ciclo de trabajo realiza un retorno al cero pieza seleccionado con el que se está trabajando
Retronar home		Regresa al home o cero máquina.
Stop		Detiene todo proceso de la máquina CNC y cierra el software y comunicaciones.

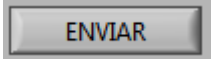
Indicador RUN	RUN 	Éste se activa cuando la maquina está realizando un ciclo de maquinado.
---------------	---	---

Fuente: Jesús Rodríguez

La máquina para trabajar recibe una secuencia de códigos G, que son instrucciones de movimiento y control por lo cual la siguiente tabla 19, muestra los botones y controles para importar archivos con extensión TXT, NC, y proceder a maquinar.

Tabla 18: Descripción de elementos del HMI de la pestaña ajustes




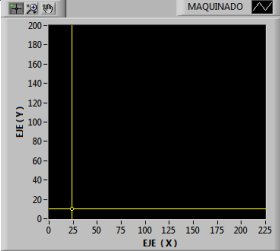
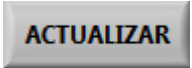
Elemento	Componente	Descripción
Cargar código		Este botone permite accede al explorador de documentos de windows donde se buscará el documento txt que contega el codigo G para maquinar.
Lista de códigos cargados		En este cuadro de texto se carga los comandos de código G cargados.
Iniciar		Este es el botón para iniciar el maquinado siempre y cuando ya esté preparado el material a trabajar y ubicado el cero pieza.
Línea de trabajo		Muestra que instrucción se está ejecutándose en el maquinado.
Próxima línea de trabajo		Muestra la siguiente operación a realizarse.
Comandos		Esta línea permite enviar a la máquina CNC instrucciones de código G por requerimiento del

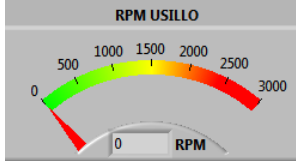
		operario.
Enviar		Este botón trabaja conjuntamente con la barra de “Comandos” y envía la instrucción a la máquina CNC.

Fuente: Jesús Rodríguez

La tabla 20, mostrada a continuación indica las coordenadas en las que se ubica a cada instante de un maquinado.

Tabla 19: Descripción de elementos del HMI de la pestaña ajustes



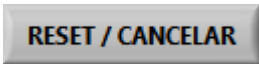





Elemento	Componente	Descripción
Coordenadas		Indica las posiciones tanto de la máquina como de la instrucción al maquinar.
Barra de estado		Muestra el estado en que se encuentra la máquina.
Sensores de límite o home		Visualiza el estado de los sensores de límite o Home de los tres ejes.
Grafica en 2D.		Representación en 2D de los trazos que la máquina CNC realiza en un maquinado.
Actualizar		Actualizar el sistema cuando se requiera para saber la condición del Arduino.

Sensor de RPM husillo		Visualizador de velocidad de giro a la que debe trabajar cuando se encienda.
-----------------------	---	--

Fuente: Jesús Rodríguez

En la siguiente tabla 21, se indican los botones de control y monitoreo principales de toda la máquina CNC junto con su descripción. Para mayor explicación y detalle ver Anexo 18, donde se ilustra el manual de operación junto con ejemplos y configuraciones básicas.

Tabla 20: Descripción de elementos del HMI de la pestaña ajustes

Elemento	Fotografía	Descripción
Borrar		Borra el código almacenado y todo lo relacionado
Estado de coordenadas		Indica el posicionamiento actualmente.
Reset /Cancelar		Este botón es utilizado para borrar el registro de errores, refrescar la memoria del Arduino, cancelar cualquier operación.
pausar		Detiene un ciclo de maquinado pero no borra su estado.
Start /Continuar		Permite continuar el ciclo de maquinado en el que se quedó antes de realiza un PAUSE.
	<p data-bbox="639 1641 740 1666">ALARMA</p> 	Se activa para indicar que se ha presionado el paro de emergencia.
	<p data-bbox="667 1749 713 1774">GIR</p> 	Este indica si el husillo está encendido
	<p data-bbox="619 1854 767 1879">COMPUERTA</p> 	Este indica si la puerta está abierta o cerrada

Fuente: Jesús Rodríguez

4.10. PROGRAMACIÓN EN LENGUAJE G DE LABVIEW.

Para realizar la comunicación entre la máquina CNC y el HMI, el dispositivo Arduino instalado en la caja eléctrica tiene que recibir instrucciones, éstas instrucciones son cadena de caracteres tipo ASCII enviadas por comunicación USB. Éstas instrucciones pueden ser líneas de código G de un maquinado o secuencia de caracteres especiales programados en el Arduino para que reconozcan e intérprete una tarea a realizar.

Para cargar un nuevo código G o instrucciones de maquinado al controlador, se utiliza los bloques de programación de LabVIEW, una de ellas es la “File Dialog” el cual al ser ejecutado busca un archivo de extensión (.txt) con el código G a maquinar, los siguientes subvi de programación abren este archivo virtualmente, extraen toda la información como caracteres “string”, y cierran el document, como lo indica la figura 51. La información se visualiza en el cuadro de texto del HMI como “lista de códigos cargados”.

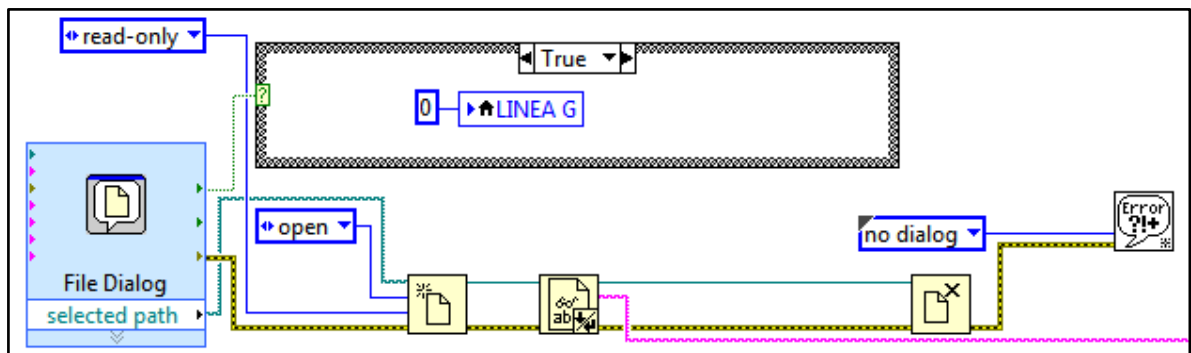


Figura 51: Cargar código de instrucciones CNC
Fuente: Jesús Rodríguez

Una vez cargado la lista de instrucciones de maquinado, el sistema espera a que la máquina muestre en la barra de estado del HMI “Sistema listo..!”, en ese momento la máquina se encuentra activa y espera a que se presione el botón virtual “INICIAR” para que la máquina empiece a trabajar, con esta secuencia el controlador realiza una instrucción ya sea de configuración o movimiento; terminada la ejecución del comando enviado, el HMI envía la siguiente instrucción de trabajo mediante el incremento en uno de “LÍNEA G” , como se muestra en la siguiente figura 52, 53.

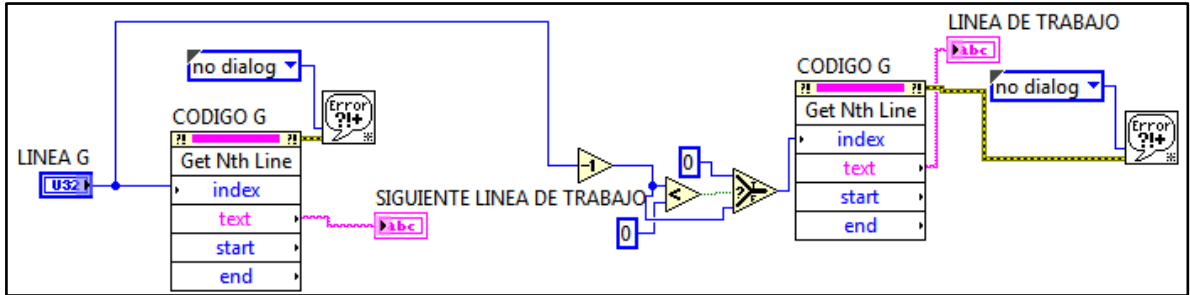


Figura 52: Envío de instrucciones
Fuente: Jesús Rodríguez

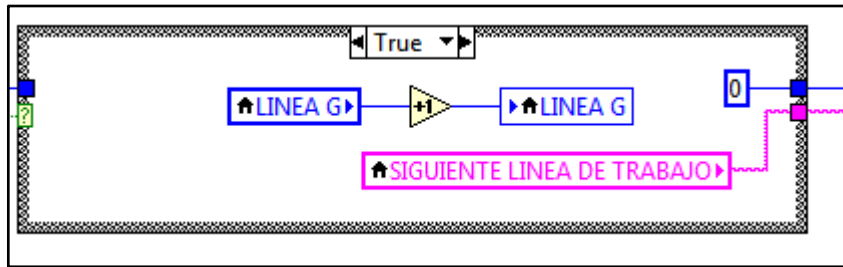


Figura 53: Código para incrementar nueva instrucciones de maquinado
Fuente: Jesús Rodríguez

Para la comunicación entre el sistema HMI y el controlador se lo realiza mediante el visa writer, en la figura 54, se detalla la forma que el software se comunica con el controlador.

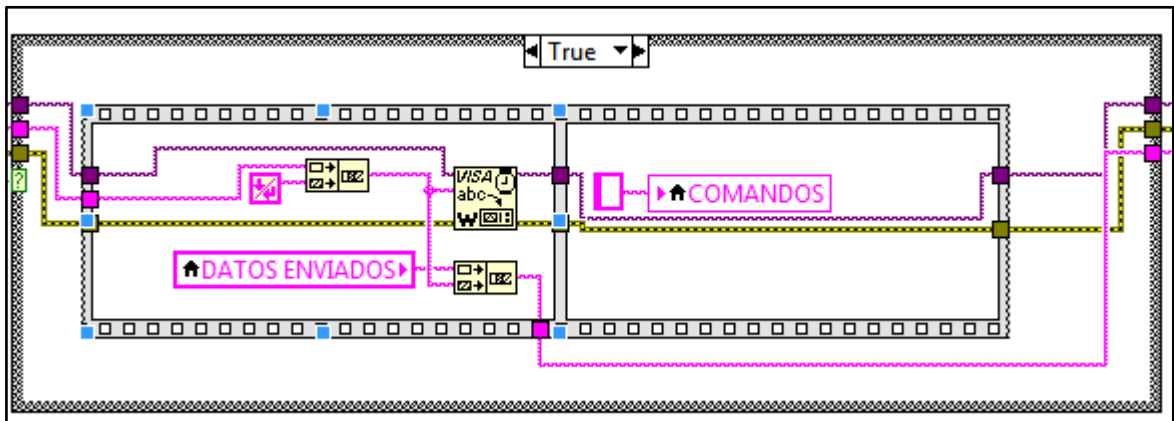


Figura 54: Envío de datos a la tarjeta controlador
Fuente: Jesús Rodríguez

La tarjeta controladora se encuentra diseñado con un micro controlador PIC de la familia 18F2550 que tiene comunicación USB, está programado en lenguaje C. Este micro controlador permite trabajar en forma paralela al Arduino para darle instrucciones en tiempo real desde el HMI incluso cuando el Arduino este en un ciclo de maquinado, estas instrucciones se envían en formato ASCII, todas las señales de los botones RESET, PAUSE, CONTINUAR, son de tipo booleana con lo cual se crea un array booleano y se trasforma a número entero de 8 bits, el puerto serial maneja caracteres por tanto se transforma este número a cadena de caracteres y se escribe en el puerto serial, ésta información recibe, procesa, y ejecuta el micro controlador PIC 18F2550, como medio de respuesta reenvía a la PC información de estado de sensores como finales de carrera, rpm del husillo, estado de la puerta, así como se ilustra en la siguiente figura 55, el codificado de programación tanto del Arduino y del micro controlador PIC 18F2550 están descritos en los Anexos 19 y 20 respectivamente.

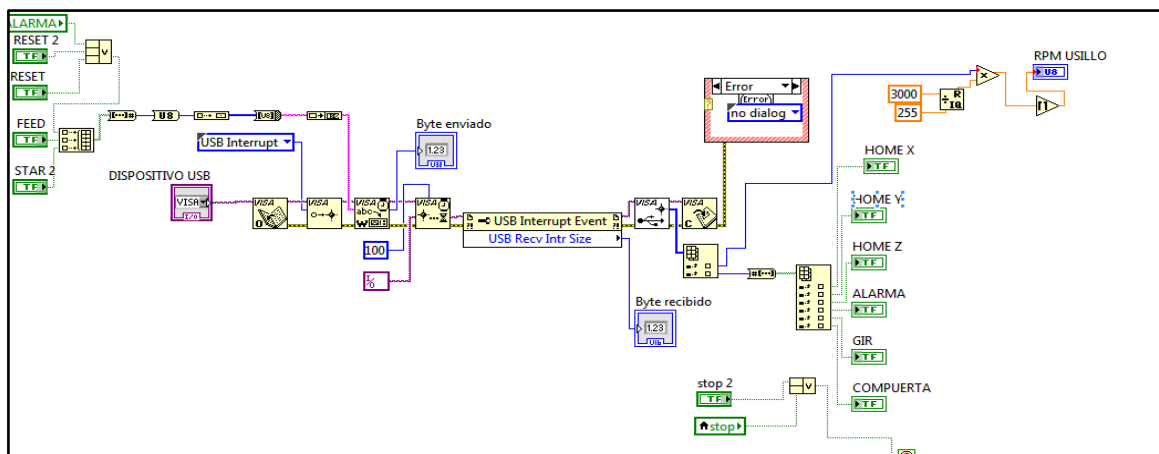


Figura 55: Código de monitoreo de señales
Fuente: Jesús Rodríguez

Las alertas HMI nos indican los límites de los carros de la máquina, y se visualiza mediante barra de texto con letras naranjas, la siguiente figura 56, indica el método utilizado para generar este mensaje en programación LABVIEW.

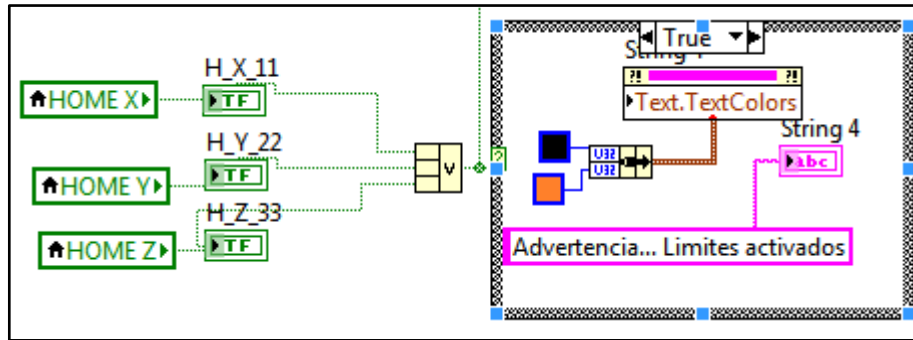


Figura 56: Código de alertas
Fuente: Jesús Rodríguez

La cadena de caracteres extraídos con la información del estado de la máquina CNC, son procesadas y decodificadas por medio del subvi que se indica en la figura 57, obteniendo datos de estado de la máquina y como esta maquinando, si tiene errores, o si espera alguna instrucción, además de las coordenadas de posición de maquinado y de posición real de la máquina CNC, éstas coordenadas sirven también para graficar en 2D en RUTEO.

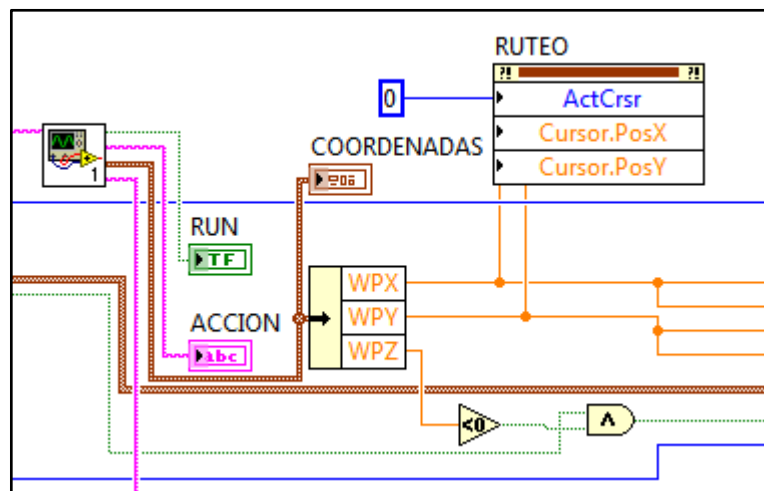


Figura 57: Decodificado de Trama serial
Fuente: Jesús Rodríguez

La forma en que decodifica la información el subvi antes mencionado se indica en la siguiente figura 58, éste proceso consiste en la búsqueda de caracteres, la trama que contiene la información está entre dos signos y su estructura es la siguiente “<estado, posición X, Y, Z, posición máquina X, Y, Z>”, éstos siete datos recibidos son los que se necesita para obtener información real de la máquina.

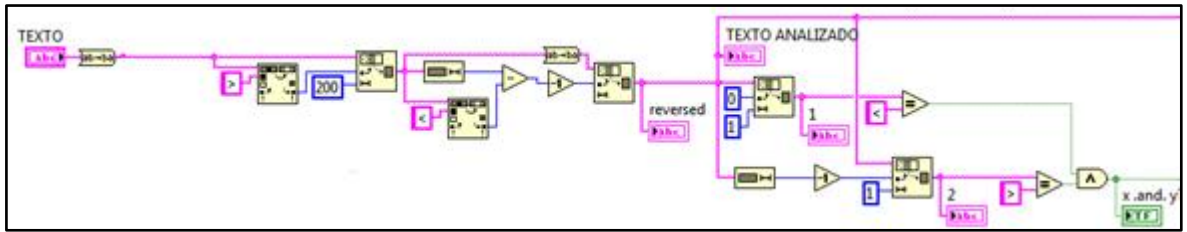


Figura 58: Decodificado de Trama serial
Fuente: Jesús Rodríguez

Encontrada la trama de caracteres es necesario pasar estos datos a formato de número, en la siguiente figura 59, se indica la forma de hacerlo, la búsqueda de caracteres para separar la información lo realiza “Scan from string” de acuerdo al formato de caracteres a buscar, si no llegara a encontrar coincidencias en el formato de ingreso con lo que se busca se crea un registro de memoria virtual hasta que sea reemplazada por la información correcta.

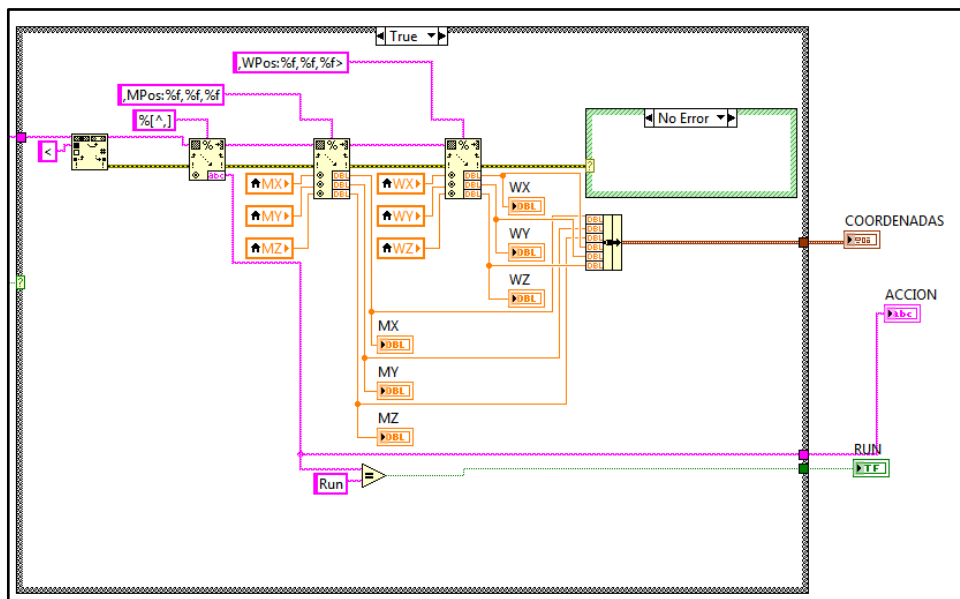


Figura 59: Decodificado de Trama serial
Fuente: Jesús Rodríguez

Con los datos de posición de coordenadas se selecciona de cada diez muestras solo una de ellas y se almacena en un array de coordenadas X, Y, éstas coordenadas representadas en la Grafica Ruteo determinan la figura que se está maquinando. En la siguiente figura 60, se indica el código de programación.

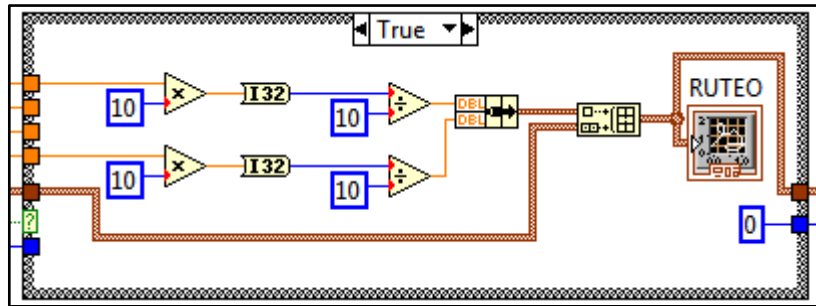


Figura 60: Visualización de maquinado
Fuente: Jesús Rodríguez

De similar forma se procede para enviar eventos de maquinado como ingresar el cero pieza, retornar al cero, quitar alarmas, o realizar un ciclo home por ejemplo en la siguiente figura 61, en este caso se envía como comando activado la secuencia de caracteres de código G correspondiente a “G00X0Y0Z0”, El Arduino interpreta esta secuencia de caracteres. En la tabla 22, se indica los comandos que interpreta el Arduino con su descripción, para mayor comprensión en los Anexos 22, se indica los códigos G admitidos y otros comandos de importancia que admite el Arduino para operar.

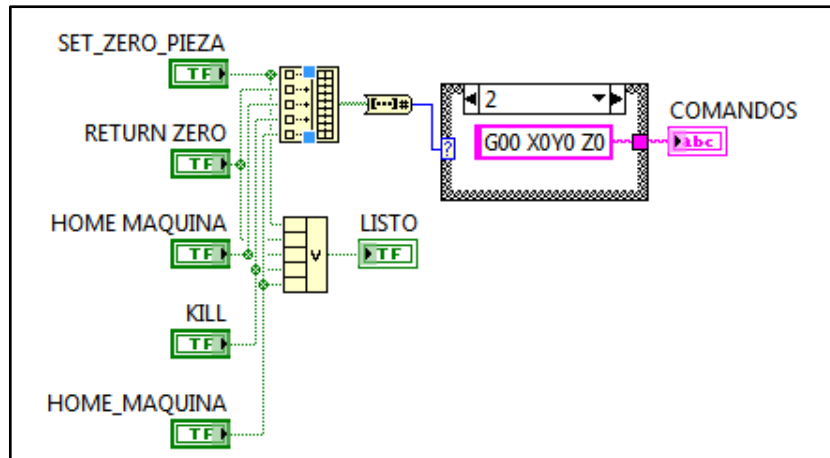


Figura 61: Envío de señales al controlador
Fuente: Jesús Rodríguez

Tabla 21: Comandos G principales

Secuencia de caracteres interpretados en el Arduino		
Nombre	Comando	Acción
Ingresar nuevo cero pieza	G92X0Y0Z0	Permite ingresar la coordenada del objeto a maquinar como cero pieza.
Ir al cero pieza	G00X0Y0Z0	Este comando permite ordenar a la máquina CNC que se dirija al cero pieza
Regresar a home	G28X0Y0Z0	Éste comando hace que la mesa de trabajo regrese a sus respectivos orígenes o llamado también cero máquina.
Crear ciclo Home	\$H	Activa el motor a máxima velocidad del eje Z hasta llegar al final de carrera, luego activa los motores X,Y y realiza el mismo procedimiento encontrando el cero máquina.
Quitar error	\$X	Borra la memoria eeprom de errores del Arduino.
Actualizar	?	Permite conocer el estado de trabajo de Arduino

Fuente: Jesús Rodríguez

4.11. IMPLMETACIÓN DEL CONTROLADOR Y DESARROLLO DEL MANUAL DE USUARIO.

El manual de usuario es un recurso muy útil para el manejo de la máquina CNC, en ella se detalla los procedimientos básicos iniciales al momento de encender la máquina CNC, así como la forma de instalar el software de control, instalación de drivers, y demás operaciones de trabajo admitidas por el equipo, así como de una forma fácil para el manejo de errores que se podrían presentar eventualmente, para mayor información el manual esta detallado en el Anexo 18.

4.12. RESULTADOS DEL FUNCIONAMIENTO

Los resultados obtenidos son eficientes los desplazamientos son precisos, éstos resultados y datos tomados se detallan a continuación, en ellos se determina la repetitividad de procesos de maquinado, tolerancias, diseños, acabados, y otros parámetros necesarios para calibrar la máquina hasta dejarla a punto.

Diseño de maquinado proveniente del MastercamX

Para realizar el diseño a maquinar se utilizó el software MastercamX, en él se diseñó un bloque de dimensiones reales de (100 x 100 x 30) mm con el logo “UTA”, para el diseño se utiliza una broca de fresado para acabados de 4 mm de diámetro de acero rápido, la profundidad para el desbaste del material se fija en 5 mm.

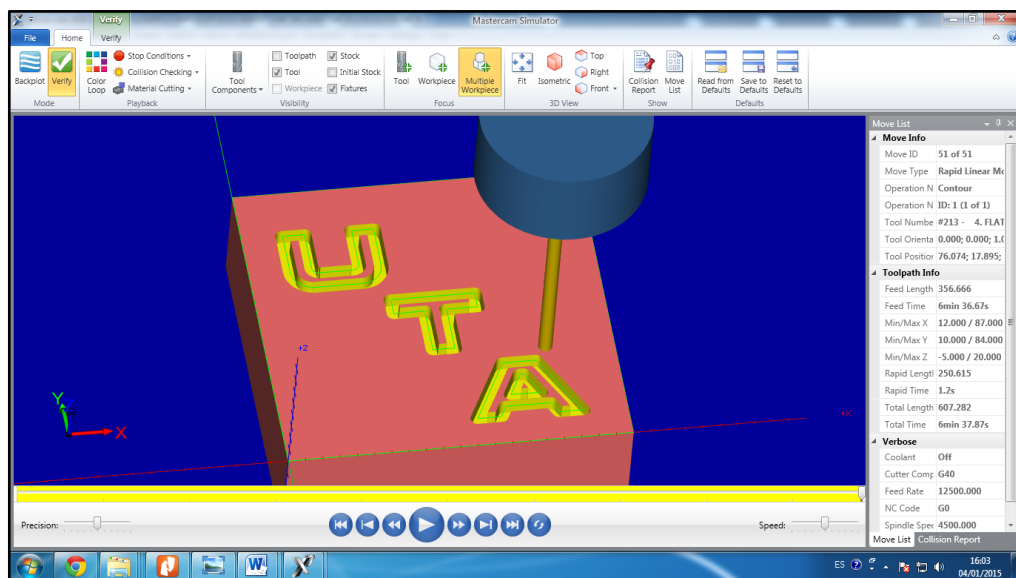


Figura 62: Diseño Logo UTA
Fuente: Jesús Rodríguez

Con el código G generado por el software MastercamX se envía sin presentar problemas al software de control de la máquina fresadora CNC con el cual se obtiene el siguiente maquinado, en un tiempo de 17 minutos programado con una velocidad de avance de 50 mm/min.



Figura 63: Maquinado Logo UTA
Fuente: Jesús Rodríguez

Se estudia analiza las trayectorias realizadas por la máquina fresadora CNC con el diseño generado por el MastercamX obteniendo el siguiente resultado.

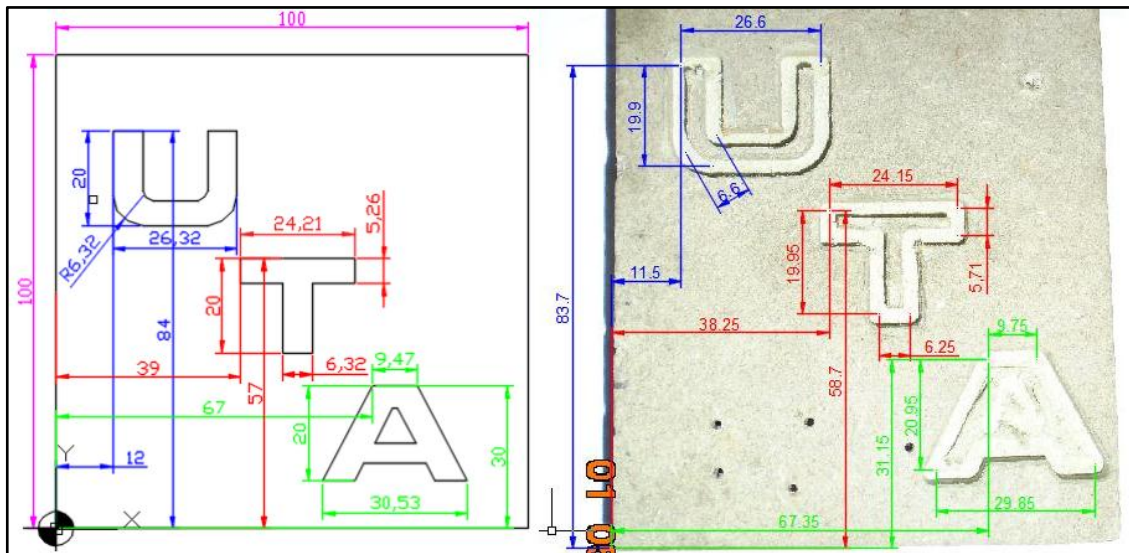


Figura 64: Análisis de medidas
Fuente: Jesús Rodríguez

Las dos gráficas están acotadas con las medidas del diseño del MastercamX y las obtenidas en el maquinado de la máquina fresadora CNC con lo cual podemos sacar la tolerancias.

Para el cálculo del error se utiliza la siguiente fórmula.

$$\text{error} = \frac{|\text{Valor teórico} - \text{Valor medido}|}{\text{Valor teórico}} * 100\% \quad (4)$$

Tabla 22: Análisis de medidas

Medida	Valor Real	Valor medido	Error
Altura U	20.00	19.90	0.50
Radio exterior U	6.32	6.60	4.43
Ancho U	26.32	26.60	1.06
Ubicación (X) U	12.00	11.50	4.17
Ubicación (Y) U	84.00	83.70	0.36
Altura T	20.00	19.95	0.25
Ancho T	24.21	24.15	0.25
Espesor T	5.26	5.71	8.56
Ancho Base T	6.32	6.25	1.11
Ubicación (X) T	39.00	38.25	1.92
Ubicación (Y) T	57.00	58.70	2.98
Altura A	20.00	20.95	4.75
Ancho A	30.53	29.85	2.23
Lado superior	67.00	67.35	0.52
Lado inferior	30.53	29.85	2.23
Ubicación (X) A	67.00	67.35	0.52
Ubicación (Y) A	30.00	31.15	3.83
		ERROR MEDIO (%)	2.33

Fuente: Jesús Rodríguez

Para el análisis de medida se utilizó un pie de rey con 0.05 de precisión con el cual se obtiene los resultados que indican que el error de maquinado es menor a 2.33%, es decir la tolerancia viene dada por la siguiente fórmula:

$$medida\ real = valor\ medido \pm 2.33\%$$

Medida de corriente de consumo en el proceso de maquinado

En el proceso de maquinado se tomó medidas de corriente de consumo en cada uno de los motores que accionan los tres ejes de la máquina fresadora CNC a un intervalo de dos minutos. El siguiente diagrama muestra el modelo utilizado para obtener datos en el equipo CNC.

Los datos obtenidos de corriente son los siguientes.

Tabla 23: Análisis de corrientes

Tiempo de maquinado (minutos)	Corriente de consumo Amperios (A)			
	Motor eje X (A)	Motor eje Y (A)	Motor eje Z (A)	Promedio de corriente
1	0.75	0.45	2.53	1.24
2	0.85	0.65	2.45	1.32
3	1.25	1.56	2.36	1.72
4	1.32	2.35	2.25	1.97
5	1.4	1.45	1.5	1.45
6	1.33	2.23	1.15	1.57
7	1.58	1.54	0.96	1.36
8	2.15	1.24	0.86	1.42
9	2.35	1.16	0.78	1.43
10	1.89	1.85	1.52	1.75
11	1.76	2.64	1.86	2.09
12	2.54	2.45	1.75	2.25
13	2.15	2.05	2.18	2.13
14	2.19	1.18	2.45	1.94
15	0.95	0.86	2.56	1.46
16	0.53	0.65	0.61	0.60

Fuente: Jesús Rodríguez

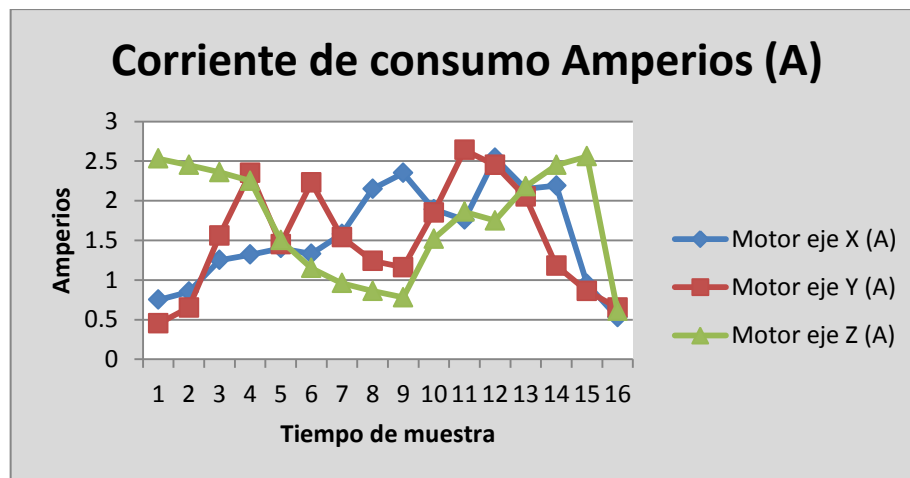


Figura 65: Consumo de corriente

Fuente: Jesús Rodríguez

La grafica de tendencia muestra las corrientes de cada uno de los motores teniendo que la corriente de trabajo del motor del eje Z, consume más corriente al inicio del maquinado como al final del mismo, esto es debido a que los movimiento de aproximación se realiza antes de proceder al maquinado y de retracción de la herramienta de corte al final del proceso.

La corriente de consumo de los motores de los ejes X y Y, fluctúan más cuando cesa la corriente de consumo del motor del eje Z, es decir cuando la herramienta de corte esta fija para desbastar material, la fluctuación de ésta corriente se debe a los cambios de velocidad que sufren éstos motores al realizar los distintos desplazamiento en el proceso de maquinado.

Medida de temperatura de trabajo en el proceso de maquinado

En el proceso de maquinado el driver de los motores a pasos genera calor producido por el consumo de corriente al desplazar los distintos ejes del equipo CNC, éstas temperaturas fueron registradas cada 2 minutos del circuito integrado de potencia L298.

Tabla 24: Análisis de temperatura

Tiempo de maquinado (minutos)	Temperatura de trabajo en Drivers (°C)			
	Driver eje X (°C)	Driver eje Y (°C)	Driver eje Z (°C)	Promedio de temperatura
1	16.2	16.25	16.21	16.22
2	16.5	16.45	16.45	16.47
3	17.32	16.98	18.92	17.74
4	20.15	18.45	20.45	19.68
5	21.56	20.35	22.89	21.60
6	23.45	22.15	24.12	23.24
7	26.48	25.18	26.35	26.00
8	28.51	26.75	28.12	27.79
9	31.45	28.18	31.16	30.26
10	33.56	32.45	32.78	32.93
11	34.12	33.18	34.15	33.82

12	32.16	33.45	35.78	33.80
13	33.45	33.56	36.15	34.39
14	33.46	33.62	37.31	34.80
15	33.96	33.59	38.01	35.19
16	33.85	33.61	38.26	35.24

Fuente: Jesús Rodríguez

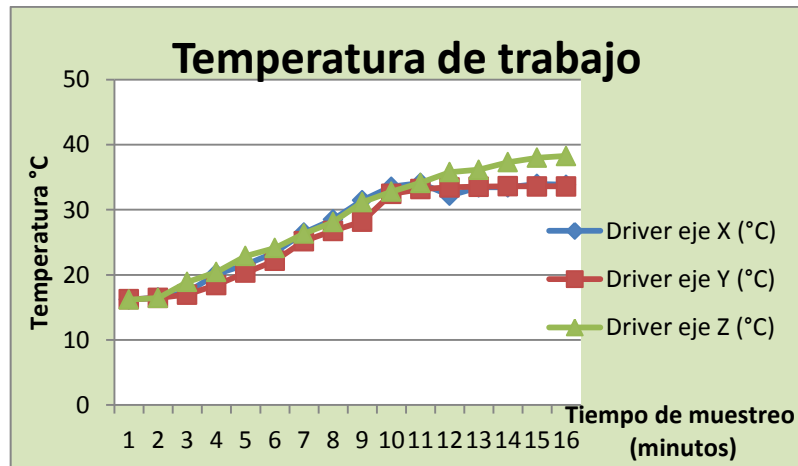


Figura 66: Tendencia de temperaturas

Fuente: Jesús Rodríguez

La grafica anterior indica que las temperaturas muestran una tendencia ascendente conformen avanza el tiempo de mecanizado, la temperatura de trabajo no excede los cuarenta grados lo cual indica que los driver están trabajando sin saturarse.

PRUEBA DE FUNCIONAMIENTO 2

Diseño y maquinado multi-trayectoria

Para realizar el diseño a maquinar se utilizó el software MastercamX, en él se diseñó un bloque de dimensiones reales de (120 x 120 x 18) mm con el diseño de un caballo, para el diseño se utiliza una broca de fresado para acabados de 4 mm de diámetro de acero rápido, la profundidad para el desbaste del material se fija en 5 mm.

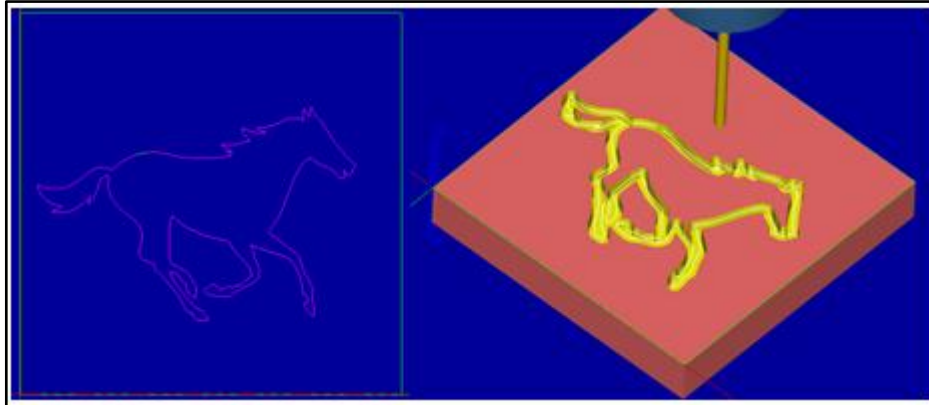


Figura 67: Diseño "Caballo"
Fuente: Jesús Rodríguez

Con el código G generado por el software MastercamX se envía sin presentar problemas al software de control de la máquina fresadora CNC con el cual se obtiene el siguiente maquinado, en un tiempo de 32 minutos programado con una velocidad de avance de 25 mm/min.

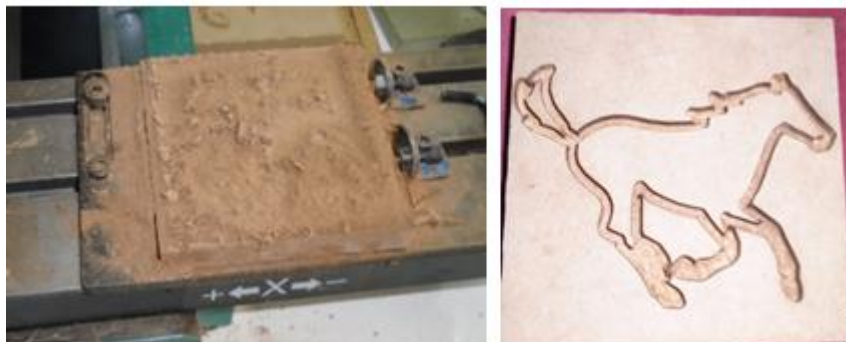


Figura 68: maquinado de "Caballo"
Fuente: Jesús Rodríguez

PRUEBA DE FUNCIONAMIENTO 3

Diseño y maquinado desbaste de superficie

Para realizar el diseño a maquinar se utilizó el software MastercamX, en él se diseñó un bloque de dimensiones reales de (120 x 120 x 18) mm con el diseño el logo Chevrolet, circunferencias, y triangulo, para el diseño se utiliza una broca de fresado para acabados de 6 mm de diámetro de acero rápido, la profundidad para el desbaste del material se fija en 2 mm.

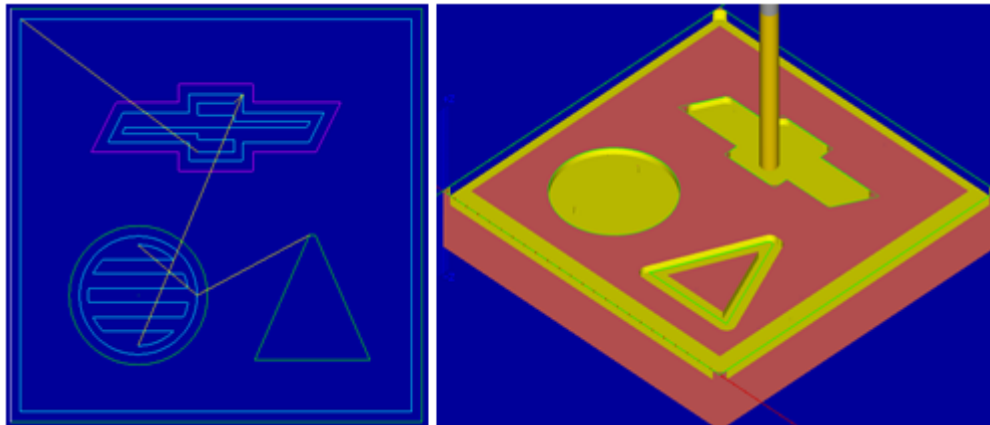


Figura 69: Desbaste de superficie
Fuente: Jesús Rodríguez

Con el código G generado por el software MastercamX se envía sin presentar problemas al software de control de la máquina fresadora CNC con el cual se obtiene el siguiente maquinado, en un tiempo de 25 minutos programado con una velocidad de avance de 25 mm/min.



Figura 70: Desbaste de superficie
Fuente: Jesús Rodríguez

PRUEBA DE FUNCIONAMIENTO 4

Diseño y maquinado desbaste multi-capas

Para realizar el diseño a maquinar se utilizó el software MastercamX, en él se diseñó un bloque de dimensiones reales de (73 x 120 x 18) mm con el diseño de un caballo, para el diseño se utiliza una broca de fresado para acabados de 6 mm de diámetro de acero

rápido, la profundidad para el desbaste del material se fija para la primera capa en 2 mm y para la segunda capa de 5 mm.

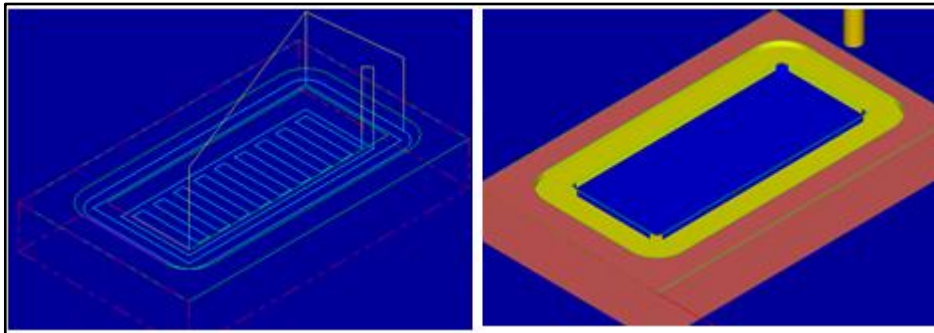


Figura 71: Diseño desbaste multi capa
Fuente: Jesús Rodríguez

Con el código G generado por el software MastercamX se envía sin presentar problemas al software de control de la máquina fresadora CNC con el cual se obtiene el siguiente maquinado, en un tiempo de 28 minutos programado con una velocidad de avance de 25 mm/min.



Figura 72: Desbaste multi capa
Fuente: Jesús Rodríguez

CAPITULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- La implementación del módulo controlador permitió habilitar la máquina fresadora CNC, con éste equipo rehabilitado los estudiantes de la Carrera de Ingeniería Industrial complementarán los conocimientos teóricos prácticos del Área de CAD/CAM.
- La causa por la cual quedó inactividad la tarjeta controladora de la máquina CNC, probablemente se la atribuye a una sobretensión, motivo por el cual varias de las pistas electrónicas de la tarjeta se encuentran fracturadas y algunos elementos que la constituyen se encuentran dañadas.
- En base al análisis de los elementos físicos de la máquina se determinó que las partes mecánicas de la máquina CNC se encuentran en buenas condiciones y funcionando, y se encontró que el desperfecto está en la tarjeta controladora, la cual no recibe ni envía señales de los diferentes sensores y actuadores que controlan los componentes mecánicos y electrónicos de la máquina.
- Los circuitos de control están diseñados de forma que emulen el funcionamiento de la tarjeta real y la interacción entre el usuario por medio del software HMI, los diferentes códigos de programación de los micros controladores así como de la tarjeta Arduino están diseñados con el propósito que su código sea abierto y fácilmente modificable en el caso de necesitar realizar actualizaciones o correcciones en el código.
- Para la fabricación de la tarjeta controladora se utilizó componentes electrónicos de fácil acceso en el mercado local, ésta fue diseñada de manera que ejecute el control y supervisión de todos los componentes de la máquina,

el diseño de ésta tarjeta es modular lo que permite instalar y manipular sin dificultad en la caja eléctrica.

- El software de control cuenta con la distribución lógica de botones virtuales con el propósito de facilitar la manipulación, control y ejecución de tareas de posicionamiento y de maquinado del equipo CNC de acuerdo a las instrucciones del operario, la programación de este software fue diseñada en la plataforma LABVIEW cuyo permiso de Licencia es de propiedad de la Facultad en donde se encuentra ésta máquina.
- Las pruebas realizadas en cuanto a funcionamiento y rendimiento, tanto de la tarjeta controladora como del software de control, pasaron los requerimientos del diseño, resolución y precisión con las que fue diseñada la máquina originalmente, esto permitió que el equipo realice su trabajo de maquinado respondiendo correctamente a las instrucciones de códigos G de las piezas mecánicas de prueba.

5.2. RECOMENDACIONES

- Revisar los manuales de instrucciones y planos eléctricos antes de realizar mantenimientos en la máquina CNC, ésta operación debe ser realizado periódicamente por personal debidamente entrenado y calificado, revisando que el equipo éste apagado y desconectado de la fuente de alimentación, con el propósito de evitar daños físicos al equipo y la integridad del personal de mantenimiento.
- Se sugiere crear un programa de mantenimiento preventivo que garantice el funcionamiento idóneo de la máquina CNC, lubricando constantemente los componentes mecánicos como rieles y tornillos sin fin, para evitar el desgaste físico de los componentes mecánicos y prevenir el recalentamiento de los motores de los tres ejes.
- Se recomienda no dejar sin supervisión la máquina cuando se encuentre realizando un proceso de maquinado, verificando que los límites de la máquina no sean sobrepasados para evitar posibles colisiones o fallas en el proceso, hay

que tener cuidado con los movimientos de la máquina CNC sobre todo cuando se está ejecutando instrucciones “G00” que son movimientos rápidos.

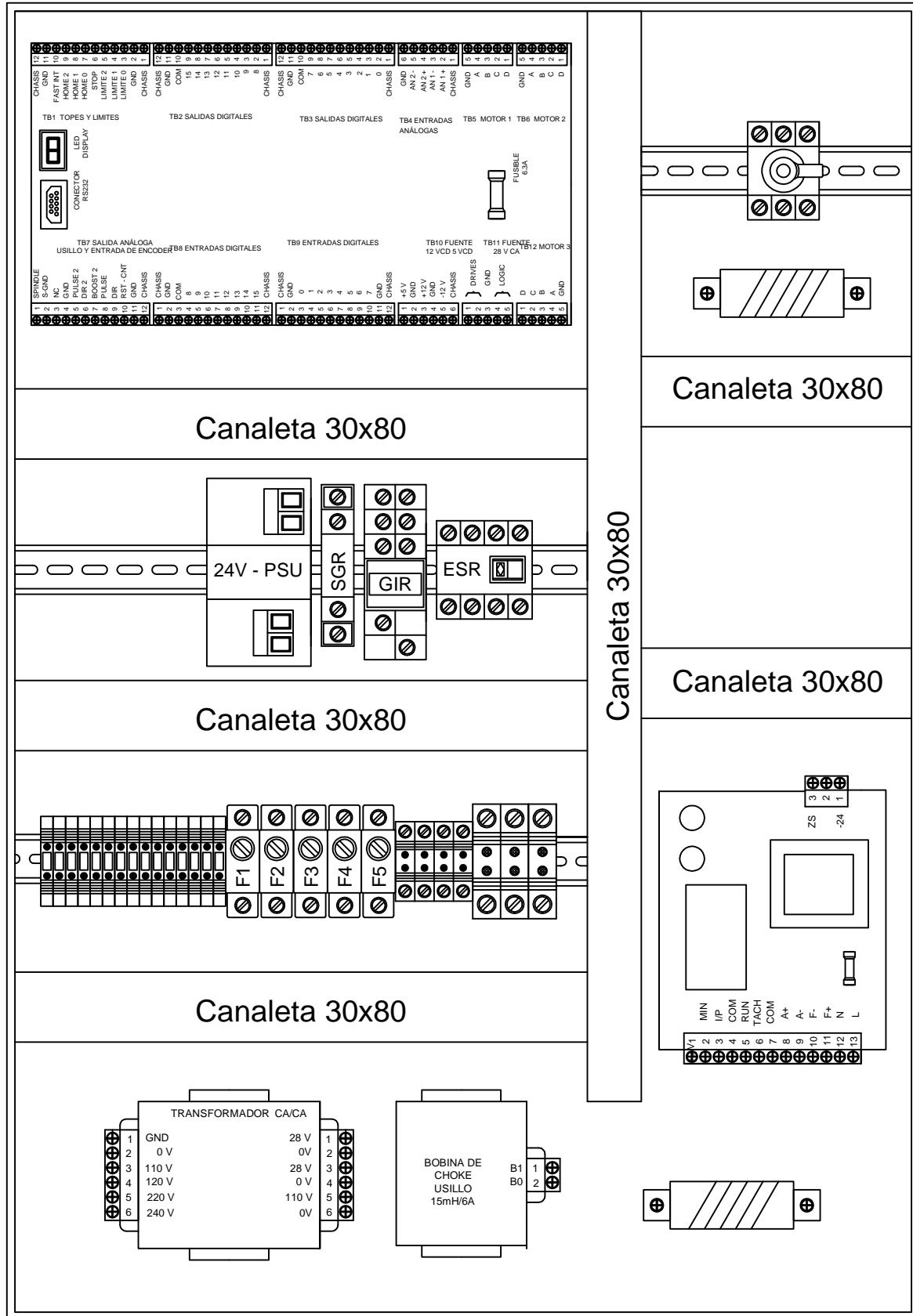
- Asegurarse que la herramienta de corte este fija y segura al mandril del husillo, comprobando también los parámetros de velocidad de avance y velocidad de corte en relación al material a maquinar, es recomendable seguir las guías de laboratorio para que los estudiantes hagan un uso adecuado y ordenado del equipo.
- Una vez terminado el proceso de maquinado se debe realizar tareas de limpieza y lubricación revisando que no queden restos de material sobre los rieles.

REFERENCIAS

- [1] Carlos San Juan, *Historia de la Ciencia y de la Técnica*. Madrid - España: Akal S.A., 1993.
- [2] Julian Rodriguez Montes, *Sistemas Industriales para materiales Metálicos*, Segunda ed. Madrid - España, 2009.
- [3] Alejandro Mejía Sierra, "Desarrollo y construcción de maquinas CNC para mejorar la competitividad en Colombia," *Revista Épsilon*, no. 20, pp. 193-214, Enero-Junio 2013.
- [4] Edwin Villarreal, "PROTOTIPO CNC PARA EL TORNEADO EN SERIE DE METALES," *Umbral Científico*, no. 12, pp. 8-19, Junio 2008.
- [5] Francisco Cruz Teruel, "Planificación de la Producción de la Cadena en un Entorno de Personalización en Masa.," *Metales & Metalurgia*, no. 37, p. 24, Abril 2013, Artículo.
- [6] Raúl Sanchez, "Diseño y construcción de un router CNC para la fabricación de puertas MDF," *Repositorio digital ESPE - Latuncunga*, pp. 96-106, Diciembre 2008.
- [7] Herrera Abda Daniel, *Diseño y construcción de una máquina para la elaboración de circuitos impresos basado en CNC*. Quito, Ecuador: Escuela Politécnica Nacional, 2012.
- [8] Arpi Trujillo, *Sistema de control para máquina CNC Laser*. Cuenca, Ecuador: Universidad Politécnica Salesiana, 2013.
- [9] Jose carlos Gallego, *Montaje de Componentes Informáticos*. España: Editex, S.A., 2010.
- [10] José Rafael Lajara, *LabVIEW entorno grafico de programación*, Segunda ed. Barcelona - España: Marcombo S.A., 2011.
- [11] Óscar Torrente, *ARDUINO. Curso práctico de formación*, Primera ed. España: RC Libros, 2013.
- [12] Katsuhiko Ogata, *Ingeniería de Control Moderna*, 2010th ed. Madrid: Person Educación, 2010.
- [13] Raquel Florez López, *Redes Neuronales*, Primera Edición ed. España: Netbiblo S.L., 2008.

- [14] Cristobal Raya, *Tecnología de sistemas de control*, Primera Edición ed. España: Edicions UPC de la Universidad Politécnica de Catalunya, 2004.
- [15] Deitel Harvey, *Como Programar C++*, Cuarta ed. Mexico: Pearson Educación, 2003.
- [16] Fernando Valdés Pérez, *Microcontroladores: Fundamentos y Aplicaciones con PIC*, Primera Edición ed. España: MARCOMBO, S.A., 2007.
- [17] Thomas Vollmann, *Sistemas de planificación y Control de la fabricación*. España: McGraw - Hill Education, 2005.
- [18] Fernando Sevillano Calvo, Folleto de Electricidad - Electrónica, 2010-2011.

Anexos



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

DISEÑADOR: Jesús Rodríguez

FECHA: 09/03/2015

ANEXO 1

TÍTULO:

DISTRIBUCIÓN CAJA ELÉCTRICA

PROYECTO:

MÁQUINA FRESADORA CNC

A4

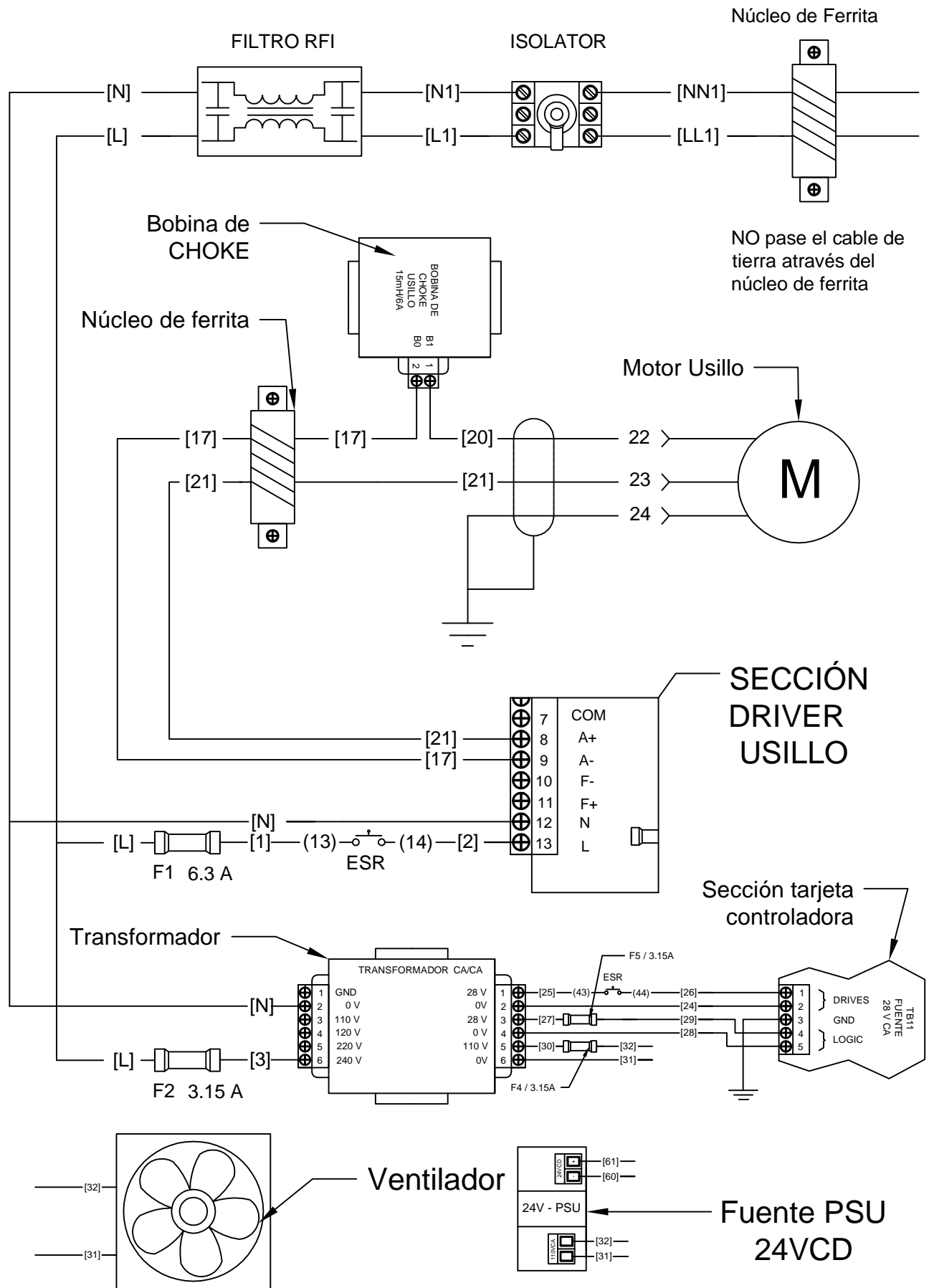
ANEXO 1

NOTA

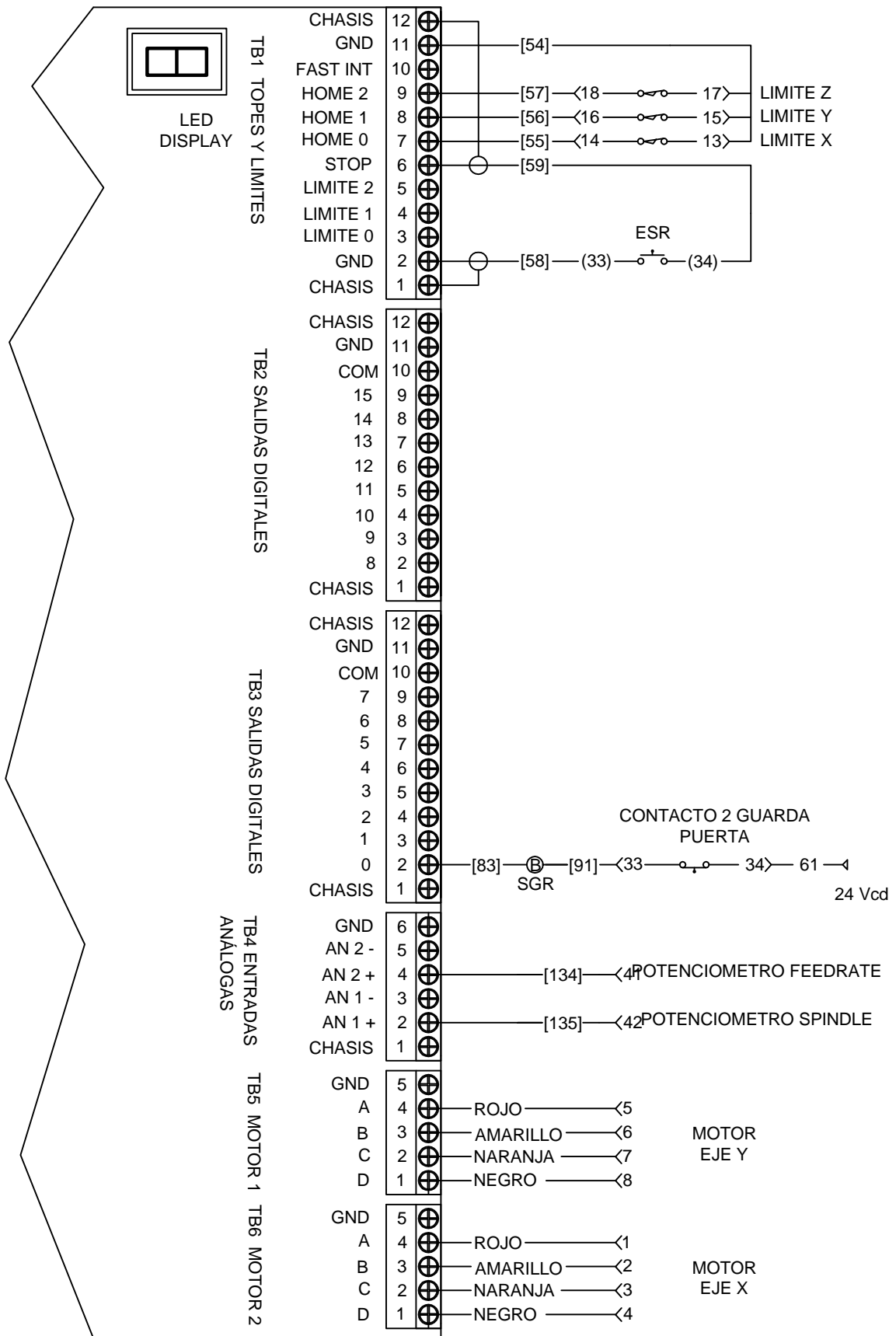
ESCALA: 1:10

PESO: NO determinado

HOJA 1 DE 31



UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO:	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		Esquema general de alimentación	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO:	
DISEÑADOR: Jesús Rodríguez	ANEXO 2	MÁQUINA FRESADORA CNC	A4
NOTA	ESCALA: 1:10	PESO: NO determinado	ANEXO 2
			HOJA 2 DE 31



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

DISEÑADOR: Jesús Rodríguez

FECHA: 09/03/2015

ANEXO 3

TÍTULO:

TARJETA CONTROLADORA ORIGINAL (Sección 1 de 2)

PROYECTO:

MÁQUINA FRESADORA CNC

A4

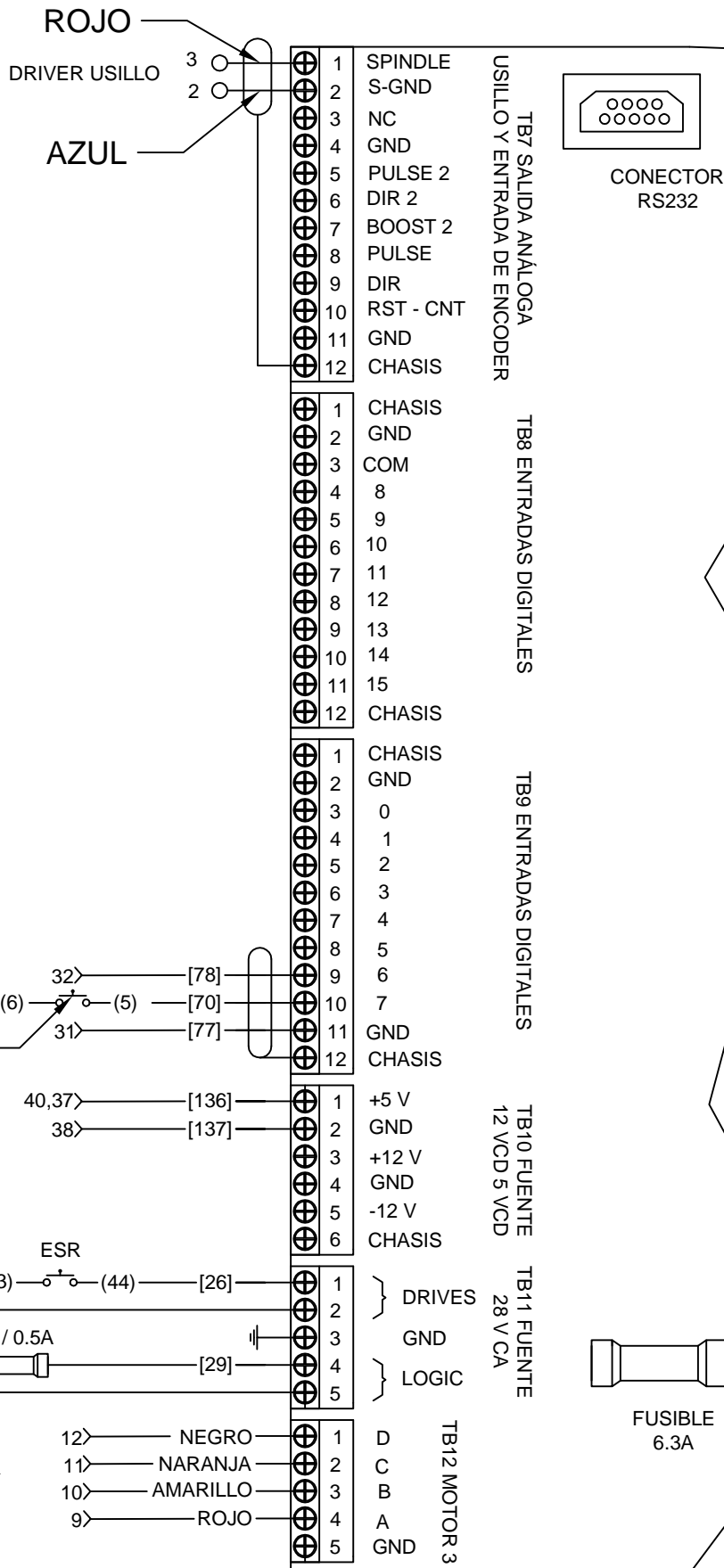
ANEXO 3

NOTA

ESCALA: 1:10

PESO: NO determinado

HOJA 3 DE 31



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

FECHA: 09/03/2015

DISEÑADOR: Jesús Rodríguez

ANEXO 3

TÍTULO:

TARJETA CONTROLADORA ORIGINAL (Sección 2 de 2)

PROYECTO:

MÁQUINA FRESADORA CNC

A4

ANEXO 3

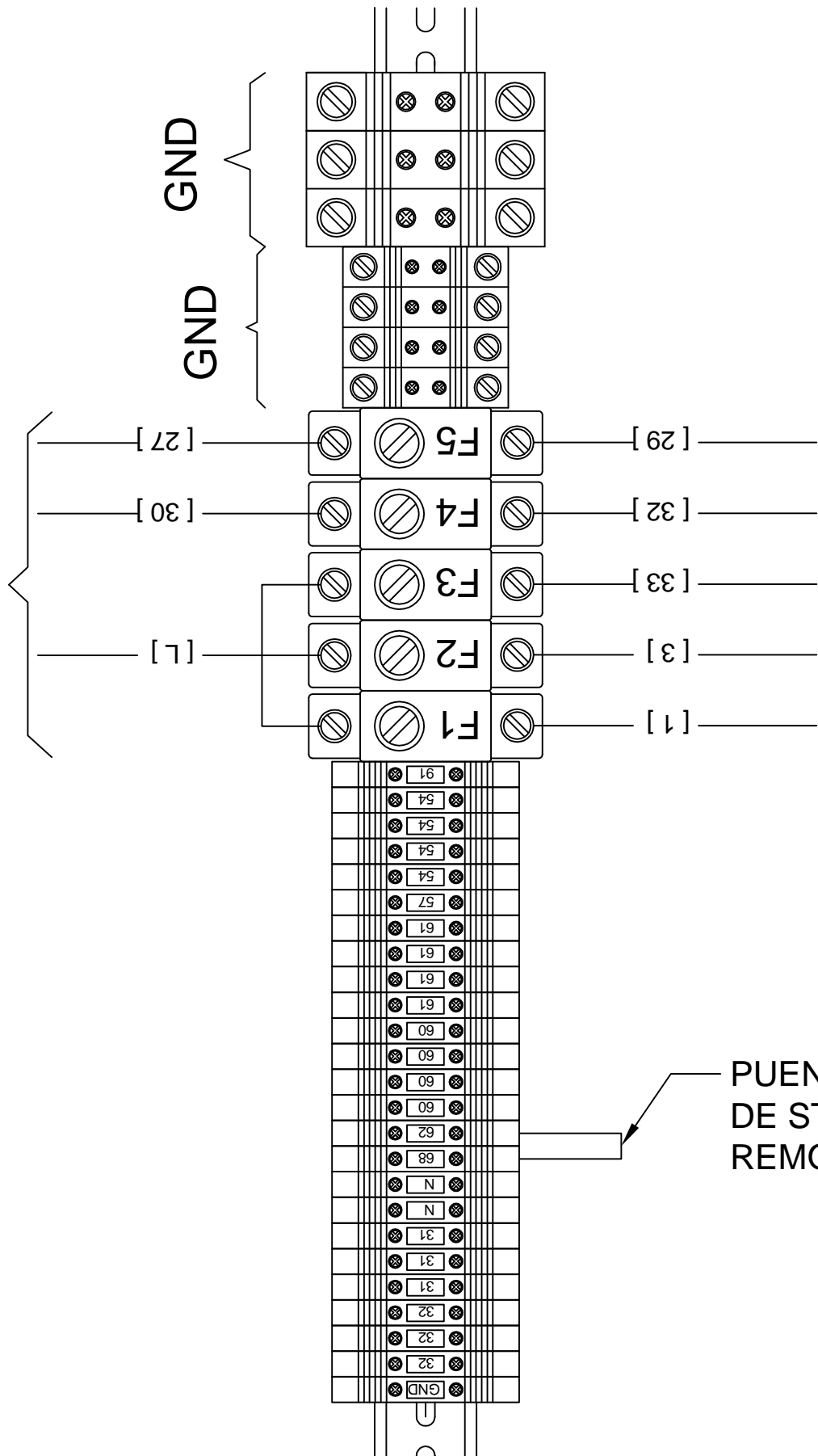
NOTA

ESCALA: 1:10

PESO: NO determinado

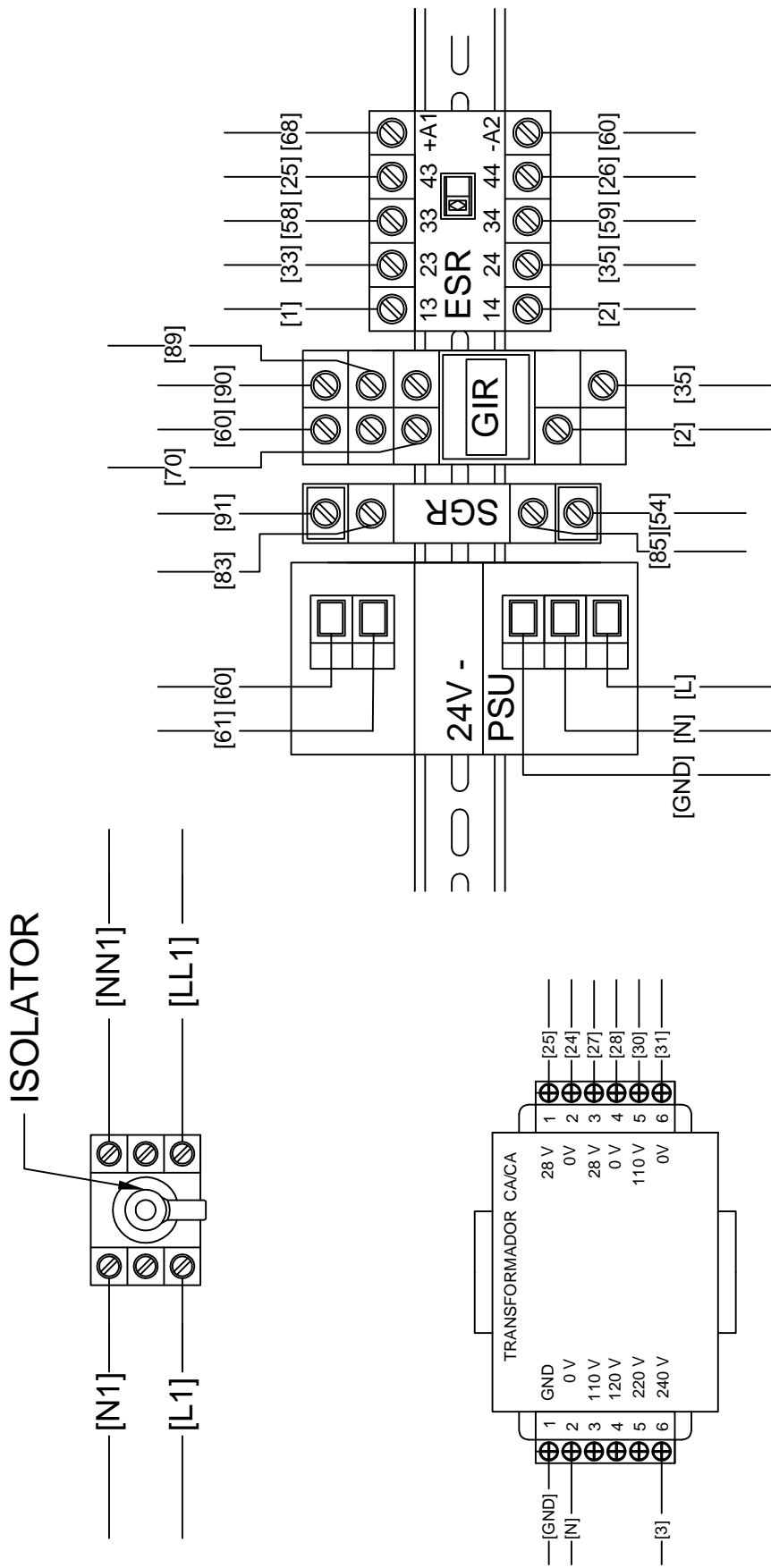
HOJA 4 DE 31

PORTA FUSIBLES



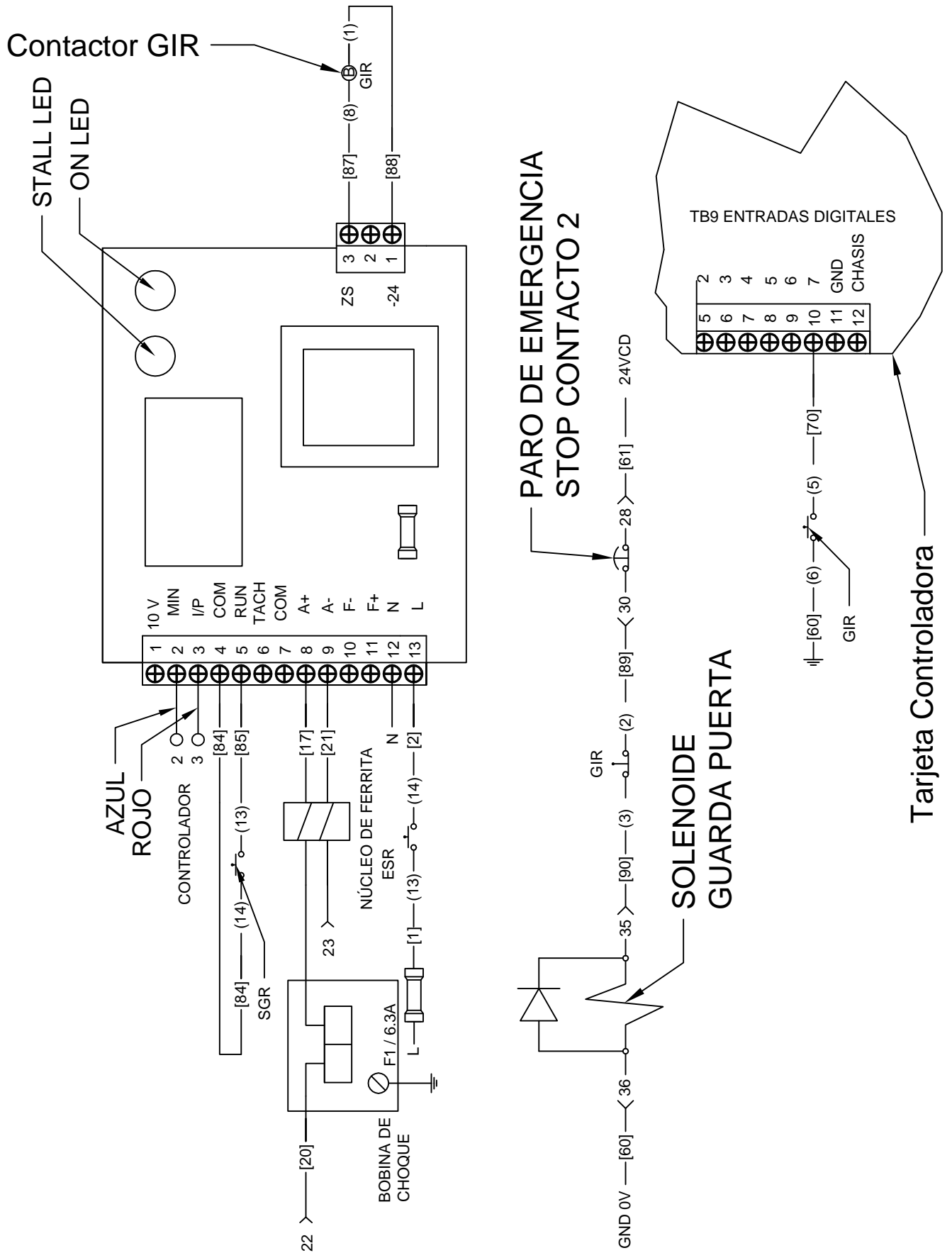
PUENTE DE STOP REMOTO

UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: DETALLE DE CABLEADO DE BORNERAS Y CONTACTORES	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		1 DE 2	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO: MÁQUINA FRESADORA CNC	A4
DISEÑADOR: Jesús Rodríguez	ANEXO 4		ANEXO 4
NOTA	ESCALA: 1:10	PESO: NO determinado	HOJA 5 DE 31



- FUSIBLES
- F1 DRIVER USILLO 6.3 AMPERIOS
 - F2 CONTROL DE TRANSFORMADOR 3.15 AMPERIOS
 - F3 ACCESORIOS CAMBIO DE HERRAMIENTA 3.15 AMPERIOS
 - F4 CIRCUITO 110VCA 3.15 AMPERIOS
 - F5 TARJETA CONTROLADORA PARTE LOGICA 3.15 AMPERIOS

UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: DETALLE DE CABLEADO DE BORNERAS Y CONTACTORES	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		2 DE 2	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO: MÁQUINA FRESADORA CNC	
DISEÑADOR: Jesús Rodríguez	ANEXO 4	A4	
NOTA	ESCALA: 1:10	PESO: NO determinado	ANEXO 4
			HOJA 6 DE 31



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

FECHA: 09/03/2015

DISEÑADOR: Jesús Rodríguez

ANEXO 5

TÍTULO:

TARJETA DRIVER
DE USILLO EN MÁQUINA CNC

PROYECTO:

MÁQUINA FRESADORA CNC

A4

ANEXO 5

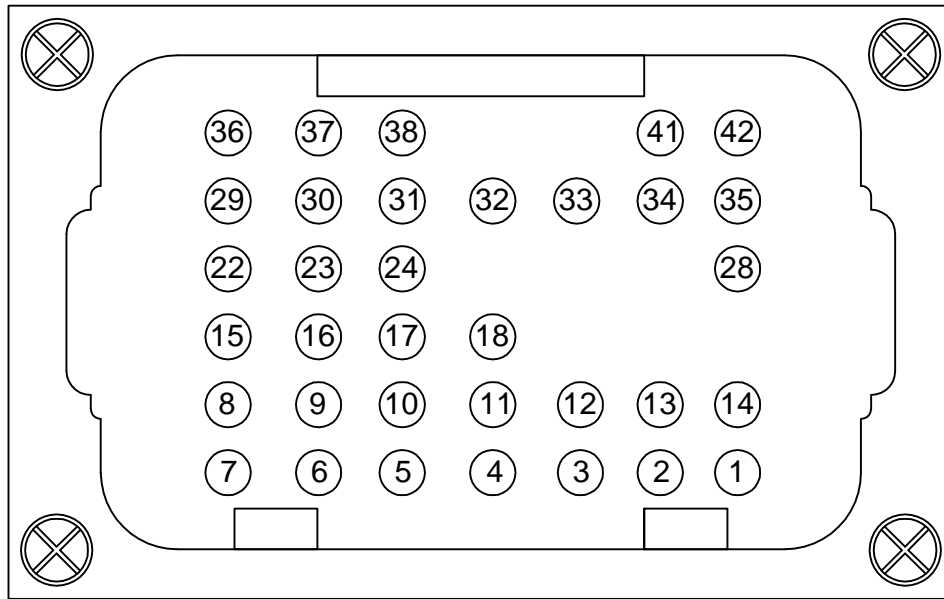
NOTA

ESCALA: 1:10

PESO: NO determinado

HOJA 7 DE 31

CONECTOR DE 42 CNC



CONECTOR DE 42 PINES

# PIN	COLOR DE CABLE	NÚMERO DE CABLE	DESCRIPCIÓN	# PIN	COLOR DE CABLE	NÚMERO DE CABLE	DESCRIPCIÓN
1	ROJO		MOTOR EJE X	22	MARRÓN	20	MOTOR USILLO
2	AMARILLO			23	AZUL	21	
3	NARANJA			24	VERDE/AMARILLO	GND	
4	NEGRO			25			
5	ROJO		26				
6	AMARILLO		27				
7	NARANJA		MOTOR EJE Y	28	ROJO	61	24 V
8	NEGRO			29	AZUL	62	STOP-1
9	ROJO		MOTOR EJE Z	30	AMARILLO	89	STOP-2
10	AMARILLO			31	ROJO	77	GUARDA CONTACTO -1
11	NARANJA			32	AZUL	78	GUARDA CONTACTO -2
12	NEGRO		LÍMITE X	33	ROJO	91	GUARDA SOLENOIDE
13	ROJO	54		34	AZUL	61	5 V
14	AMARILLO	55	LÍMITE Y	35	ROJO	90	0V-GND
15	ROJO	54		36	AZUL	60	
16	AMARILLO	56	LÍMITE Z	37	ROJO	136	
17	ROJO	54		38	AZUL	137	
18	AMARILLO	57		39			
19				40			
20				41	ROJO	134	FEEDRATE
21				42	AZUL	135	USILLO

UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

FECHA: 09/03/2015

DISEÑADOR: Jesús Rodríguez

ANEXO 6

TÍTULO:

CONECTOR DE
42 VIAS EN MÁQUINA CNC

PROYECTO:

MÁQUINA FRESADORA CNC

A4

ANEXO 6

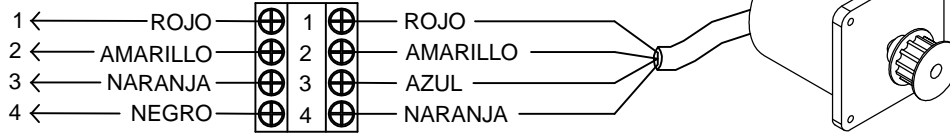
NOTA

ESCALA: 1:10

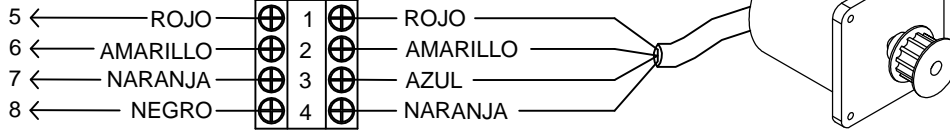
PESO: NO determinado

HOJA 8 DE 31

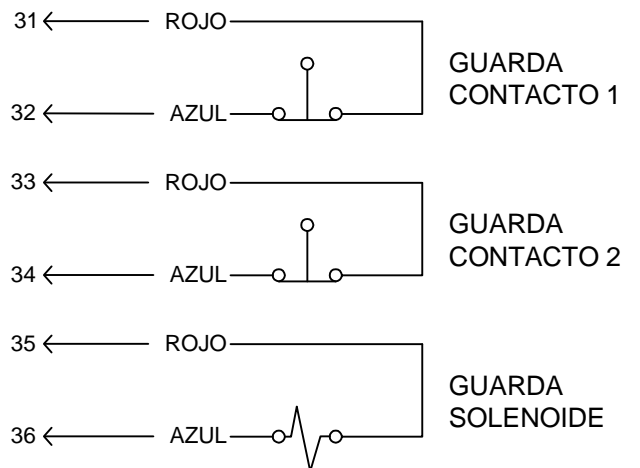
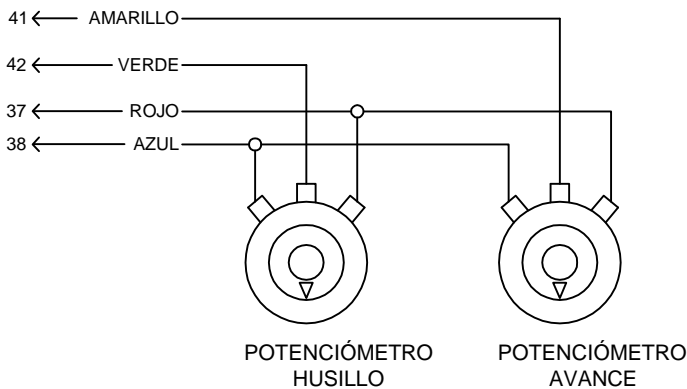
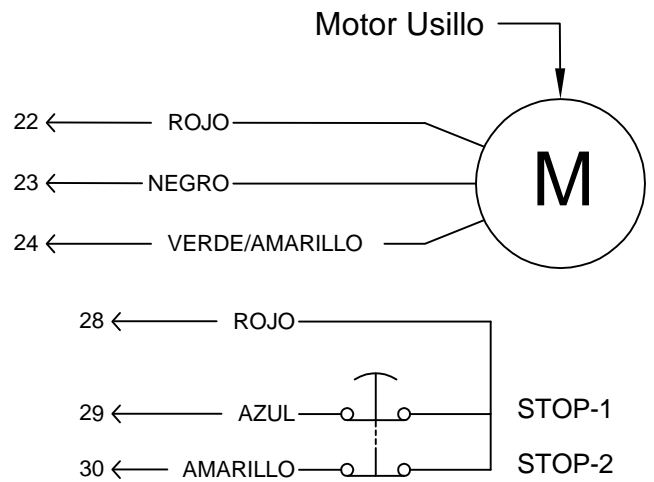
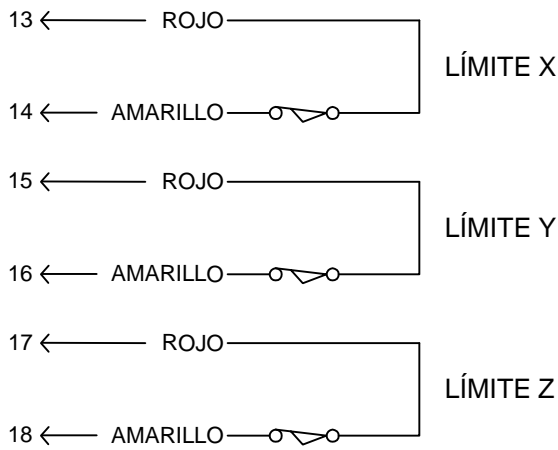
MOTOR EJE X



MOTOR EJE Y



MOTOR EJE Z



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

FECHA: 09/03/2015

DISEÑADOR: Jesús Rodríguez

ANEXO 7

TÍTULO:

CABLEADO DE MOTORES, ACCESORIOS DE MÁQUINA CNC

PROYECTO:

MÁQUINA FRESADORA CNC

A4

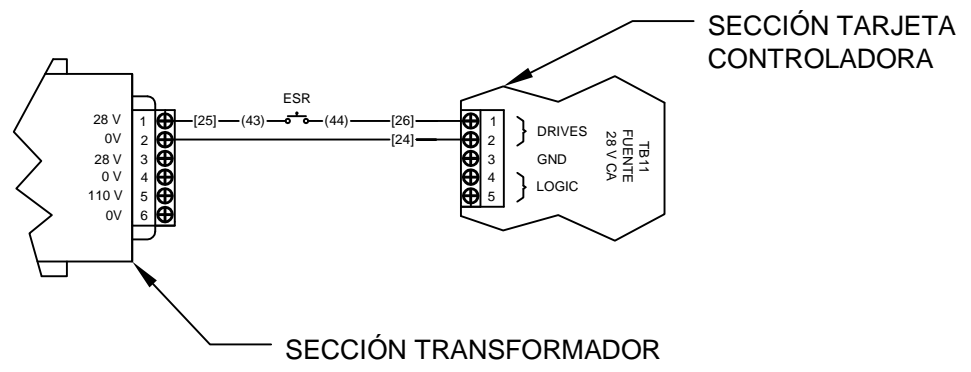
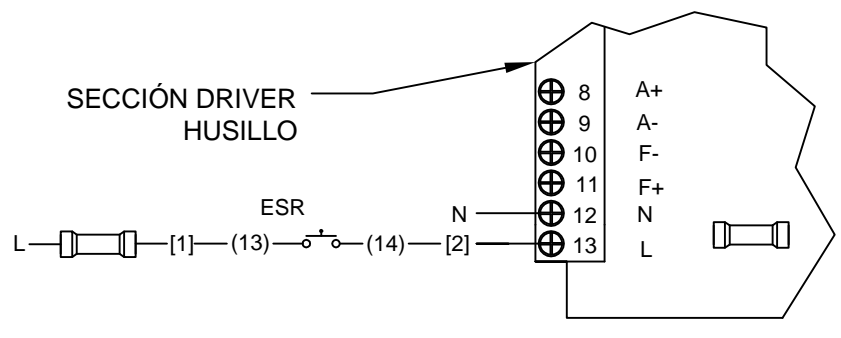
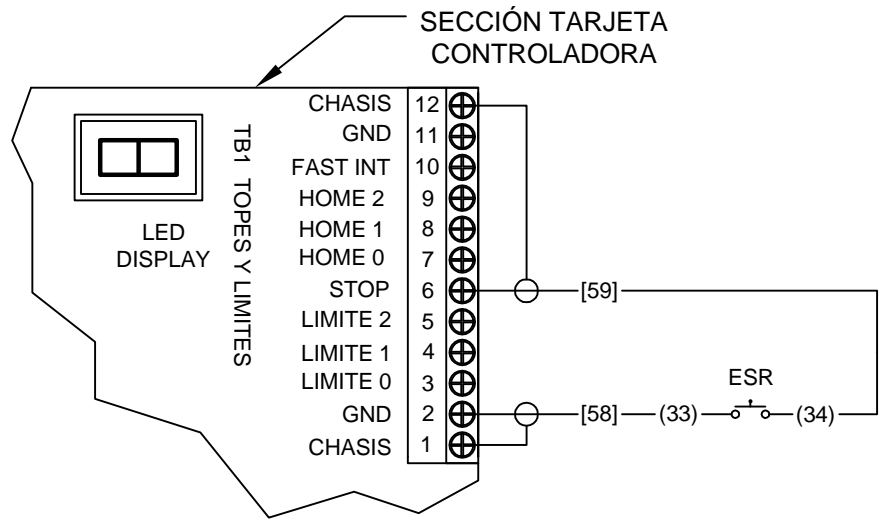
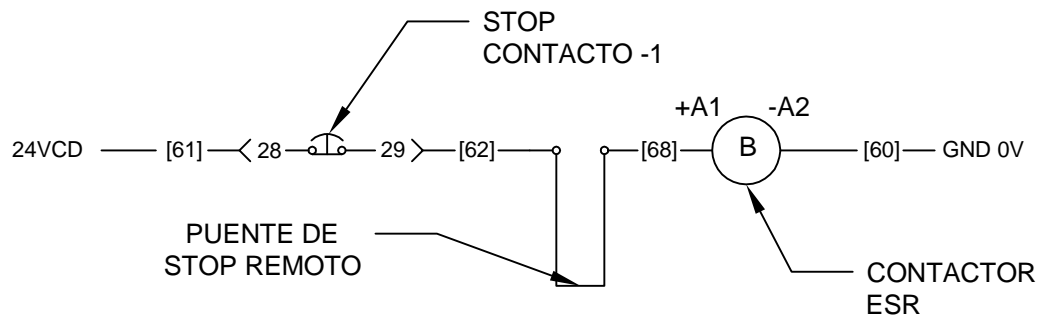
ANEXO 7

NOTA

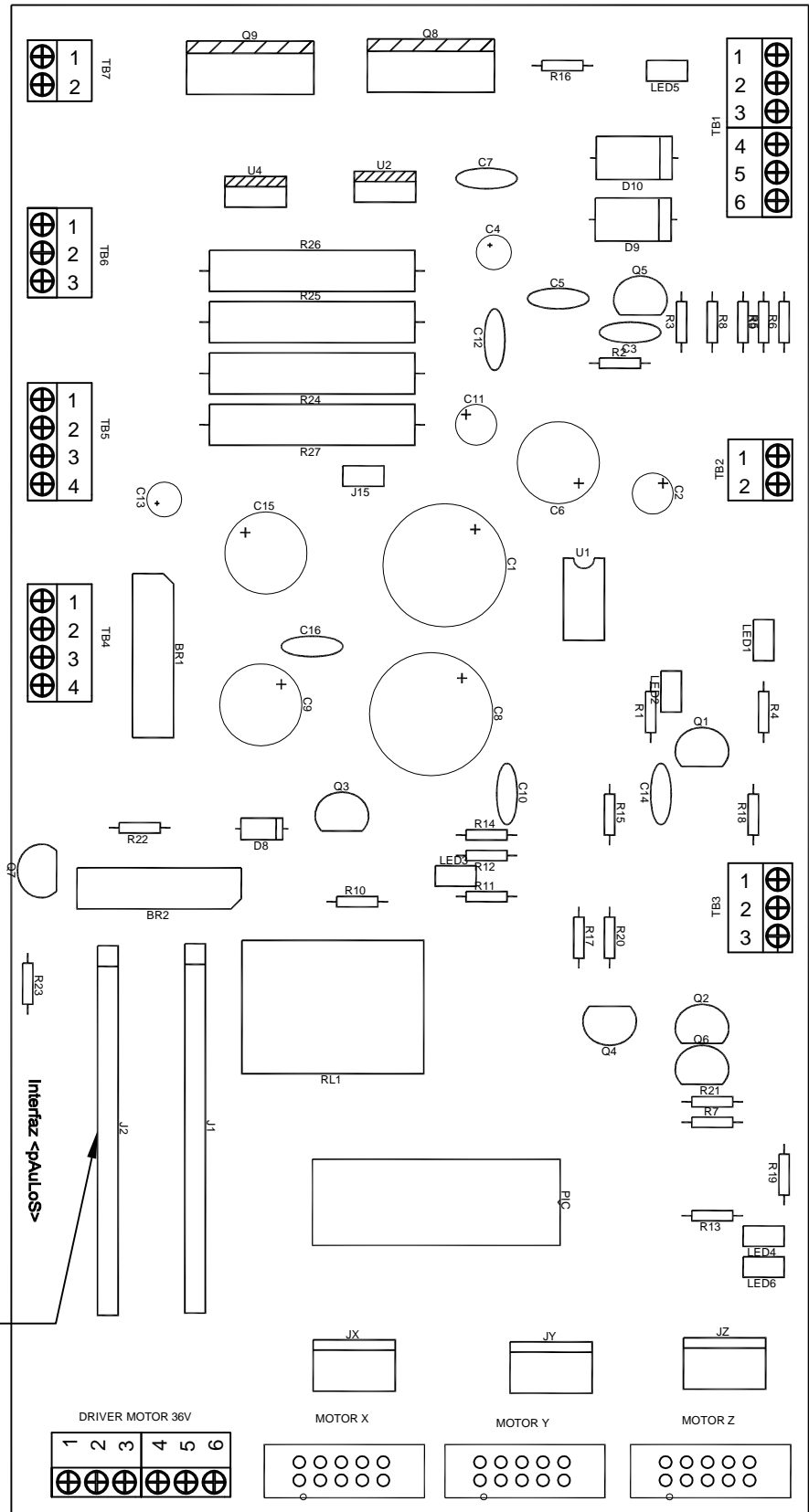
ESCALA: 1:10

PESO: NO determinado

HOJA 9 DE 31



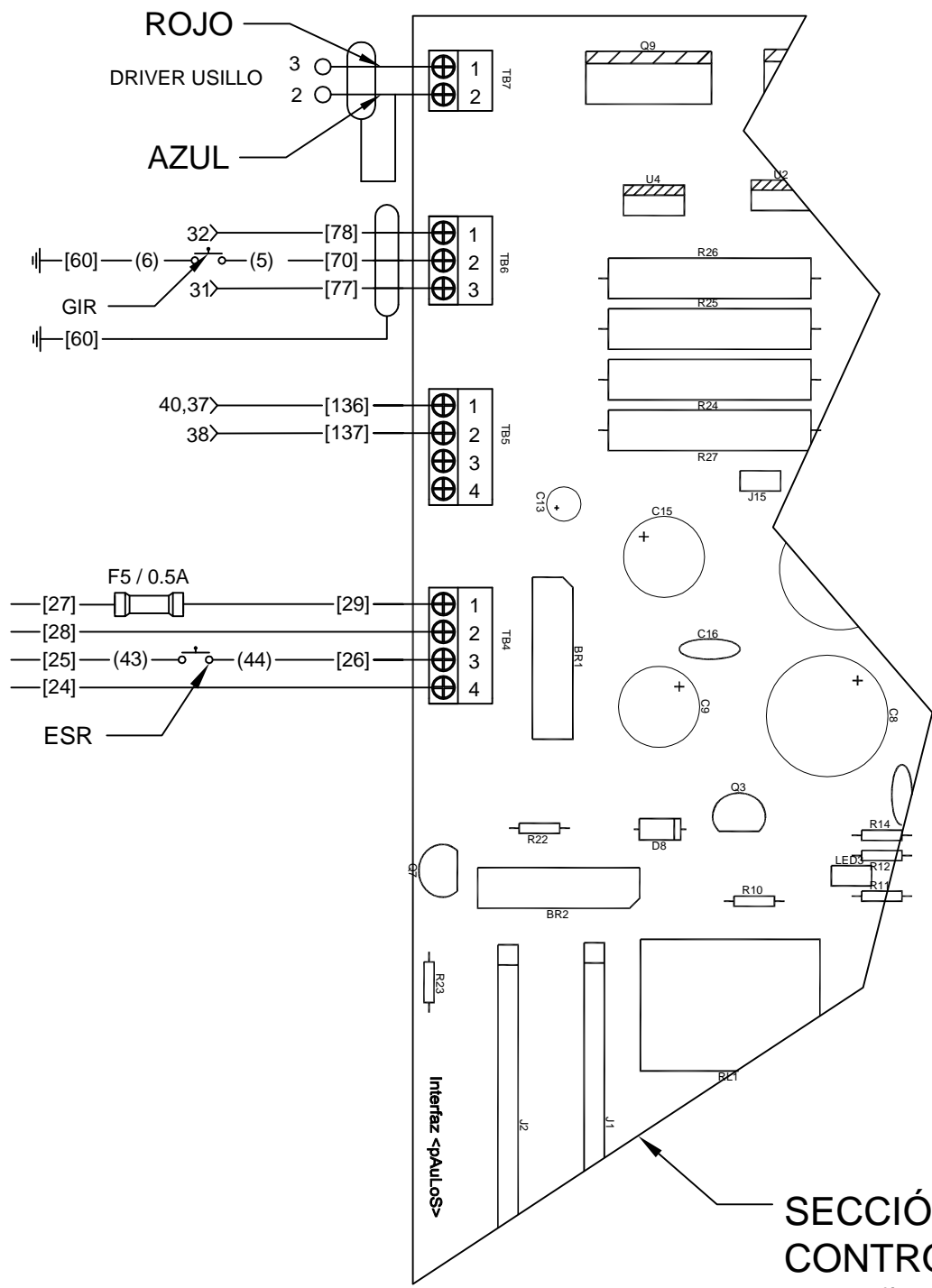
UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: CIRCUITO DE PARO DE EMERGENCIA	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN			
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO: MÁQUINA FRESADORA CNC	
DISEÑADOR: Jesús Rodríguez	ANEXO 8		A4
NOTA	ESCALA: 1:10	PESO: NO determinado	ANEXO 8
			HOJA 10 DE 31



Conector de tarjeta de comunicación

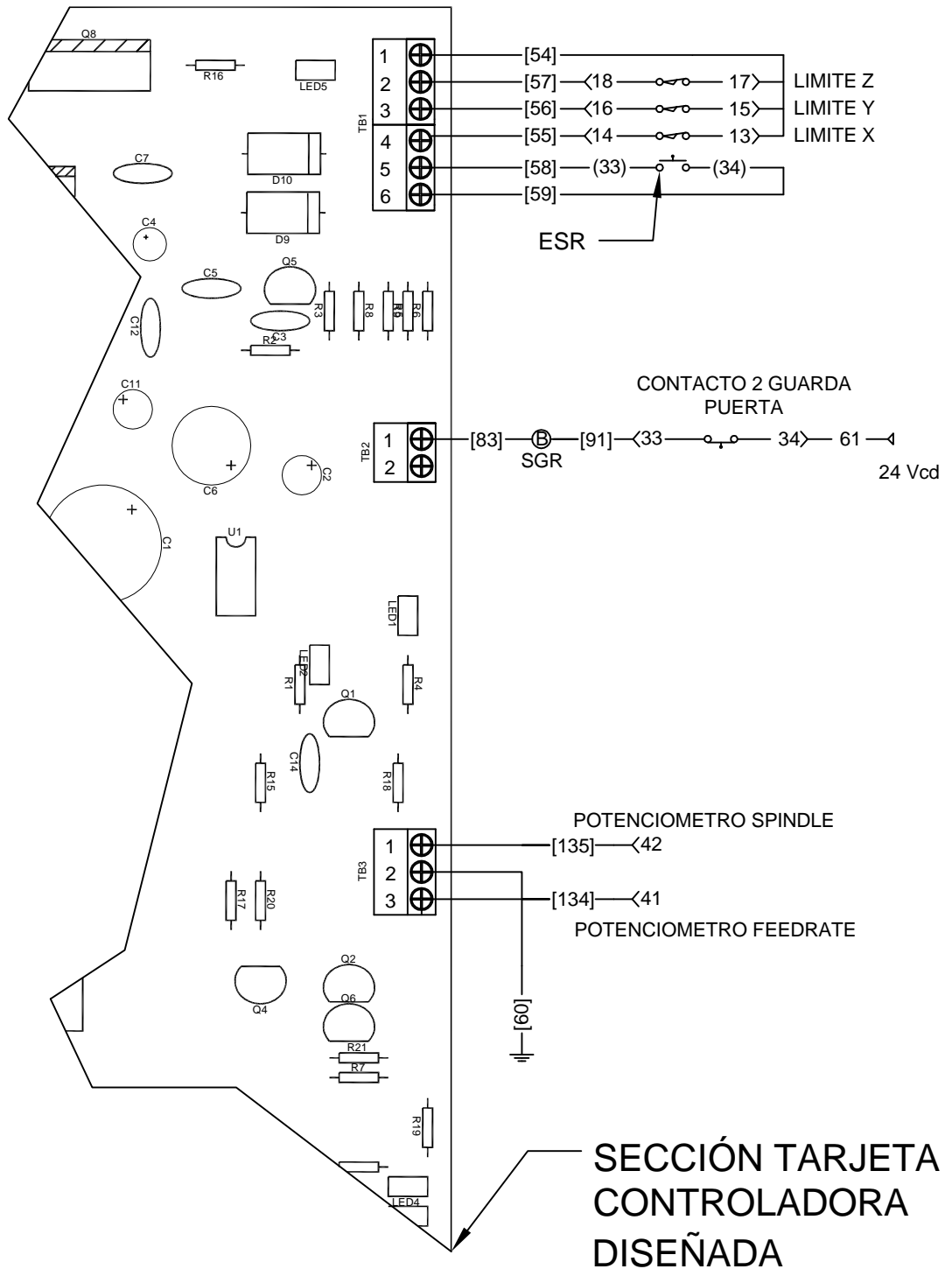
Interfaz <Paulos>

UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: TARJETA CONTROLADORA MÓDULO 1 (Detalle 1 de 4)	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		FECHA: 09/03/2015	
REVISIÓN: ING. Wilian López	ANEXO 9		PROYECTO: MÁQUINA FRESADORA CNC
DISEÑADOR: Jesús Rodríguez			A4
NOTA	ESCALA: 1:10	PESO: NO determinado	HOJA 11 DE 31
		ANEXO 9	

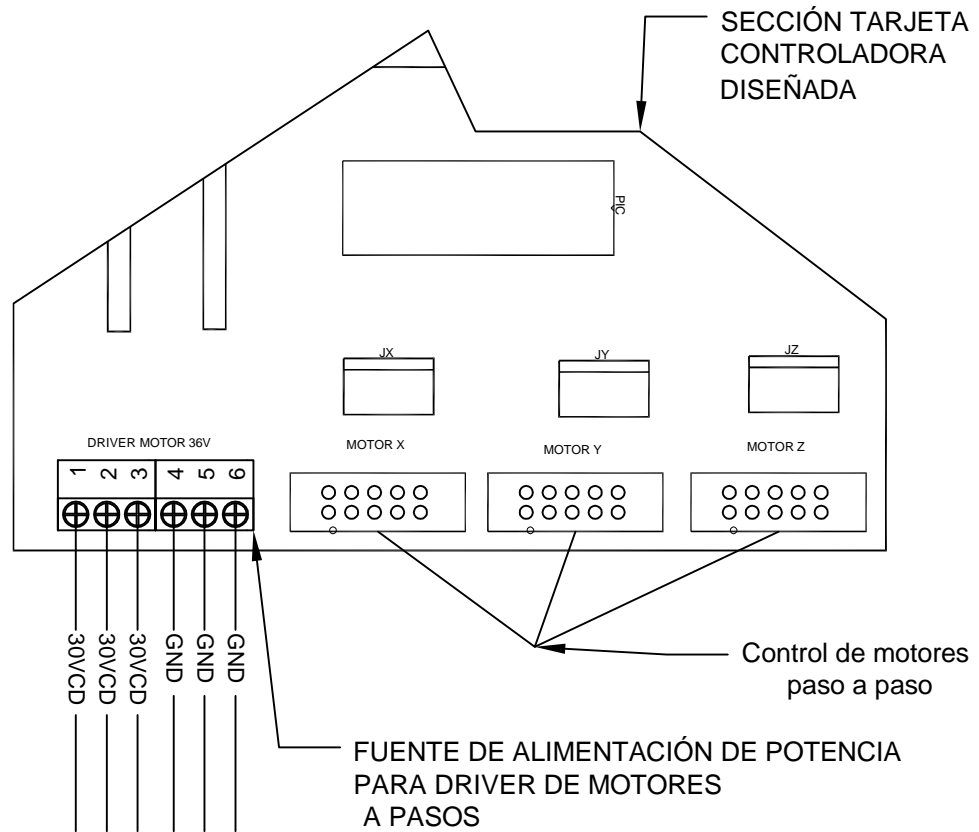


SECCIÓN TARJETA CONTROLADORA DISEÑADA

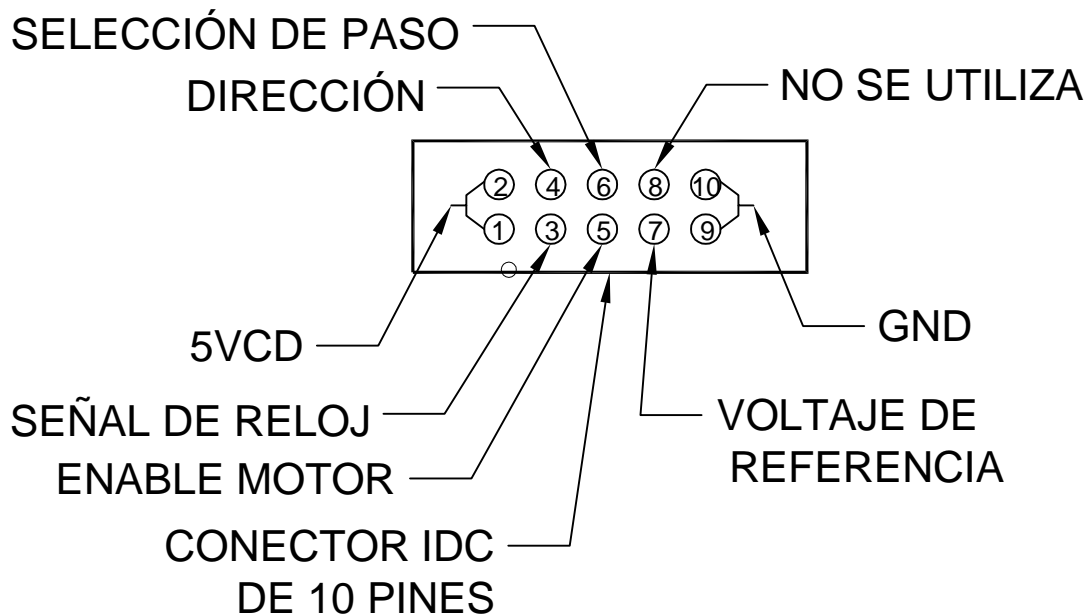
UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: TARJETA CONTROLADORA	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		MÓDULO 1 (Detalle 2 de 4)	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO: MÁQUINA FRESADORA CNC	A4
DISEÑADOR: Jesús Rodríguez	ANEXO 9		ANEXO 9
NOTA	ESCALA: 1:10	PESO: NO determinado	HUJA 12 DE 31



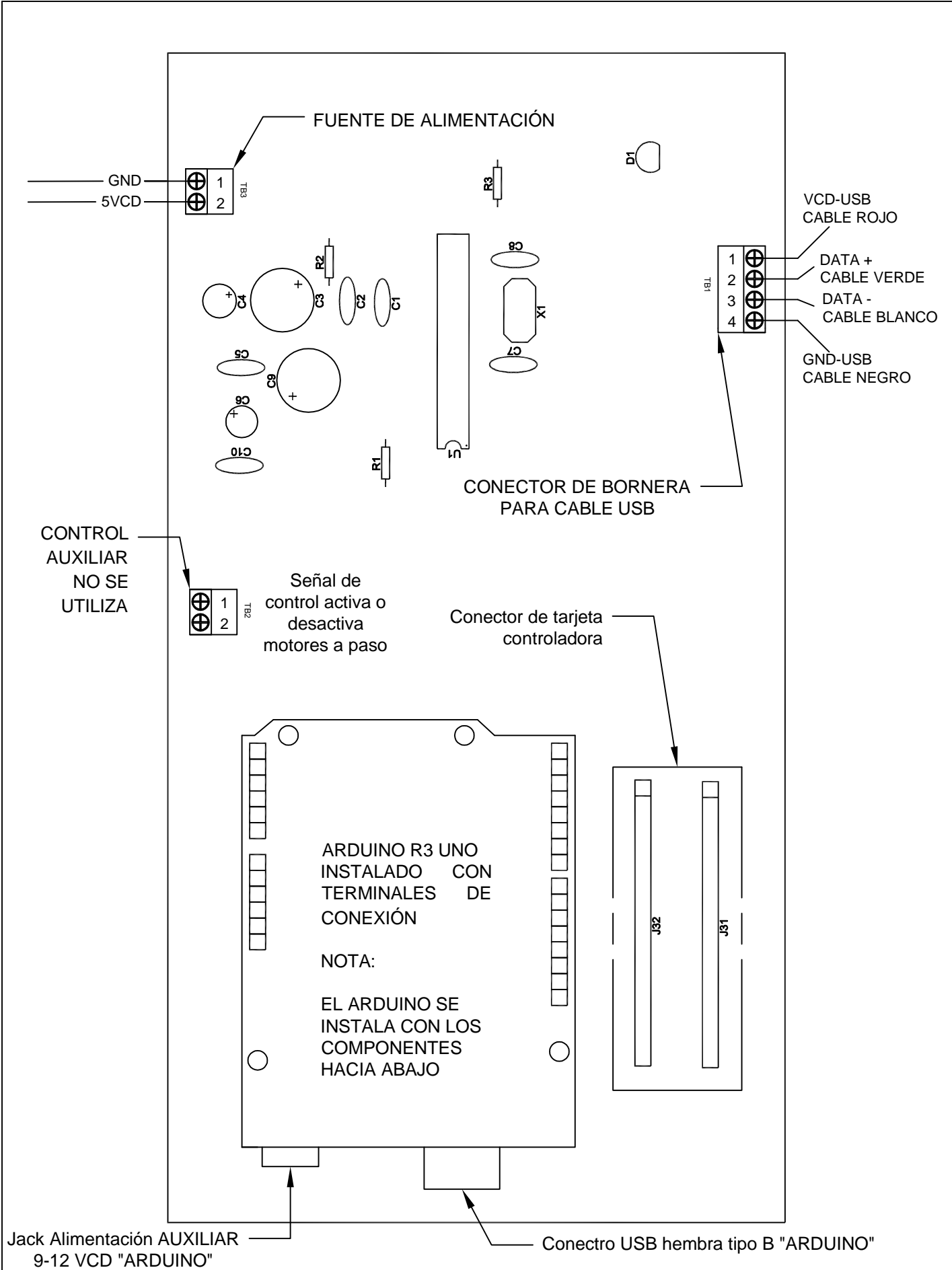
UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: TARJETA CONTROLADORA MÓDULO 1 (Detalle 3 de 4)	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		PROYECTO: MÁQUINA FRESADORA CNC	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	ANEXO 9	A4
DISEÑADOR: Jesús Rodríguez			ANEXO 9
NOTA	ESCALA: 1:10	PESO: NO determinado	HUJA 13 DE 31



CONECTOR PARA CONTROL DEL DRIVER DE MOTORES PASO A PASO DE LOS EJES X, Y, Z



UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: TARJETA CONTROLADORA	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		MÓDULO 1 (Detalle 4 de 4)	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO:	A4
DISEÑADOR: Jesús Rodríguez	ANEXO 9	MÁQUINA FRESADORA CNC	ANEXO 9
NOTA	ESCALA: 1:10	PESO: NO determinado	HUJA 14 DE 31



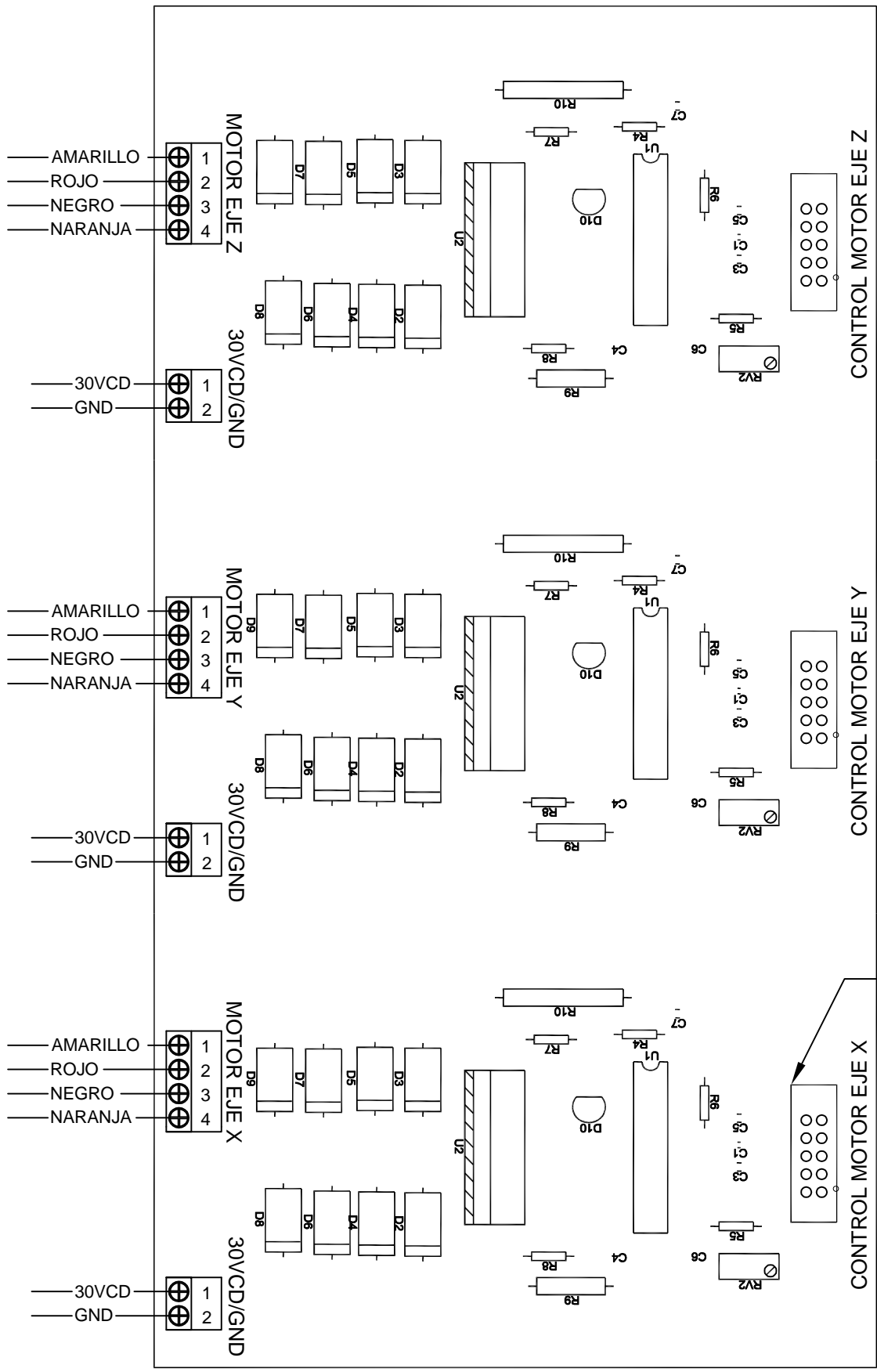
Jack Alimentación AUXILIAR 9-12 VCD "ARDUINO"

Conectoro USB hembra tipo B "ARDUINO"

ARDUINO R3 UNO
 INSTALADO CON
 TERMINALES DE
 CONEXIÓN

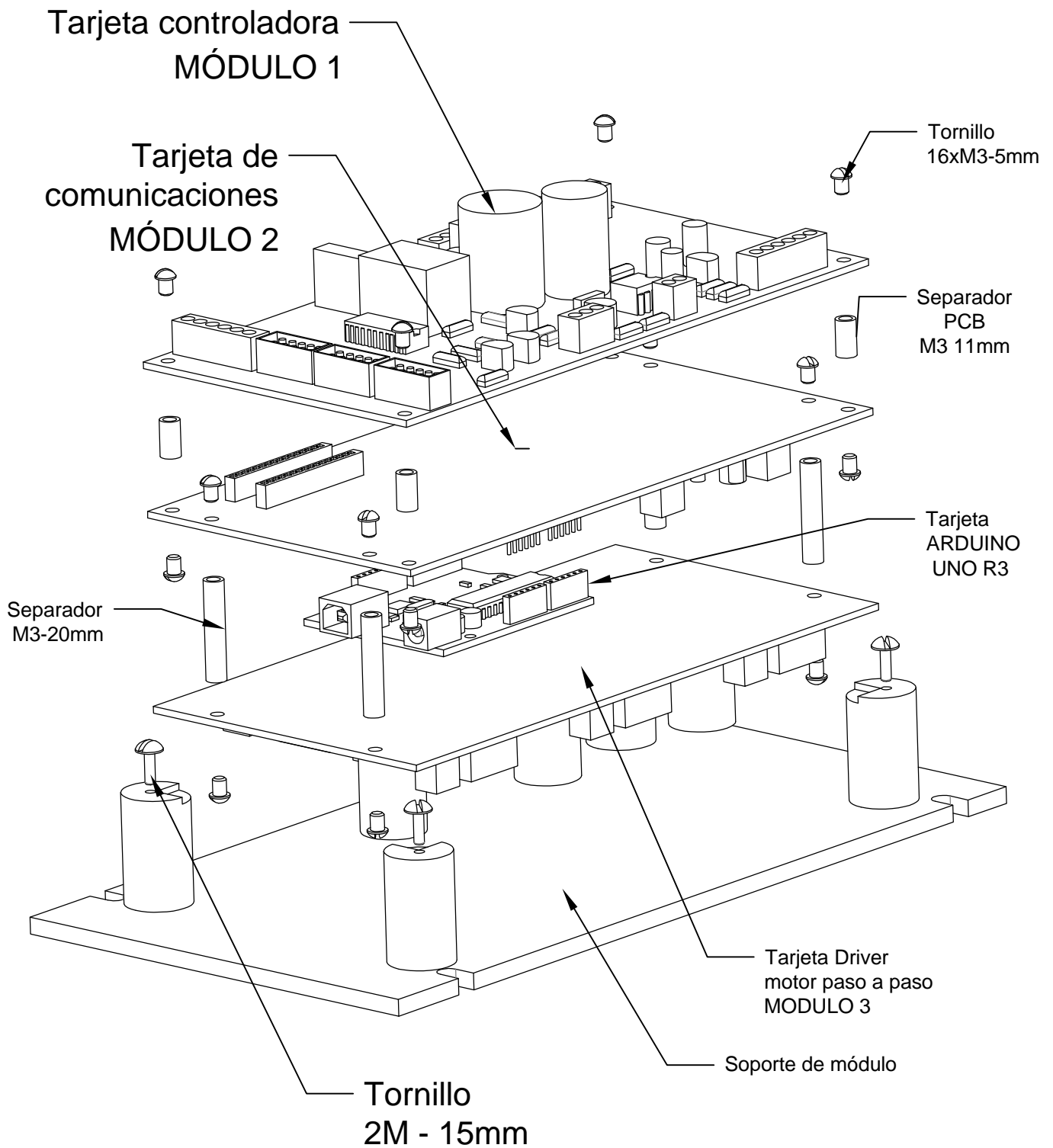
NOTA:
 EL ARDUINO SE
 INSTALA CON LOS
 COMPONENTES
 HACIA ABAJO

UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: TARJETA de COMUNICACIÓN MÓDULO 2	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN			
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015		
DISEÑADOR: Jesús Rodríguez	ANEXO 9	PROYECTO: MÁQUINA FRESADORA CNC	A4
NOTA	ESCALA: 1:10	PESO: NO determinado	ANEXO 9
			HUJA 15 DE 31



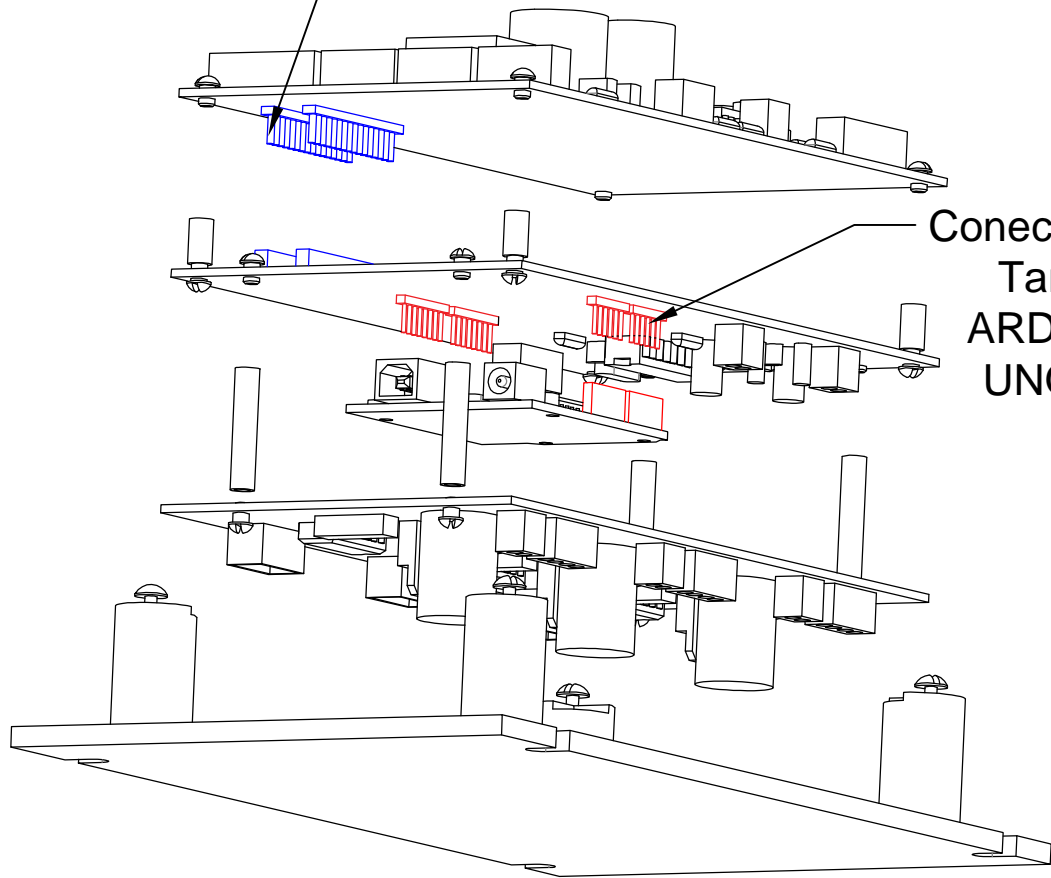
CONECTOR IDC DE 10 PINES PARA MEJOR INFORMACIÓN VE TARJETA CONTROLADORA DISEÑADA (Sección 4 de 4)

UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: TARJETA DRIVER MOTORES MÓDULO 3	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN			
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO: MÁQUINA FRESADORA CNC	
DISEÑADOR: Jesús Rodríguez	ANEXO 9	A4	
NOTA	ESCALA: 1:10	PESO: NO determinado	ANEXO 9
			HOJA 16 DE 31



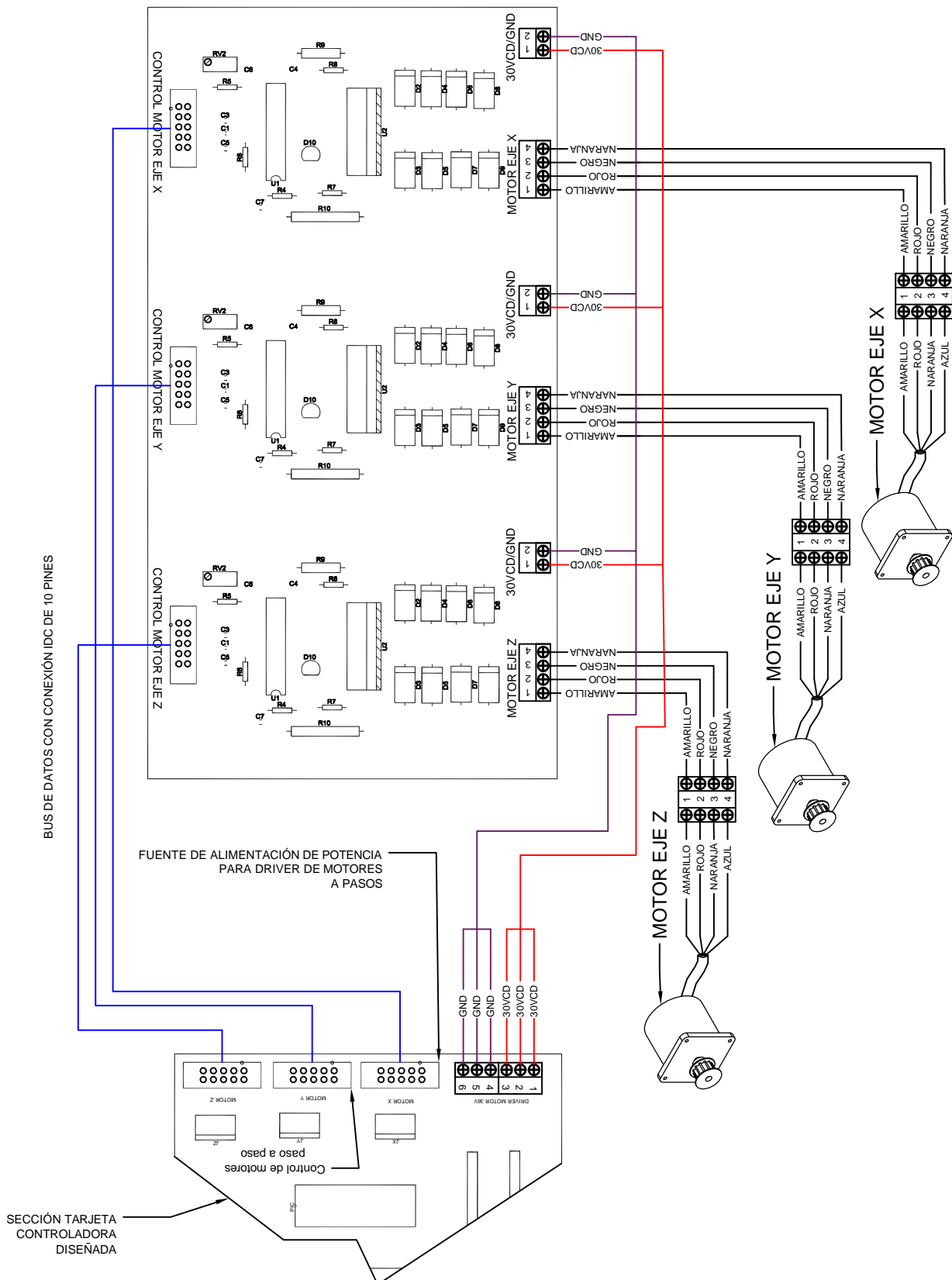
UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: ENSAMBLE MÓDULO CONTROLADOR (Detalle 1 de 2)	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		PROYECTO: MÁQUINA FRESADORA CNC	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	A4	
DISEÑADOR: Jesús Rodríguez	ANEXO 9		
NOTA	ESCALA: 1:10	PESO: NO determinado	HOJA 17 DE 31
		ANEXO 9	

Conector para tarjeta controladora y comunicaciones



Conector para Tarjeta ARDUINO UNO R3

UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: ENSAMBLE MÓDULO CONTROLADOR (Detalle 2 de 2)	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		PROYECTO: MÁQUINA FRESADORA CNC	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015		
DISEÑADOR: Jesús Rodríguez	ANEXO 9		A4
NOTA	ESCALA: 1:10	PESO: NO determinado	HOJA 18 DE 31
		ANEXO 9	



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

DISEÑADOR: Jesús Rodríguez

FECHA: 09/03/2015

ANEXO 9

TÍTULO:

CONEXIÓN MODULO 1, 2 Y 3
(Detalle 1 de 2)

PROYECTO:

MÁQUINA FRESADORA CNC

A4

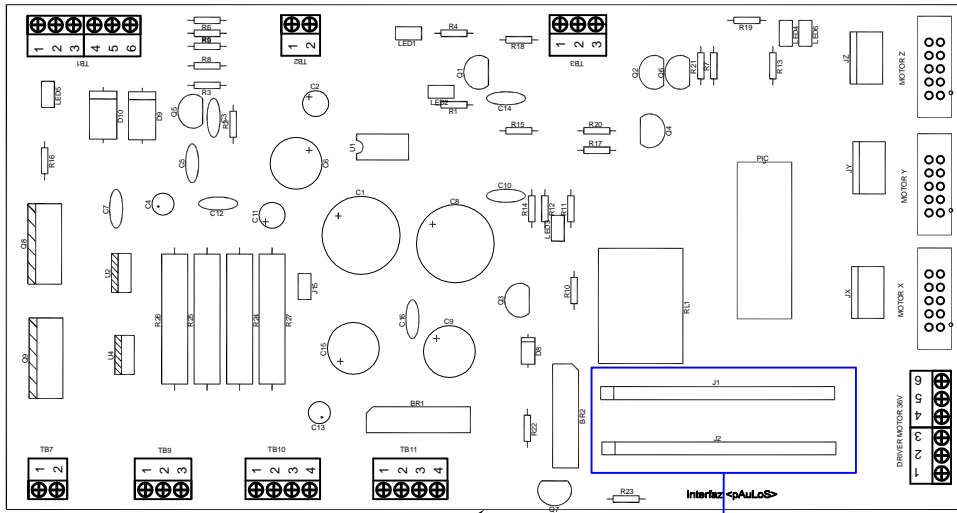
ANEXO 9

NOTA

ESCALA: 1:10

PESO: NO determinado

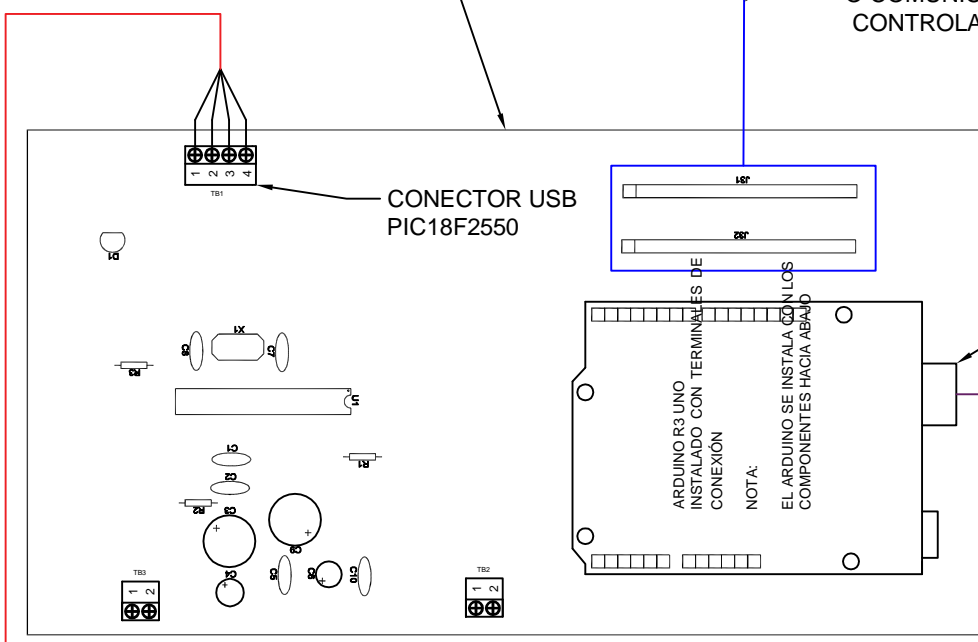
HUJA 19 DE 31



TARJETA CONTROLADORA

TARJETA DE COMUNICACIÓN

CONECTOR PARA BUS DE DATOS O COMUNICACIÓN ENTRE TARJETAS CONTROLADORA Y COMUNICACIONES

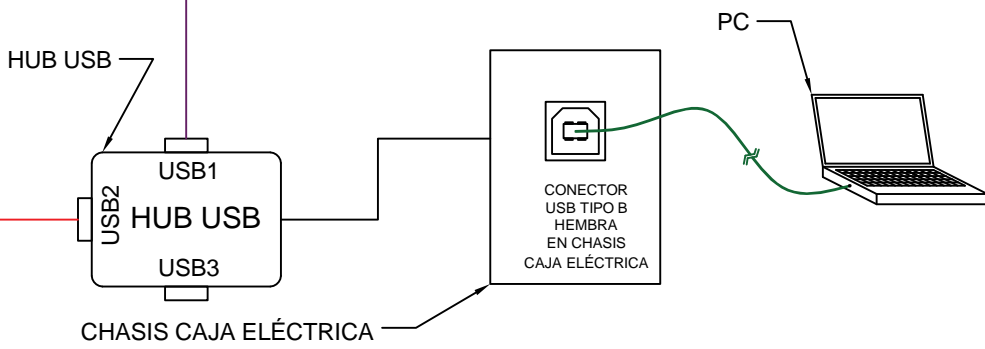


CONECTOR USB PIC18F2550

CONECTOR USB ARDUINO

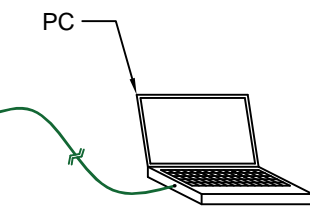
CABLE USB PIC18F2550 TIPO-A A BORNERA

CABLE USB ARDUINO TIPO-A Y B MACHO



HUB USB

CHASIS CAJA ELÉCTRICA

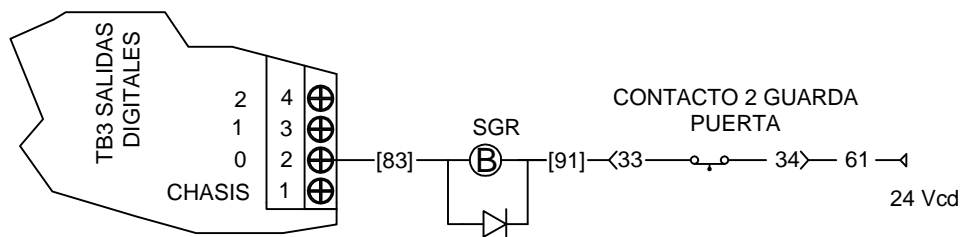
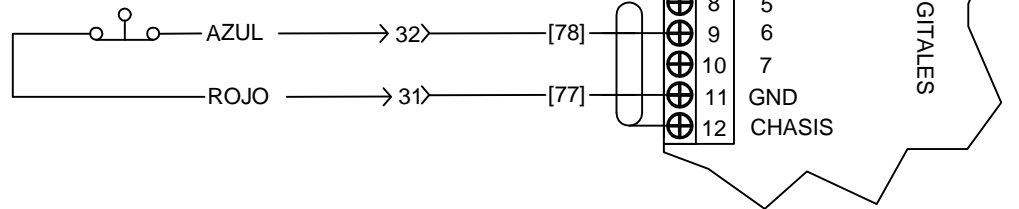


PC

CONECTOR USB TIPO B HEMBRA EN CHASIS CAJA ELÉCTRICA

UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: CONEXIÓN MODULO 1, 2 Y 3 (Detalle 2 de 2)	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN			
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO:	A4
DISEÑADOR: Jesús Rodríguez	ANEXO 9	MÁQUINA FRESADORA CNC	ANEXO 9
NOTA	ESCALA: 1:10	PESO: NO determinado	HUJA 20 DE 31

GUARDA CONTACTO 1



LEYENDA DE DISEÑO

SIMBOLO	DESCRIPCIÓN
	Conector de zócalo de 42 pines
	Cable de conexión
	Conexión a tornillo contactor
	Pulsador de emergencia
	Solenoides
	Diodo de protección
	Contacto normalmente cerrado
	Contacto normalmente abierto
	Bobina contactor
	Final de carrera
	Conexión a tierra
	Cable blindado a tierra
	Fusible

UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

FECHA: 09/03/2015

DISEÑADOR: Jesús Rodríguez

ANEXO 10

TÍTULO:

CIRCUITO DE PROTECCIÓN
PUERTA

PROYECTO:

MÁQUINA FRESADORA CNC

A4

ANEXO 10

NOTA

ESCALA: 1:10

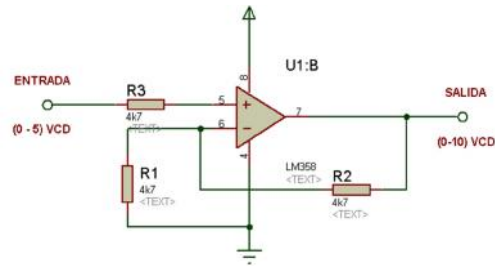
PESO: NO determinado

HUJA 21 DE 31

ANEXO 11

DISEÑO DEL CONTROL DE VELOCIDAD PARA EL USILLO

Circuito de diseño con amplificador operacional, Lm358 con factor de amplificación de 2, la configuración es de amplificador no inversor. El siguiente circuito esquemático indica la descripción de componentes utilizados, como circuito integrado y resistencias.

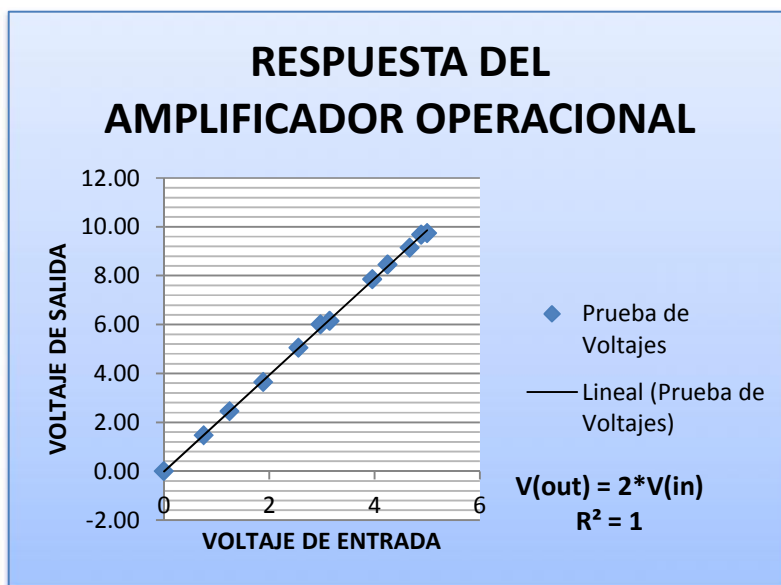


La ecuación que gobierna este diseño de circuito electrónico, ésta dada por la siguiente ecuación, de similar forma determina valores para los diferentes componentes.

$$R1 = 4.7k\Omega \quad R2 = 4.7k\Omega \quad V_{out} = V_{in} \left(1 + \frac{R2}{R1} \right) \quad out = V_{in} \left(1 + \frac{4.7k\Omega}{4.7k\Omega} \right)$$

$$out = V_{in}(2)$$

Los siguientes datos tabulados son pruebas en el circuito real obteniendo datos de voltaje en la salida del amplificador operacional, en respuesta voltaje variable proveniente del potenciómetro de la máquina CNC.

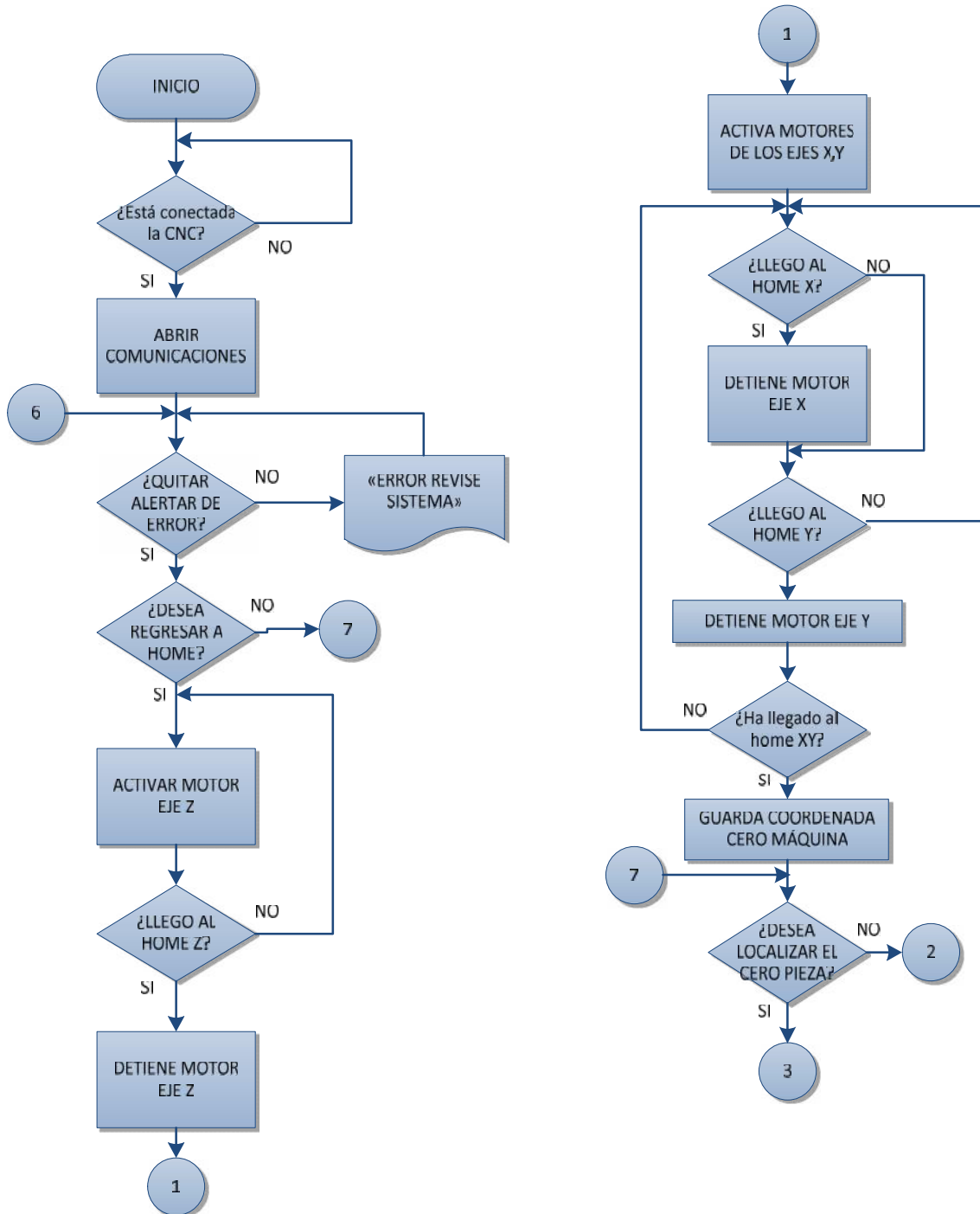


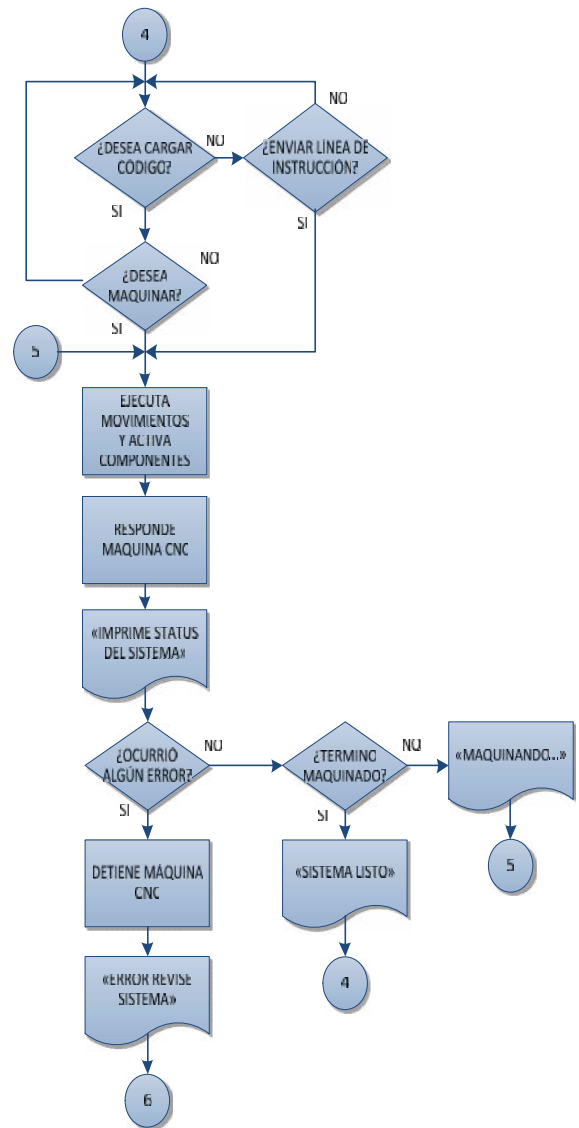
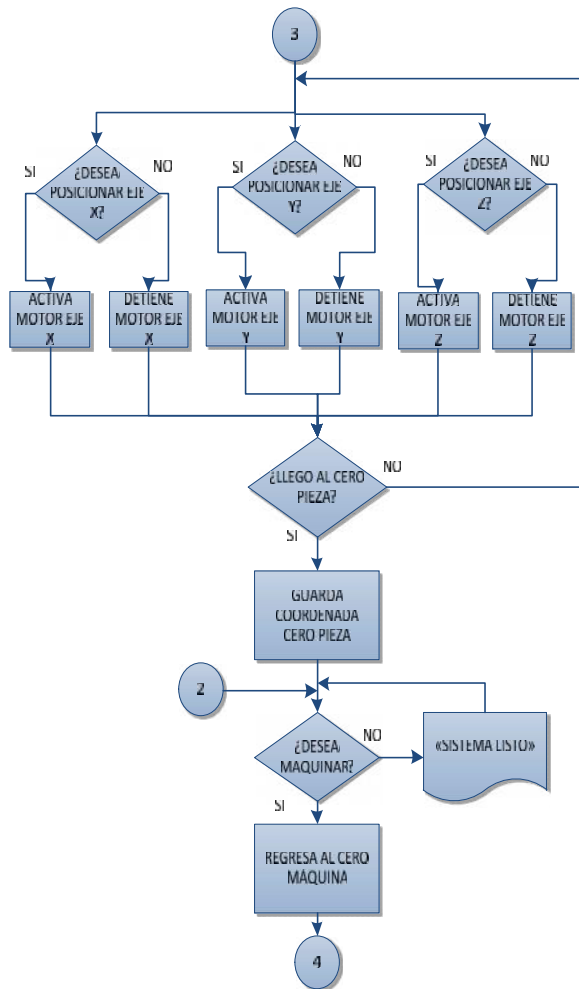
Prueba de Voltajes	
ENTRADA (V)	SALIDA (V)
0	0.00
0.76	1.47
1.25	2.45
1.89	3.65
2.56	5.05
2.98	6.01
3.15	6.15
3.96	7.86
4.25	8.46
4.67	9.15
4.89	9.68
5	9.75

ANEXO 12

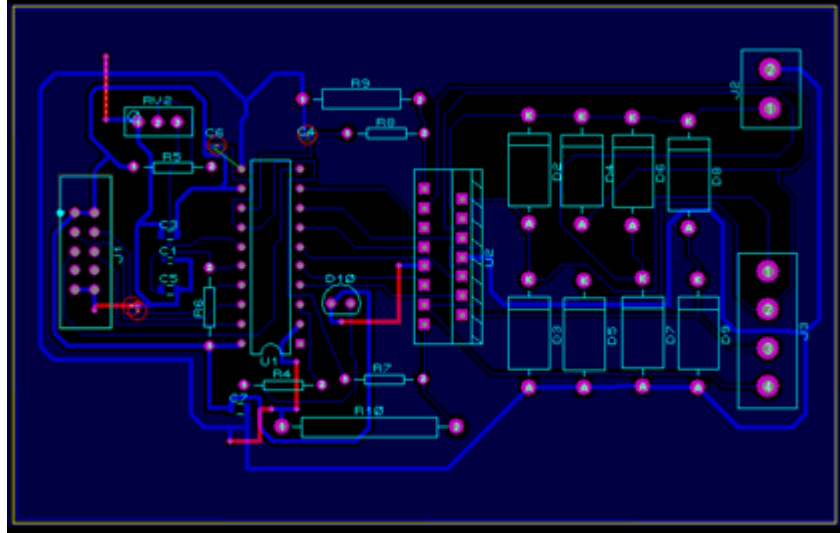
DIAGRAMA DE FLUJO DE CONTROL DE MÁQUINA CNC

El siguiente diagrama de flujo describe el funcionamiento y control de la máquina CNC.

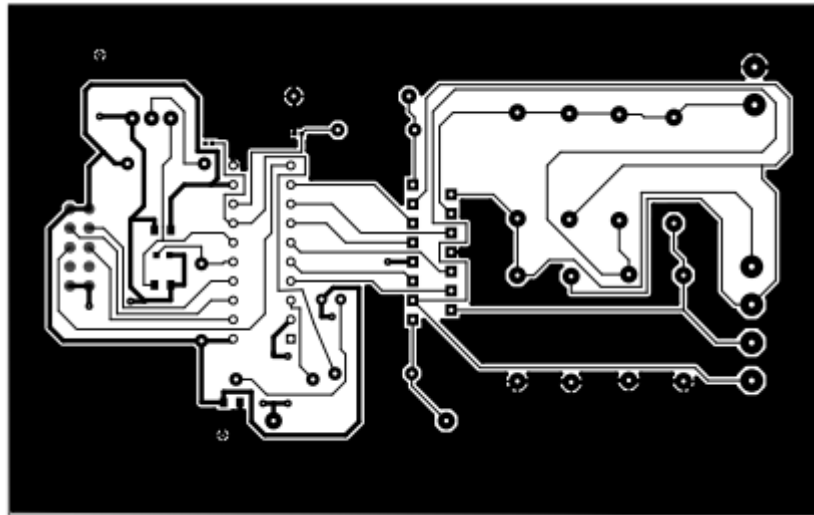




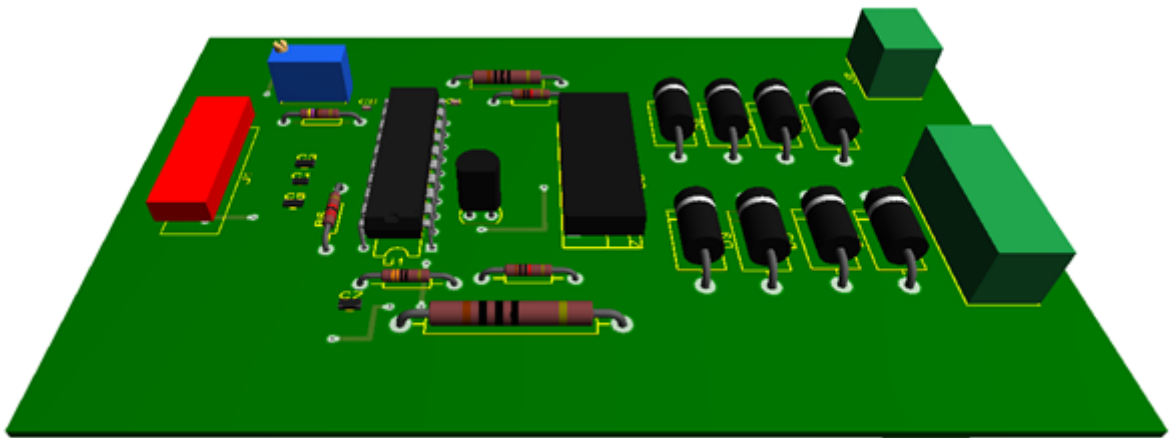
ÖÙÒ÷ ÛÁÛÓÁÖÜØÙÁ ÚVUÛÚÇÈUÁÇÁÇÈU



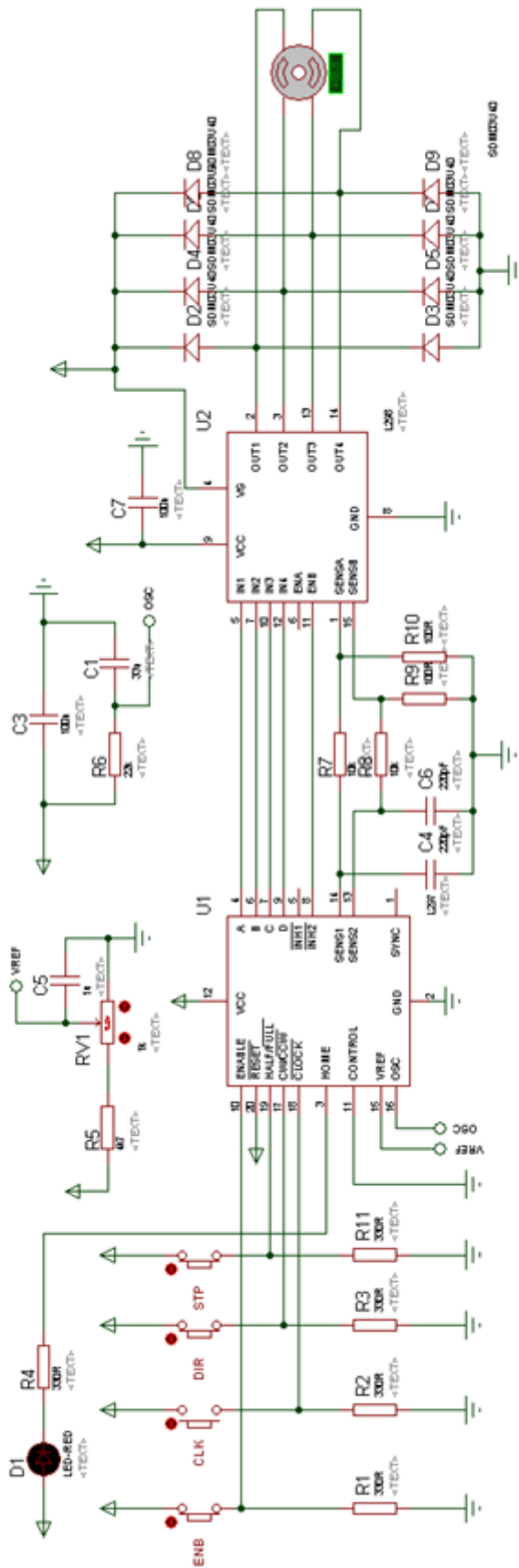
ÚÙVÇÈ



ÚÛÙÒÙÒßVÇÓÇßÁÖ



UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO:	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		DISEÑO PCB DRIVER MOTORES DE LOS EJES X, Y, Z	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO:	
DISEÑADOR: Jesús Rodríguez	ANEXO 13	MÁQUINA FRESADORA CNC	A4
NOTA	ESCALA: 1:10	PESO: NO determinado	ANEXO 13
			HUJA 22 DE 31



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

DISEÑADOR: Jesús Rodríguez

FECHA: 09/03/2015

ANEXO 14

TÍTULO:

CIRCUITO ESQUEMÁTICO
DRIVER MOTORES A PASOS

PROYECTO:

MÁQUINA FRESADORA CNC

A4

ANEXO 14

NOTA

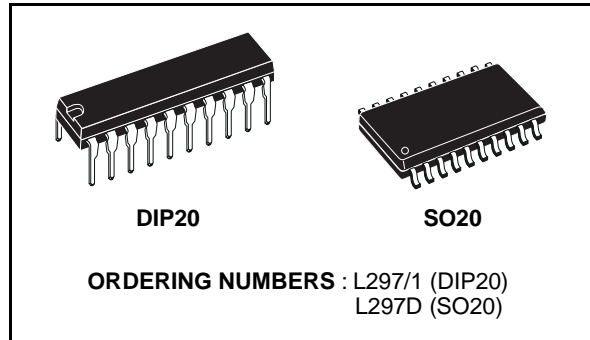
ESCALA: 1:10

PESO: NO determinado

HOJA 23 DE 31

STEPPER MOTOR CONTROLLERS

- NORMAL/WAVE DRIVE
- HALF/FULL STEP MODES
- CLOCKWISE/ANTICLOCKWISE DIRECTION
- SWITCHMODE LOAD CURRENT REGULATION
- PROGRAMMABLE LOAD CURRENT
- FEW EXTERNAL COMPONENTS
- RESET INPUT & HOME OUTPUT
- ENABLE INPUT



DESCRIPTION

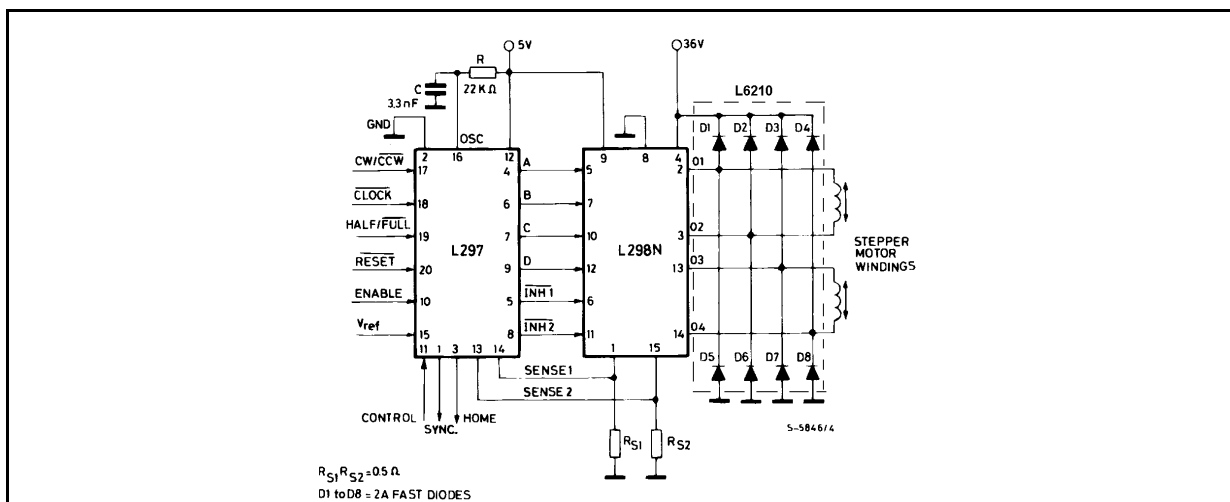
The L297 Stepper Motor Controller IC generates four phase drive signals for two phase bipolar and four phase unipolar step motors in microcomputer-controlled applications. The motor can be driven in half step, normal and wave drive modes and on-chip PWM chopper circuits permit switch-mode control of the current in the windings. A feature of

this device is that it requires only clock, direction and mode input signals. Since the phase are generated internally the burden on the microprocessor, and the programmer, is greatly reduced. Mounted in DIP20 and SO20 packages, the L297 can be used with monolithic bridge drives such as the L298N or L293E, or with discrete transistors and darlingtonts.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_s	Supply voltage	10	V
V_i	Input signals	7	V
P_{tot}	Total power dissipation ($T_{amb} = 70^\circ\text{C}$)	1	W
T_{stg}, T_j	Storage and junction temperature	-40 to + 150	$^\circ\text{C}$

TWO PHASE BIPOLAR STEPPER MOTOR CONTROL CIRCUIT



PIN FUNCTIONS - L297/1 - L297D

N°	NAME	FUNCTION
1	SYNC	Output of the on-chip chopper oscillator. The SYNC connections of all L297s to be synchronized are connected together and the oscillator components are omitted on all but one. If an external clock source is used it is injected at this terminal.
2	GND	Ground connection.
3	HOME	Open collector output that indicates when the L297 is in its initial state (ABCD = 0101). The transistor is open when this signal is active.
4	A	Motor phase A drive signal for power stage.
5	$\overline{\text{INH1}}$	Active low inhibit control for driver stage of A and B phases. When a bipolar bridge is used this signal can be used to ensure fast decay of load current when a winding is de-energized. Also used by chopper to regulate load current if CONTROL input is low.
6	B	Motor phase B drive signal for power stage.
7	C	Motor phase C drive signal for power stage.
8	$\overline{\text{INH2}}$	Active low inhibit control for drive stages of C and D phases. Same functions as INH1.
9	D	Motor phase D drive signal for power stage.
10	ENABLE	Chip enable input. When low (inactive) INH1, INH2, A, B, C and D are brought low.
11	CONTROL	Control input that defines action of chopper. When low chopper acts on INH1 and INH2; when high chopper acts on phase lines ABCD.
12	V_s	5V supply input.
13	SENS ₂	Input for load current sense voltage from power stages of phases C and D.
14	SENS ₁	Input for load current sense voltage from power stages of phases A and B.
15	V_{ref}	Reference voltage for chopper circuit. A voltage applied to this pin determines the peak load current.
16	OSC	An RC network (R to V_{CC} , C to ground) connected to this terminal determines the chopper rate. This terminal is connected to ground on all but one device in synchronized multi - L297 configurations. $f \cong 1/0.69 RC$
17	$\text{CW}/\overline{\text{CCW}}$	Clockwise/counterclockwise direction control input. Physical direction of motor rotation also depends on connection of windings. Synchronized internally therefore direction can be changed at any time.
18	$\overline{\text{CLOCK}}$	Step clock. An active low pulse on this input advances the motor one increment. The step occurs on the rising edge of this signal.

PIN FUNCTIONS - L297/1 - L297D (continued)

N°	NAME	FUNCTION
19	HALF/FULL	Half/full step select input. When high selects half step operation, when low selects full step operation. One-phase-on full step mode is obtained by selecting FULL when the L297's translator is at an even-numbered state. Two-phase-on full step mode is set by selecting FULL when the translator is at an odd numbered position. (The home position is designate state 1).
20	RESET	Reset input. An active low pulse on this input restores the translator to the home position (state 1, ABCD = 0101).

THERMAL DATA

Symbol	Parameter	DIP20	SO20	Unit	
R _{th-j-amb}	Thermal resistance junction-ambient	max	80	100	°C/W

CIRCUIT OPERATION

The L297 is intended for use with a dual bridge driver, quad darlington array or discrete power devices in step motor driving applications. It receives step clock, direction and mode signals from the systems controller (usually a microcomputer chip) and generates control signals for the power stage.

The principal functions are a translator, which generates the motor phase sequences, and a dual PWM chopper circuit which regulates the current in the motor windings. The translator generates three different sequences, selected by the HALF/FULL input. These are normal (two phases energised), wave drive (one phase energised) and half-step (alternately one phase energised/two phases energised). Two inhibit signals are also generated by the L297 in half step and wave drive modes. These signals, which connect directly to the L298's enable inputs, are intended to speed current decay when a winding is de-energised. When the L297 is used to drive a unipolar motor the chopper acts on these lines.

An input called CONTROL determines whether the chopper will act on the phase lines ABCD or the inhibit lines INH1 and INH2. When the phase lines

are chopped the non-active phase line of each pair (AB or CD) is activated (rather than interrupting the line then active). In L297 + L298 configurations this technique reduces dissipation in the load current sense resistors.

A common on-chip oscillator drives the dual chopper. It supplies pulses at the chopper rate which set the two flip-flops FF1 and FF2. When the current in a winding reaches the programmed peak value the voltage across the sense resistor (connected to one of the sense inputs SENS₁ or SENS₂) equals V_{ref} and the corresponding comparator resets its flip flop, interrupting the drive current until the next oscillator pulse arrives. The peak current for both windings is programmed by a voltage divider on the V_{ref} input.

Ground noise problems in multiple configurations can be avoided by synchronising the chopper oscillators. This is done by connecting all the SYNC pins together, mounting the oscillator RC network on one device only and grounding the OSC pin on all other devices.

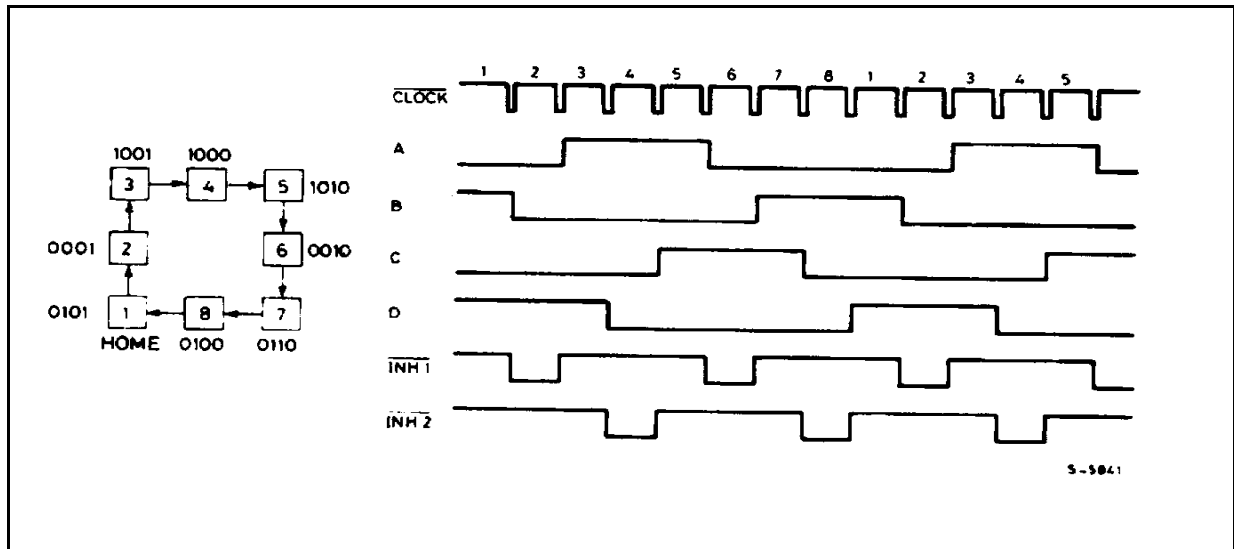
MOTOR DRIVING PHASE SEQUENCES

The L297's translator generates phase sequences for normal drive, wave drive and half step modes. The state sequences and output waveforms for these three modes are shown below. In all cases the translator advances on the low to high transition of $\overline{\text{CLOCK}}$.

Clockwise rotation is indicated; for anticlockwise rotation the sequences are simply reversed. $\overline{\text{RESET}}$ restores the translator to state 1, where ABCD = 0101.

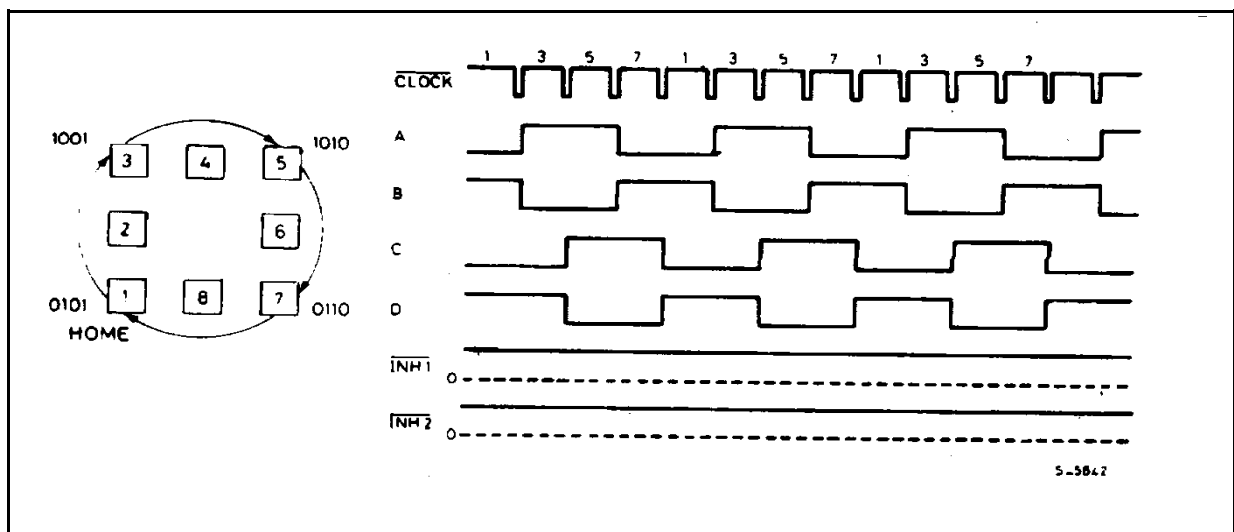
HALF STEP MODE

Half step mode is selected by a high level on the $\overline{\text{HALF/FULL}}$ input.



NORMAL DRIVE MODE

Normal drive mode (also called "two-phase-on" drive) is selected by a low level on the $\overline{\text{HALF/FULL}}$ input when the translator is at an odd numbered state (1, 3, 5 or 7). In this mode the $\overline{\text{INH1}}$ and $\overline{\text{INH2}}$ outputs remain high throughout.



APPLICATION INFORMATION

TWO PHASE BIPOLAR STEPPER MOTOR CONTROL CIRCUIT

This circuit drives bipolar stepper motors with winding currents up to 2A. The diodes are fast 2A types.

Figure 2.

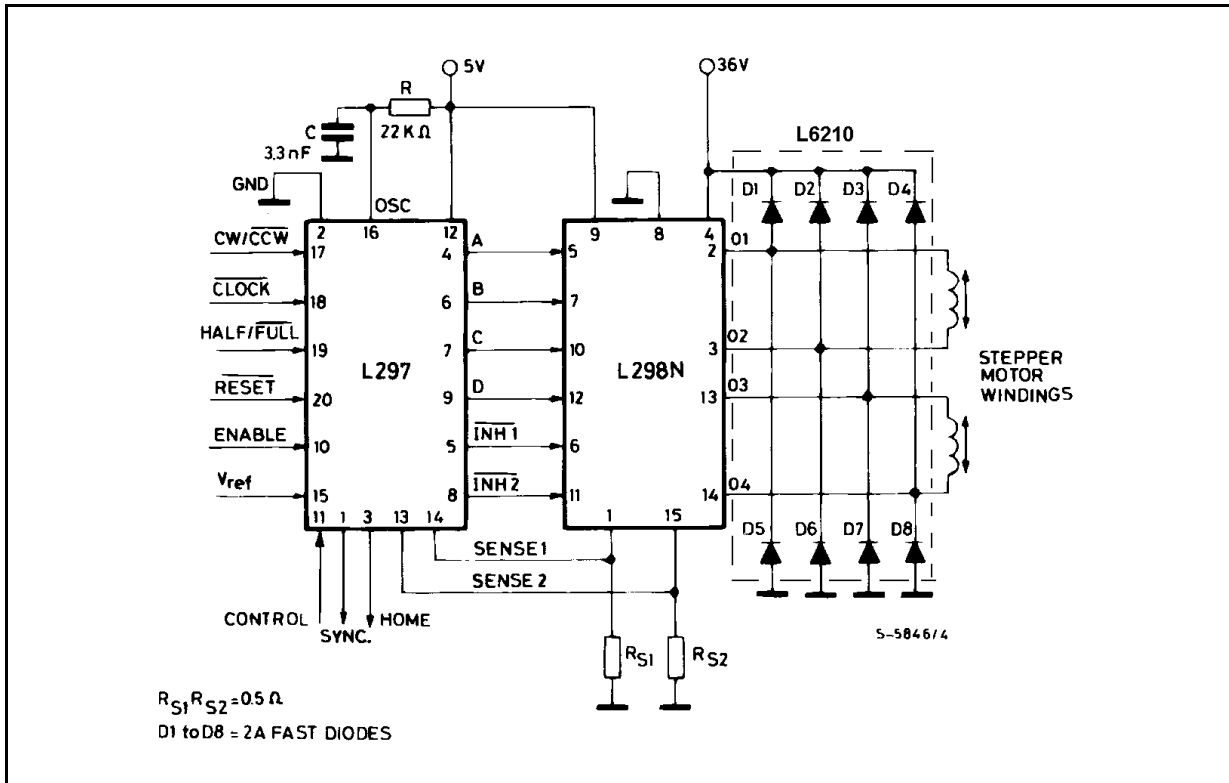
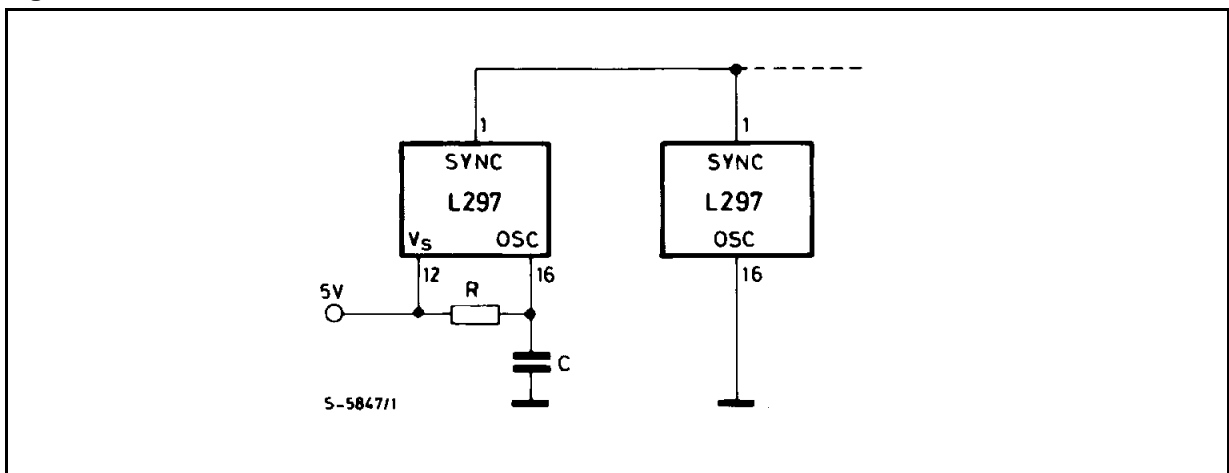


Figure 3 : Synchronising L297s

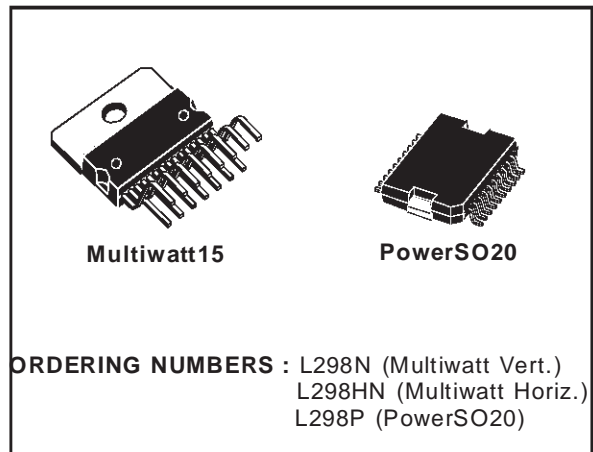


DUAL FULL-BRIDGE DRIVER

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)

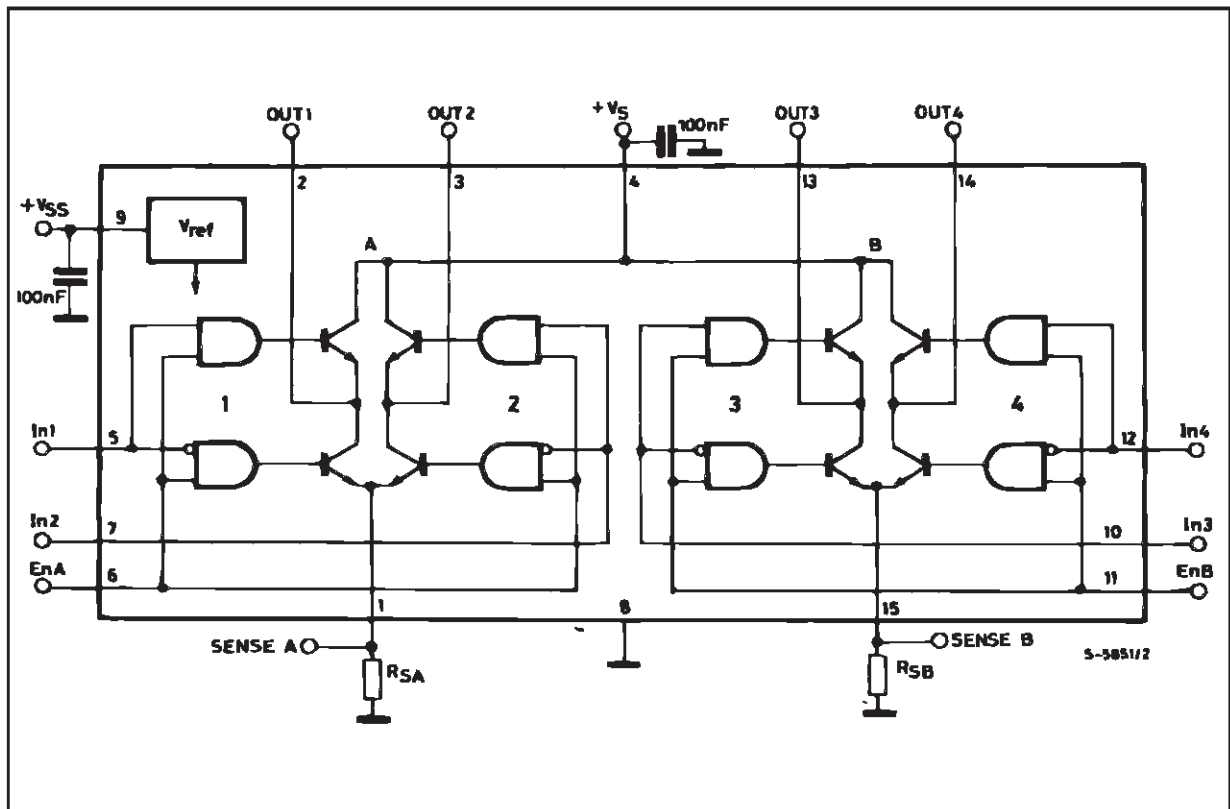
DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the con-



nection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

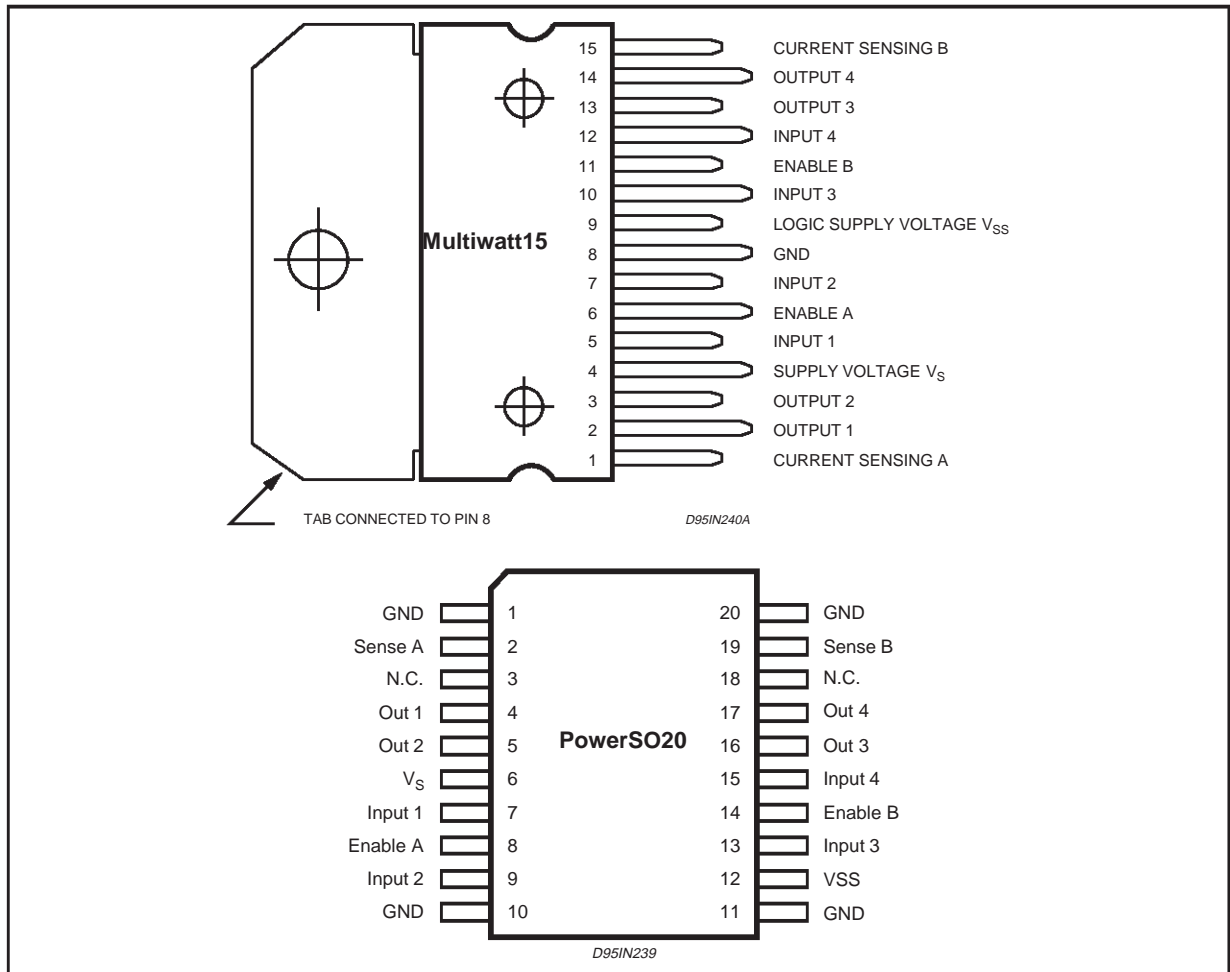
BLOCK DIAGRAM



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_S	Power Supply	50	V
V_{SS}	Logic Supply Voltage	7	V
V_I, V_{en}	Input and Enable Voltage	-0.3 to 7	V
I_O	Peak Output Current (each Channel)		
	- Non Repetitive ($t = 100\mu s$)	3	A
	- Repetitive (80% on -20% off; $t_{on} = 10ms$)	2.5	A
	-DC Operation	2	A
V_{sens}	Sensing Voltage	-1 to 2.3	V
P_{tot}	Total Power Dissipation ($T_{case} = 75^\circ C$)	25	W
T_{op}	Junction Operating Temperature	-25 to 130	$^\circ C$
T_{stg}, T_j	Storage and Junction Temperature	-40 to 150	$^\circ C$

PIN CONNECTIONS (top view)



THERMAL DATA

Symbol	Parameter		PowerSO20	Multiwatt15	Unit
$R_{th\ j-case}$	Thermal Resistance Junction-case	Max.	-	3	$^\circ C/W$
$R_{th\ j-amb}$	Thermal Resistance Junction-ambient	Max.	13 (*)	35	$^\circ C/W$

(*) Mounted on aluminum substrate

PIN FUNCTIONS (refer to the block diagram)

MW.15	PowerSO	Name	Function
1;15	2;19	Sense A; Sense B	Between this pin and ground is connected the sense resistor to control the current of the load.
2;3	4;5	Out 1; Out 2	Outputs of the Bridge A; the current that flows through the load connected between these two pins is monitored at pin 1.
4	6	V _S	Supply Voltage for the Power Output Stages. A non-inductive 100nF capacitor must be connected between this pin and ground.
5;7	7;9	Input 1; Input 2	TTL Compatible Inputs of the Bridge A.
6;11	8;14	Enable A; Enable B	TTL Compatible Enable Input: the L state disables the bridge A (enable A) and/or the bridge B (enable B).
8	1,10,11,20	GND	Ground.
9	12	V _{SS}	Supply Voltage for the Logic Blocks. A100nF capacitor must be connected between this pin and ground.
10; 12	13;15	Input 3; Input 4	TTL Compatible Inputs of the Bridge B.
13; 14	16;17	Out 3; Out 4	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15.
–	3;18	N.C.	Not Connected

ELECTRICAL CHARACTERISTICS (V_S = 42V; V_{SS} = 5V, T_j = 25°C; unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V _S	Supply Voltage (pin 4)	Operative Condition	V _{IH} +2.5		46	V
V _{SS}	Logic Supply Voltage (pin 9)		4.5	5	7	V
I _S	Quiescent Supply Current (pin 4)	V _{en} = H; I _L = 0 V _i = L V _i = H		13 50	22 70	mA mA
		V _{en} = L V _i = X			4	mA
I _{SS}	Quiescent Current from V _{SS} (pin 9)	V _{en} = H; I _L = 0 V _i = L V _i = H		24 7	36 12	mA mA
		V _{en} = L V _i = X			6	mA
V _{iL}	Input Low Voltage (pins 5, 7, 10, 12)		–0.3		1.5	V
V _{iH}	Input High Voltage (pins 5, 7, 10, 12)		2.3		V _{SS}	V
I _{iL}	Low Voltage Input Current (pins 5, 7, 10, 12)	V _i = L			–10	μA
I _{iH}	High Voltage Input Current (pins 5, 7, 10, 12)	V _i = H ≤ V _{SS} –0.6V		30	100	μA
V _{en} = L	Enable Low Voltage (pins 6, 11)		–0.3		1.5	V
V _{en} = H	Enable High Voltage (pins 6, 11)		2.3		V _{SS}	V
I _{en} = L	Low Voltage Enable Current (pins 6, 11)	V _{en} = L			–10	μA
I _{en} = H	High Voltage Enable Current (pins 6, 11)	V _{en} = H ≤ V _{SS} –0.6V		30	100	μA
V _{CEsat(H)}	Source Saturation Voltage	I _L = 1A I _L = 2A	0.95	1.35 2	1.7 2.7	V V
V _{CEsat(L)}	Sink Saturation Voltage	I _L = 1A (5) I _L = 2A (5)	0.85	1.2 1.7	1.6 2.3	V V
V _{CEsat}	Total Drop	I _L = 1A (5) I _L = 2A (5)	1.80		3.2 4.9	V V
V _{sens}	Sensing Voltage (pins 1, 15)		–1 (1)		2	V

This solution can drive until 3 Amps In DC operation and until 3.5 Amps of a repetitive peak current.

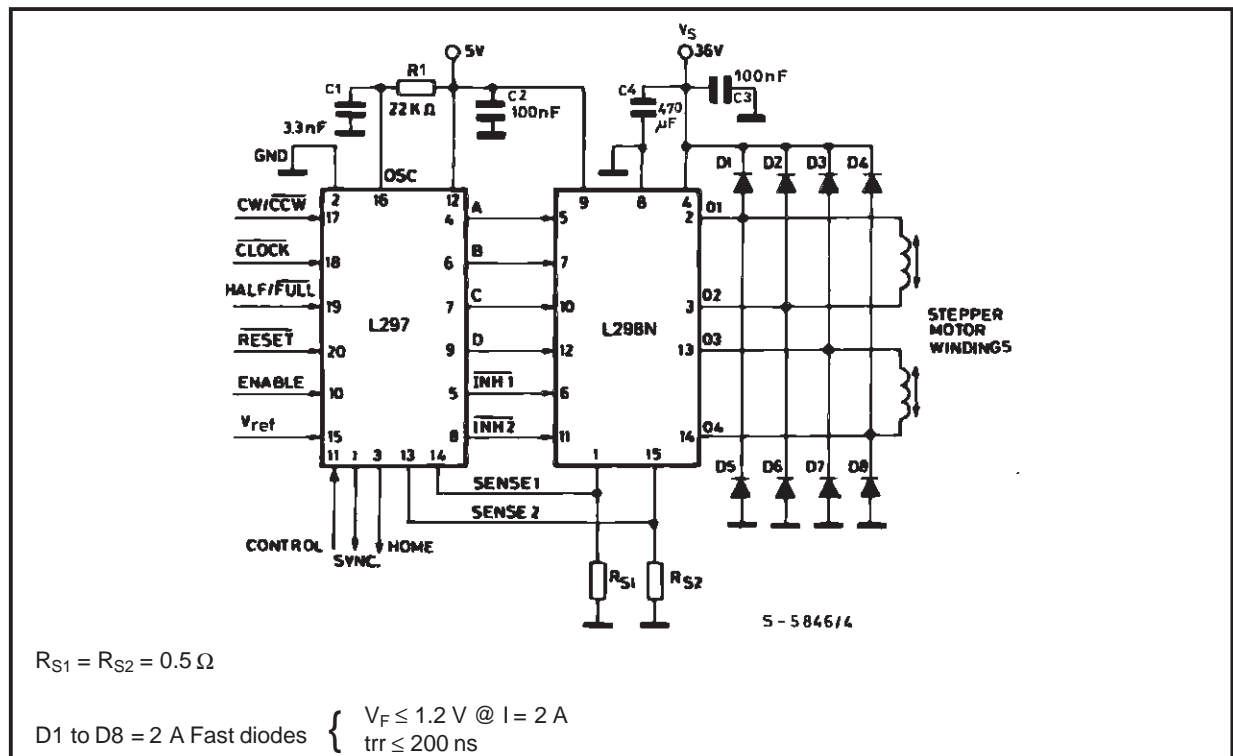
On Fig 8 it is shown the driving of a two phase bipolar stepper motor ; the needed signals to drive the inputs of the L298 are generated, in this example, from the IC L297.

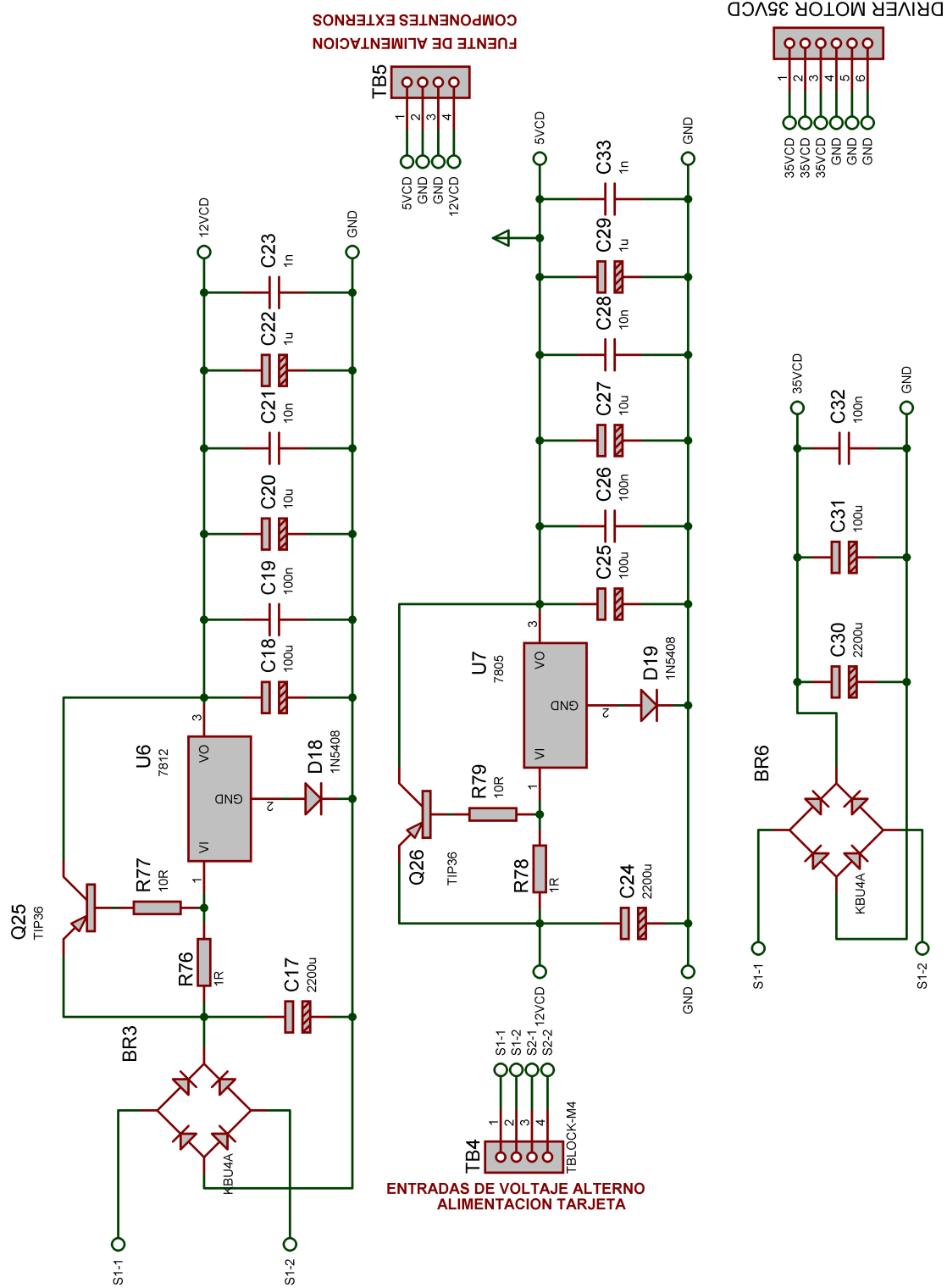
Fig 9 shows an example of P.C.B. designed for the application of Fig 8.

Figure 8 : Two Phase Bipolar Stepper Motor Circuit.

This circuit drives bipolar stepper motors with winding currents up to 2 A. The diodes are fast 2 A types.

Fig 10 shows a second two phase bipolar stepper motor control circuit where the current is controlled by the I.C. L6506.





UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

FECHA: 09/03/2015

DISEÑADOR: Jesús Rodríguez

ANEXO 16

TÍTULO:

CIRCUITO ESQUEMÁTICO
MÓDULO 1 (sección 1 de 3)

PROYECTO:

MÁQUINA FRESADORA CNC

A4

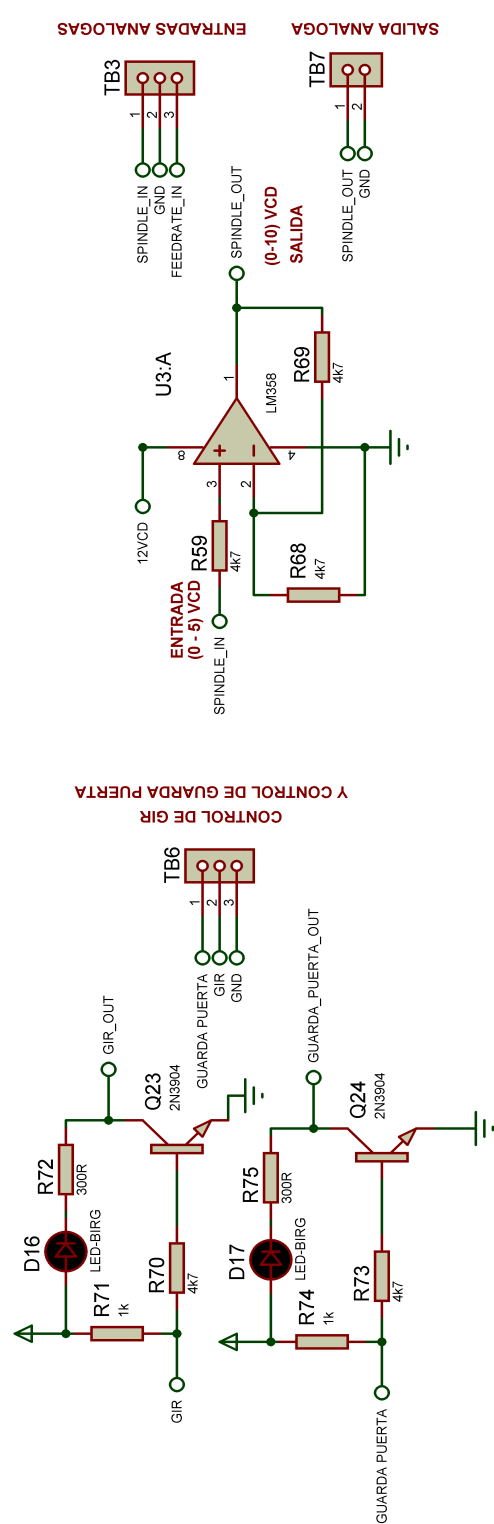
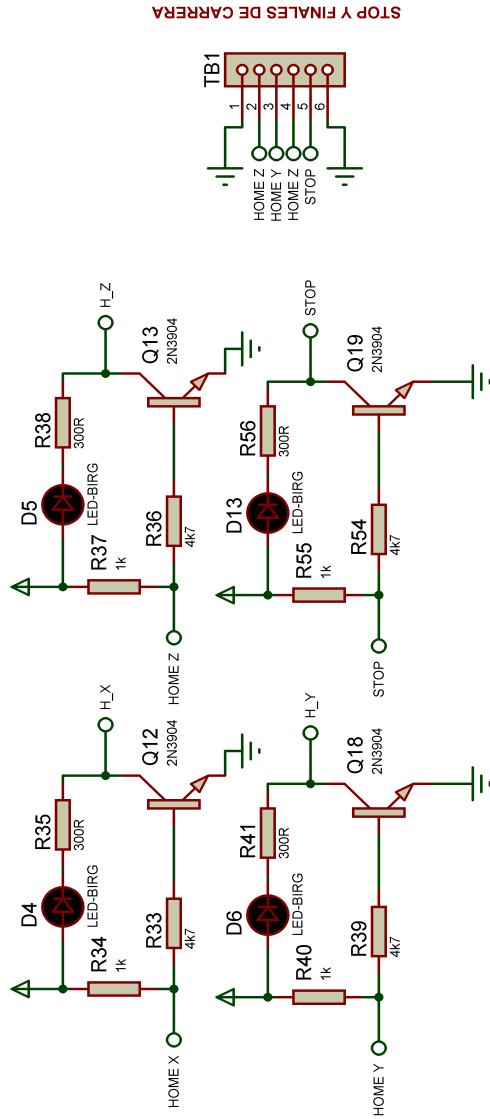
ANEXO 16

NOTA

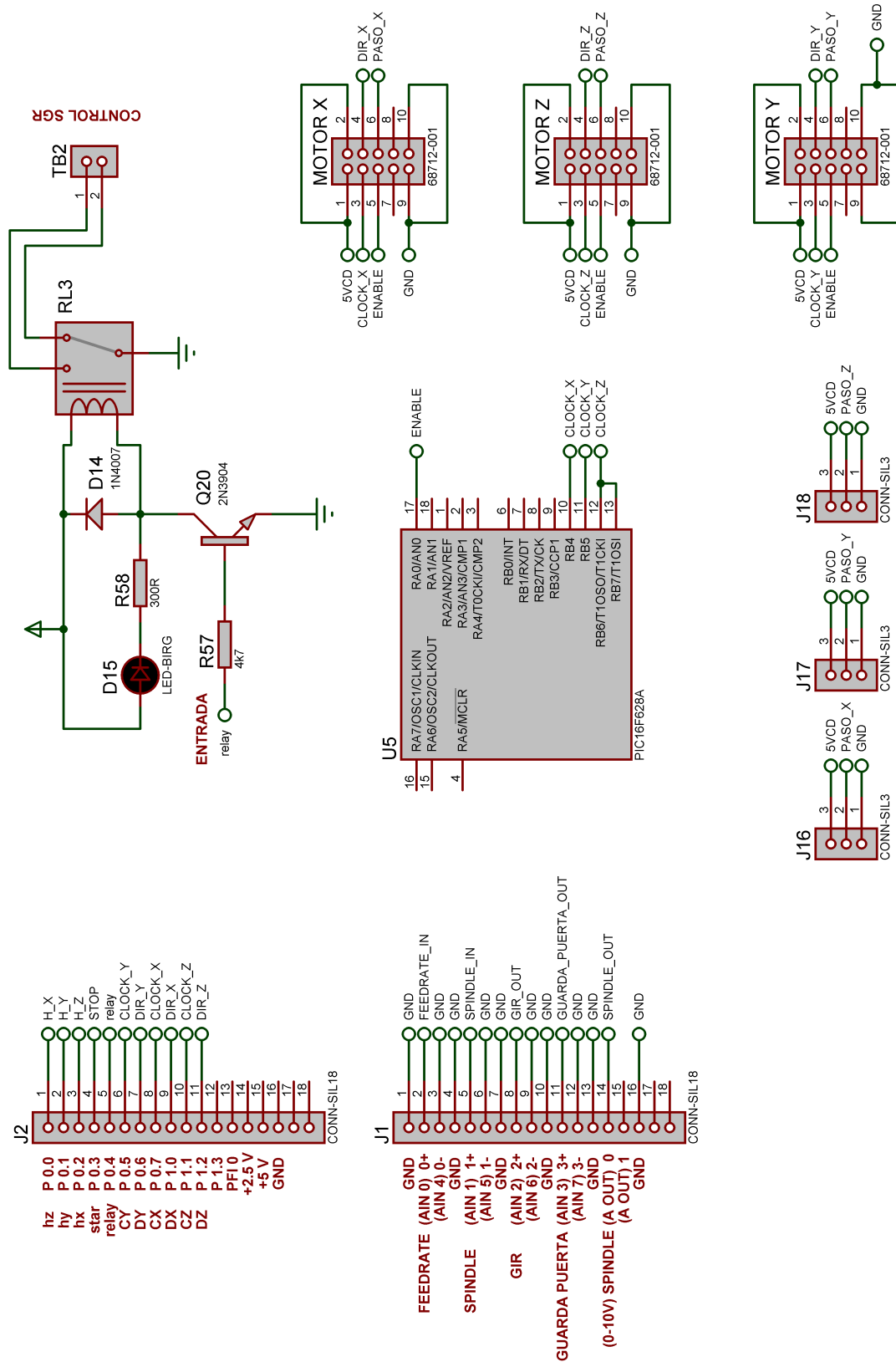
ESCALA: 1:10

PESO: NO determinado

HUJA 24 DE 31



UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: CIRCUITO ESQUEMÁTICO MÓDULO 1 (sección 2 de 3)	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		PROYECTO: MÁQUINA FRESADORA CNC	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	ANEXO 16	A4
DISEÑADOR: Jesús Rodríguez			ANEXO 16
NOTA	ESCALA: 1:10	PESO: NO determinado	HOJA 25 DE 31



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN

REVISIÓN: ING. Wilian López

DISEÑADOR: Jesús Rodríguez

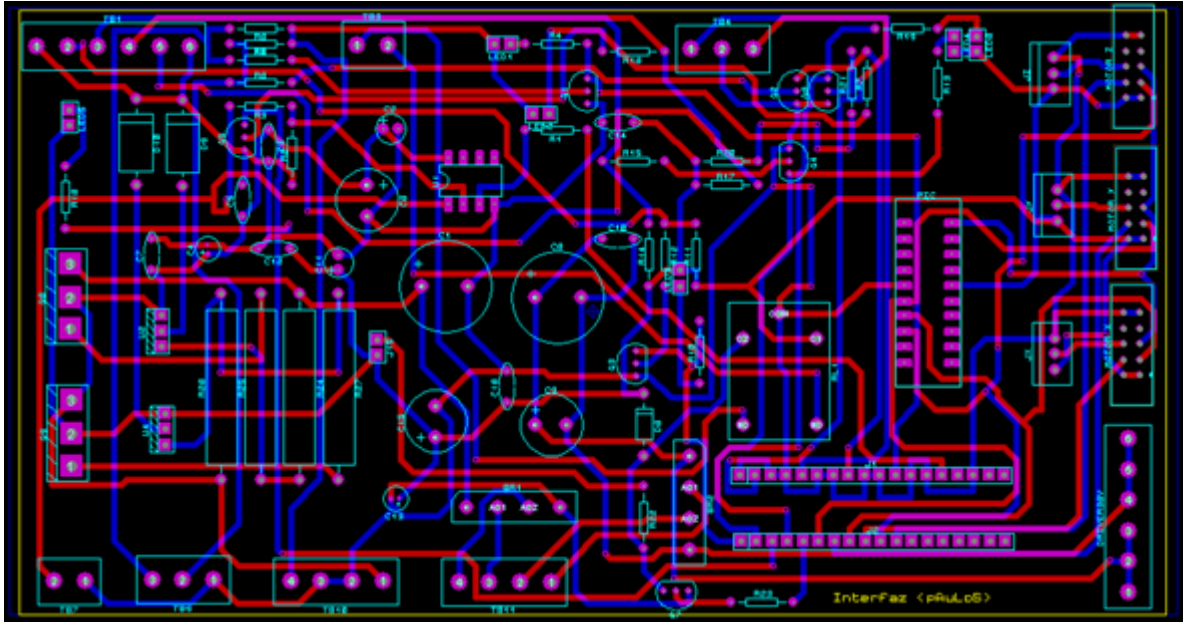
FECHA: 09/03/2015

ANEXO 16

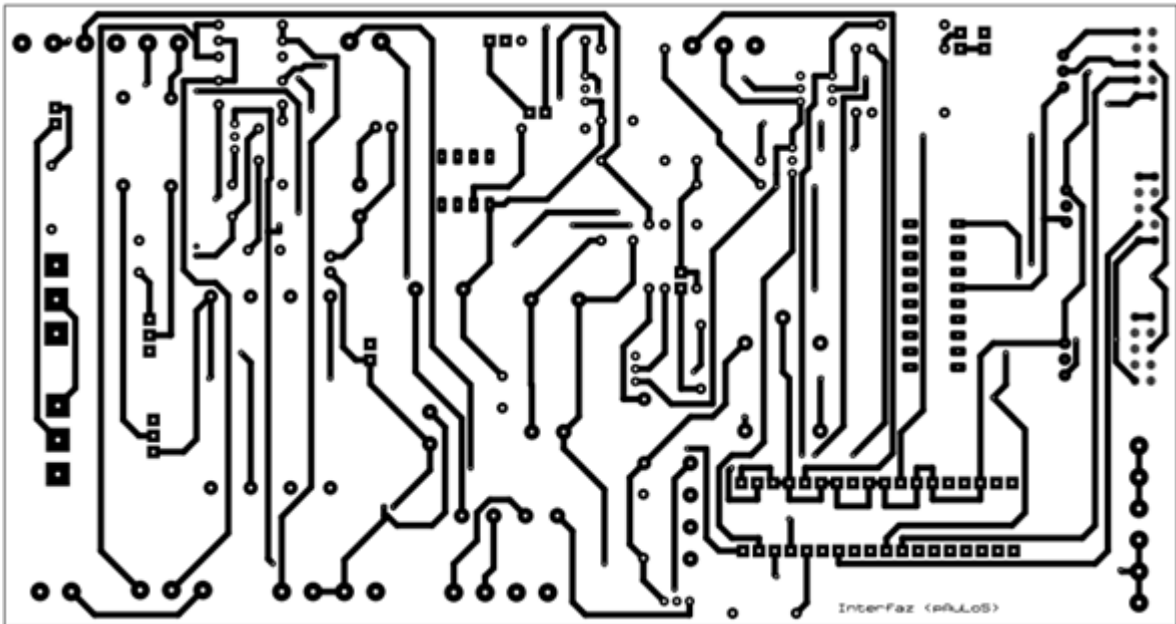
TÍTULO: CIRCUITO ESQUEMÁTICO
MÓDULO 1 (sección 3 de 3)

PROYECTO: MÁQUINA FRESADORA CNC
A4
ANEXO 16

ΤΙΤΛΟΣ: ΠΡΟΤΥΠΟ ΣΧΕΔΙΟ ΚΑΡΤΗΣ 3 ΔΙΑΣΤΑΣΕΩΣ

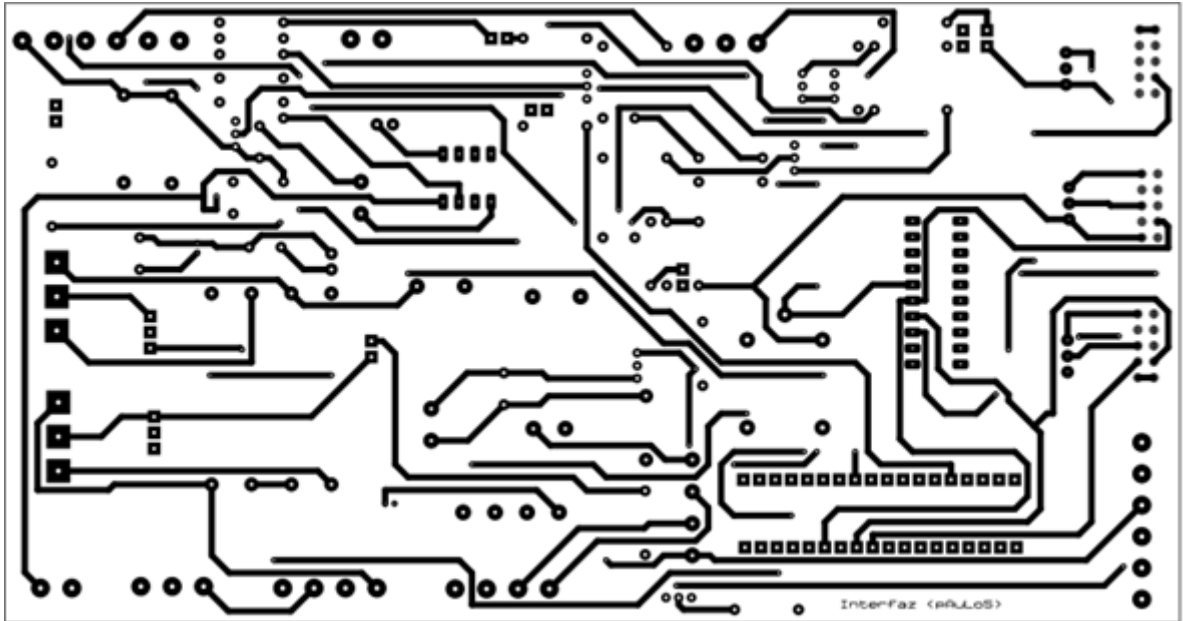


ΠΡΟΤΥΠΟ ΣΧΕΔΙΟ ΚΑΡΤΗΣ

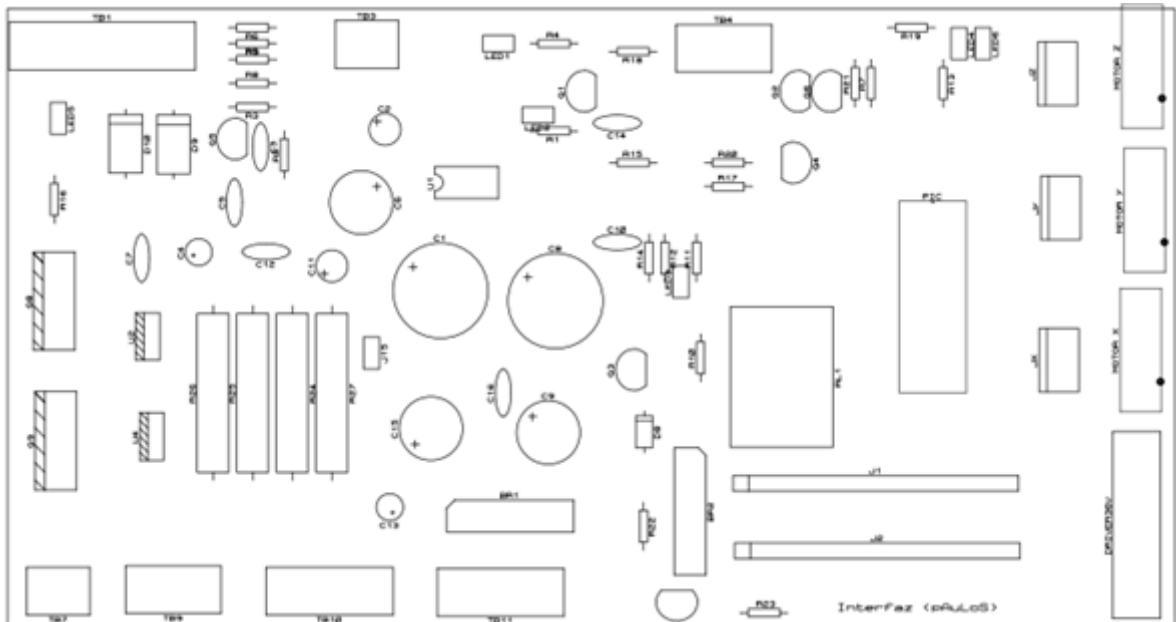


UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO:	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		DISEÑO PCB	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	MÓDULO 1 (sección 1 de 2)	
DISEÑADOR: Jesús Rodríguez	ANEXO 16	PROYECTO:	A4
		MÁQUINA FRESADORA CNC	ANEXO 16
NOTA	ESCALA: 1:10	PESO: NO determinado	HOJA 27 DE 31

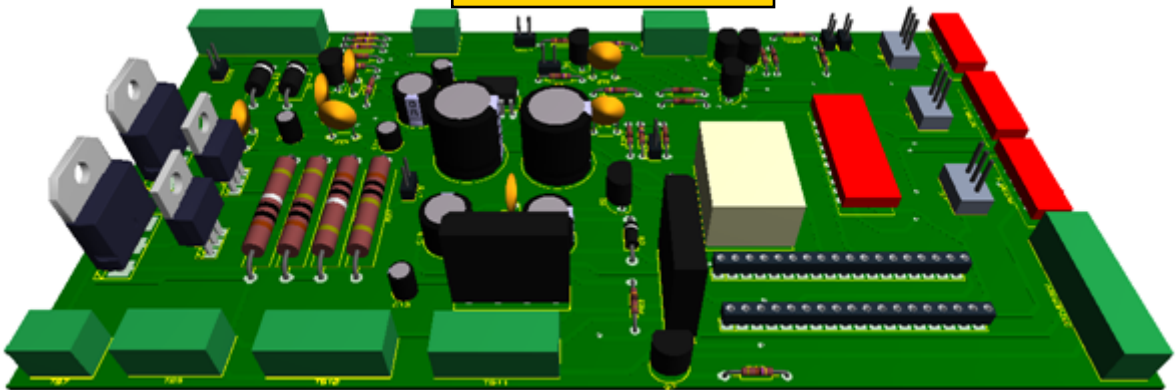
ΥΠΟΝΟΜΙΑ ΕΝΟΛΩΜΟΥ



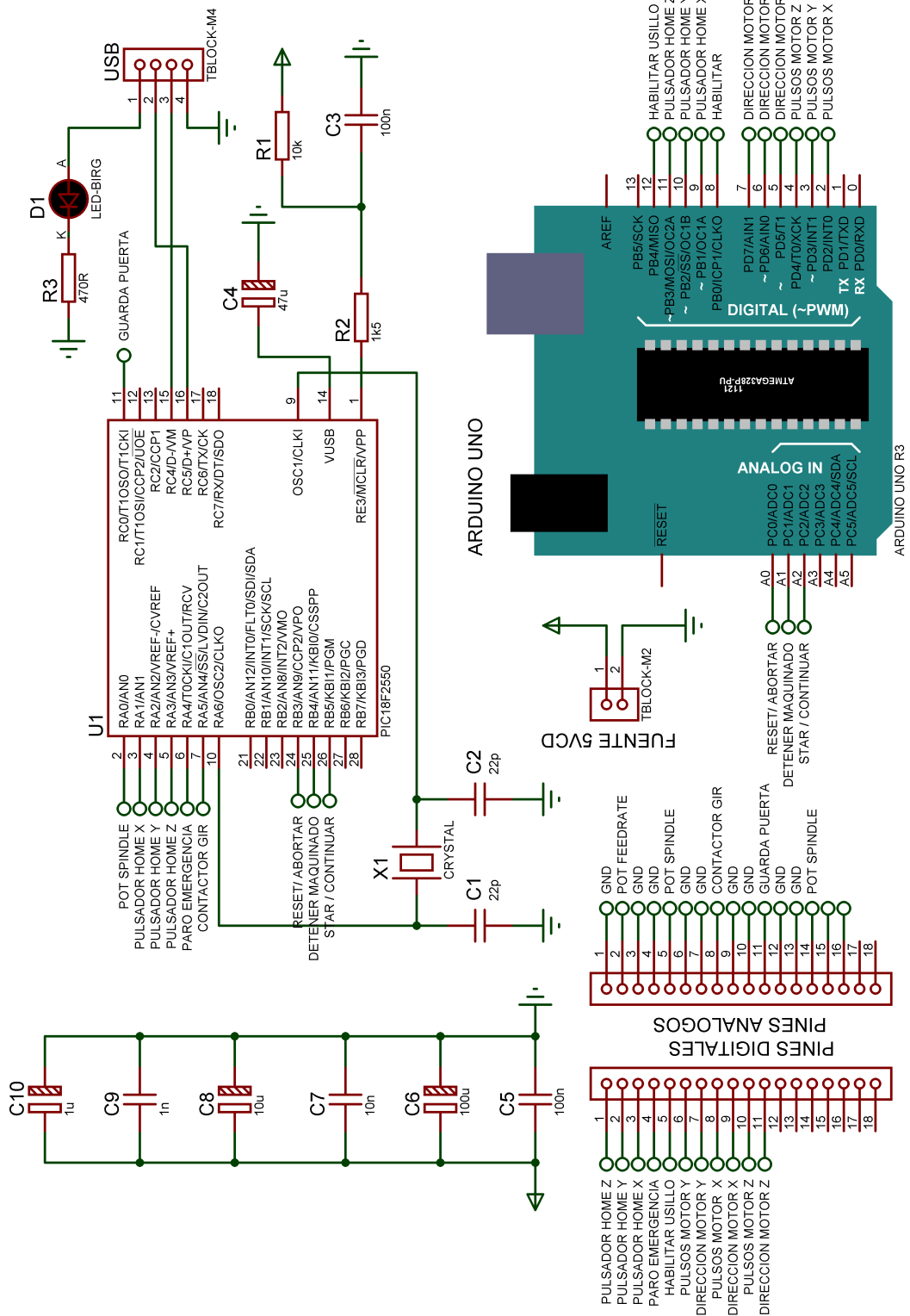
ΠΛΑΝΟ ΤΟΠΟΓΡΑΦΙΑΣ



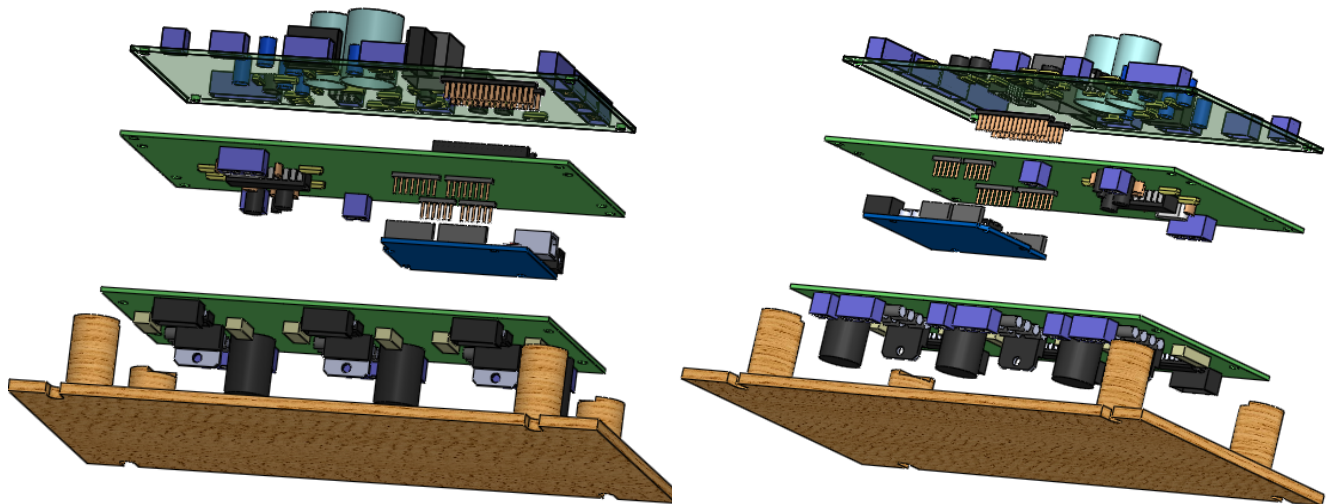
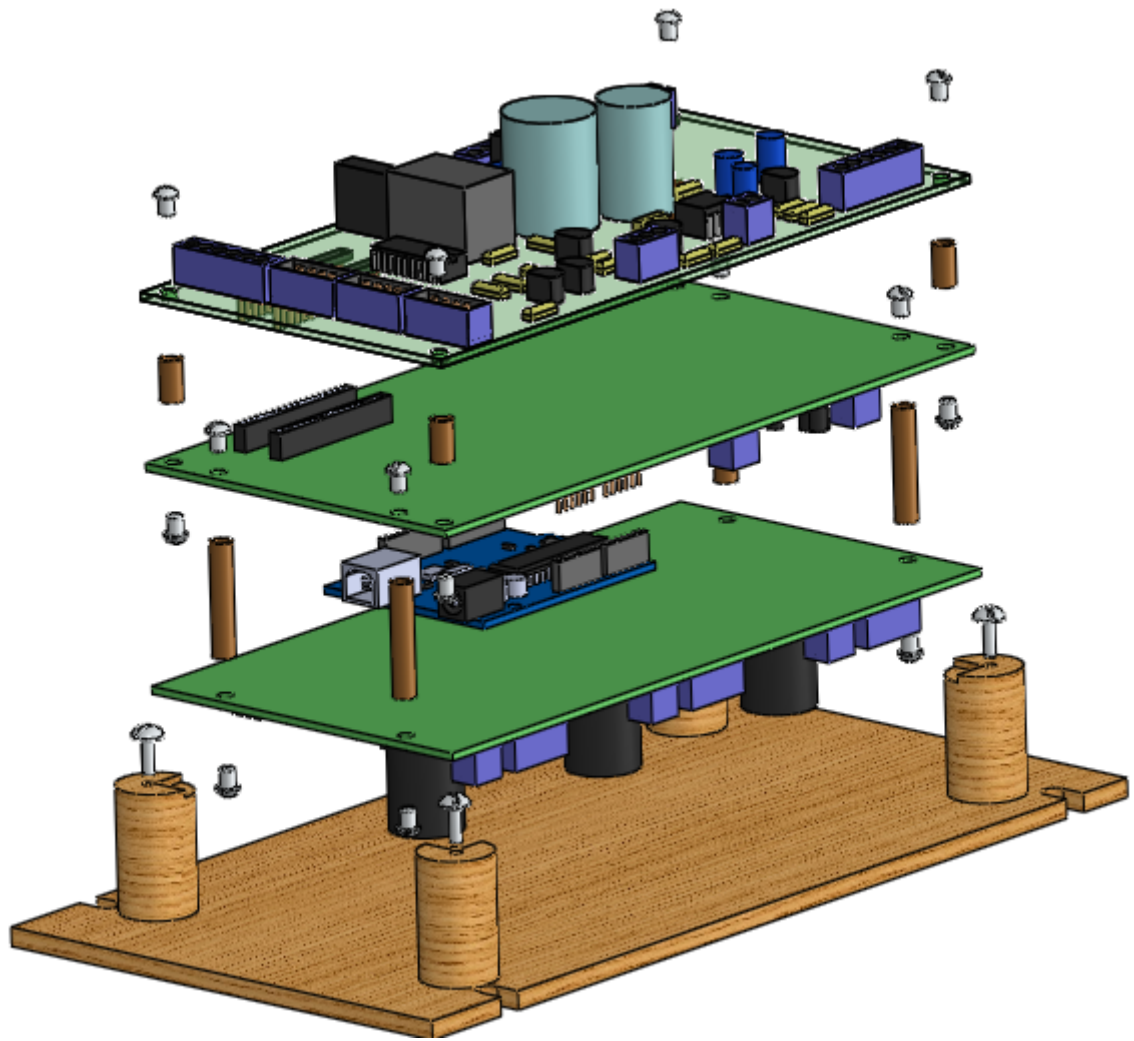
3D ΑΝΤΙΣΤΡΟΦΗ



UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: DISEÑO PCB	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		MÓDULO 1 (sección 2 de 2)	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO: MÁQUINA FRESADORA CNC	A4
DISEÑADOR: Jesús Rodríguez	ANEXO 16		ANEXO 16
NOTA	ESCALA: 1:10	PESO: NO determinado	HOJA 28 DE 31



UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO: CIRCUITO ESQUEMÁTICO	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		MÓDULO 2 Tarjeta de comunicación	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO: MÁQUINA FRESADORA CNC	A4
DISEÑADOR: Jesús Rodríguez	ANEXO 17		ANEXO 17
NOTA	ESCALA: 1:10	PESO: NO determinado	HUJA 29 DE 31



UNIVERSIDAD TÉCNICA DE AMBATO		TÍTULO:	
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL CARRERA DE INGENIERÍA INDUSTRIAL Y PROCESOS DE AUTOMATIZACIÓN		ENSAMBLE 3D DEL MÓDULO 1, 2 Y 3	
REVISIÓN: ING. Wilian López	FECHA: 09/03/2015	PROYECTO:	A4
DISEÑADOR: Jesús Rodríguez	ANEXO 17		
NOTA	ESCALA: 1:10	PESO: NO determinado	HOJA 31 DE 31
		ANEXO 17	

Contenido

Sección 1	4
Introducción	4
Sección 2.....	5
Características de la PC.....	5
Sección 3.....	6
Conexión máquina CNC	6
Sección 4.....	7
Instalación del software.....	7
Sección 5.....	17
Inicio y configuración del software.....	17
Sección 6.....	27
Ejemplo de estructura de código G	27
Sección 7.....	32
Ejemplo de maquinado CNC	32
Sección 8.....	36
Mantenimiento del equipo CNC	36

CONDICIONES DE SEGURIDAD

Leer las siguientes medidas de seguridad con objeto de evitar lesiones a personas y prevenir daños al equipo CNC y a los productos conectados a él.

El aparato sólo podrá repararlo personal autorizado y capacitado en electrónica y control previamente revisado la documentación referente a planos y manuales.

El personal encargado del laboratorio de CNC no se responsabiliza de cualquier daño físico o material derivado del incumplimiento de estas normas básicas de seguridad.

PRECAUCIONES ANTE DAÑOS A PERSONAS





- Interconexión de módulos.
Utilizar los cables de unión proporcionados con el aparato.
- Utilizar cables de red apropiados.
Para evitar riesgos, utilizar sólo cables de red recomendados para este aparato.
- Evitar sobrecargas eléctricas.
Para evitar descargas eléctricas y riesgos de incendio no aplicar tensión eléctrica fuera del rango seleccionado en la parte posterior de la unidad central del aparato.
- Conexionado a tierra.
Con objeto de evitar descargas eléctricas conectar las bornas de tierra de todos los módulos al punto central de tierras. Asimismo, antes de efectuar la conexión de las entradas y salidas de este producto asegurarse de que la conexión a tierras está efectuada.
- Antes de encender el aparato cerciorarse de que se ha conectado a tierra.
Con objeto de evitar descargas eléctricas cerciorarse de que se ha efectuado la conexión de tierras.
- No trabajar en ambientes húmedos.
Para evitar descargas eléctricas trabajar siempre en ambientes con humedad relativa inferior al 90% sin condensación a 45 °C.
- No trabajar en ambientes explosivos.
Con objeto de evitar riesgos, lesiones o daños, no trabajar en ambientes explosivos.



PRECAUCIONES ANTE DAÑOS AL EQUIPO CNC

- Ambiente de trabajo.
Este aparato está preparado para su uso en ambientes industriales debe tener precaución si se monta en otro tipo de condiciones (ambientes residenciales o domésticos).
- Instalar el aparato en el lugar apropiado.
Se recomienda que, siempre que sea posible, la instalación de la máquina CNC se realice alejada de líquidos refrigerantes, productos químicos, golpes, etc. que pudieran dañarlo.
Es aconsejable mantenerlo apartado de fuentes de perturbación electromagnética, como son:
 - Cargas potentes conectadas a la misma red que el equipo.
 - Transmisores portátiles cercanos (Teléfonos, radio/TV).
 - Máquinas de soldadura por arco cercanas.

No manipular el interior del aparato. Sólo personal encargado del laboratorio de CNC puede manipular el interior del aparato.

No manipular los conectores con el aparato conectado a la red eléctrica. Antes de manipular los conectores (entradas/salidas, captación, Etc.) cerciorarse de que el aparato no se encuentra conectado a la red eléctrica.

 PRECAUCIÓN:	La máquina Fresadora CNC DENFORD debe ser utilizada únicamente por personal formado y autorizado de conformidad con el Manual del operador y procedimientos e instrucciones de seguridad para la operación segura de la máquina.
 PRECAUCIÓN:	No existe un mensaje de advertencia que alerte al operador sobre cualquier posible choque.
 DANGER:	No acceda a la zona de mecanizado cuando la máquina esté en movimiento; pueden producirse lesiones graves o incluso la muerte.
 DANGER:	Las piezas fijadas de forma inadecuada o piezas sobredimensionadas podrían salir despedidas con una fuerza mortal.

 DANGER:	No intente anular las funciones de seguridad. De lo contrario, la máquina no funcionará de forma segura y se anulará la garantía de seguridad.
 PRECAUCIÓN:	Después de cada proceso de maquinado se debe realizar labores de limpieza y lubricación.

Sección 1

Introducción

Éste software es un paquete basado en programación gráfica LabVIEW para plataforma Windows que permite el control de los archivos de control numérico, directamente por línea de comando o por archivo de códigos G para máquina CNC.

La interfaz es amigable al usuario, los distintos botones son de fácil acceso y están ubicados de una forma lógica y agrupada por tareas para un uso más intuitivo. El editor de códigos G solo permite cargar archivos con extensión (*.txt; *.nc), y solo basta presionar el botón <INICIAR>, y la máquina CNC debe comenzar a realizar movimientos de acuerdo a las instrucciones enviadas.

Este software no tiene una licencia es de código abierto por lo que cualquier persona que esté interesada en realizar ajustes actualizaciones lo puede hacer siempre y cuando tenga uso del software LabVIEW.

Las características disponibles en el paquete de HMI de Máquina Fresadora CNC son las siguientes:

- Configuración y calibración por software de la máquina CNC.
- Monitoreo de estado del equipo en tiempo real.
- Visualización en 2D de maquinado CNC
- Desplazamiento manual por controles virtuales para selección del cero pieza.
- Visualización de posición en tiempo real.
- Indicador de texto para estado de máquina CNC.
- Botones para control de movimiento.
- Indicadores visuales de componentes y alertas.

Sección 2

Características de la PC

Para la instalación del software de control de la máquina Fresadora CNC en el PC, éste ordenador debe cumplir los siguientes requisitos de hardware y software. Para sistemas operativos basados en Windows.

WINDOWS	
Procesador	Pentium 4/M o equivalente
RAM	1 GB
Resolución de Pantalla	1024 x 768 píxeles
Sistema operativo	Windows 8.1/8/7/Vista (32 bits y 64 bits) Windows XP SP3 (32 bits)
Espacio en Disco	5 GB

Para sistemas operativos basados en Mac OS X.

Mac OS X	
Procesador	Procesador basado en Intel
RAM	2 GB
Resolución de Pantalla	1024 x 768 píxeles
Sistema operativo	OS X 10.8 o 10.9
Espacio en Disco	1.2 GB para la instalación completa

Para sistemas operativos basados en Linux.

Linux	
Procesador	Pentium 4/M o equivalente
RAM	1 GB
Resolución de Pantalla	1024 x 768 píxeles
Sistema operativo	Red Hat Enterprise Linux Desktop + Workstation 6 o posterior.
Espacio en Disco	1.4 GB para la instalación completa de LabVIEW de 32 y 64 bits

La máquina debe tener al menos un puerto USB disponible y una unidad lectora de CD, este puede ser instalado en una PC de escritorio o en un equipo portátil.

Sección 3

Conexión máquina CNC

CONEXIÓN DE LA MÁQUINA

La caja eléctrica se energiza con una fuente de alimentación de 220 VCA, ésta caja eléctrica tiene una manguera flexible de conexión por la cual se encuentran cables eléctricos que controlan y monitorean la máquina Fresadora CNC. Las conexiones deben hacer con precaución y como se ilustran en las siguientes gráficas.



Para poder comunicar las instrucciones del operario hacia la máquina se utiliza cualquier computador u ordenador de mediana capacidad que tenga al menos un puerto USB disponible, este cable de comunicación debe ser conectado a la caja eléctrica utilizando dos cables USB de tipo A-B macho. La siguiente gráfica indica el tipo de cable y la conexión entre la PC y la caja eléctrica.

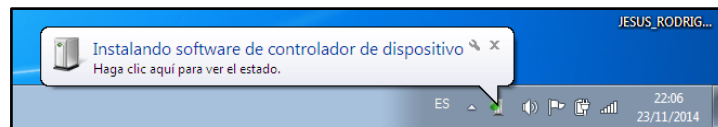


Sección 4

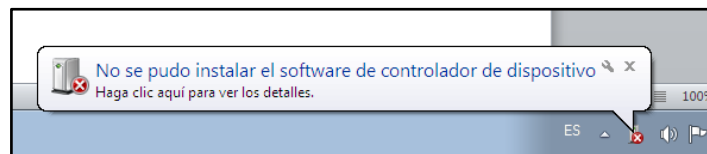
Instalación del software

INSTALACIÓN DRIVER CONTROLADOR PARA ARDUINO

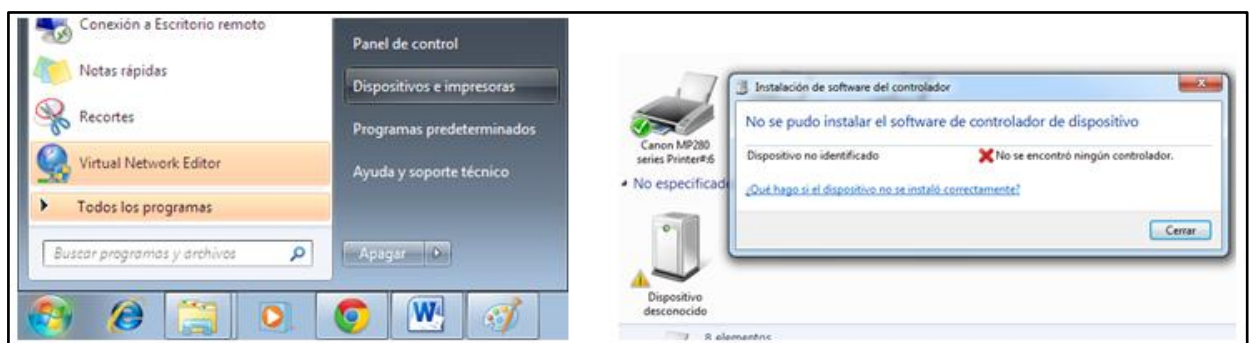
Para la instalación de éste dispositivo es necesario tener encendida la máquina Fresadora CNC y conectada por medio del cable de comunicación USB a la PC de control, como segundo paso hay que instalar los driver de control de los dispositivos para ello se debe seguir los siguiente pasos de instalación, el diálogo en la PC identificará que el dispositivo está siendo reconocido, en la siguiente gráfica se ilustra la identificación del dispositivo de control.



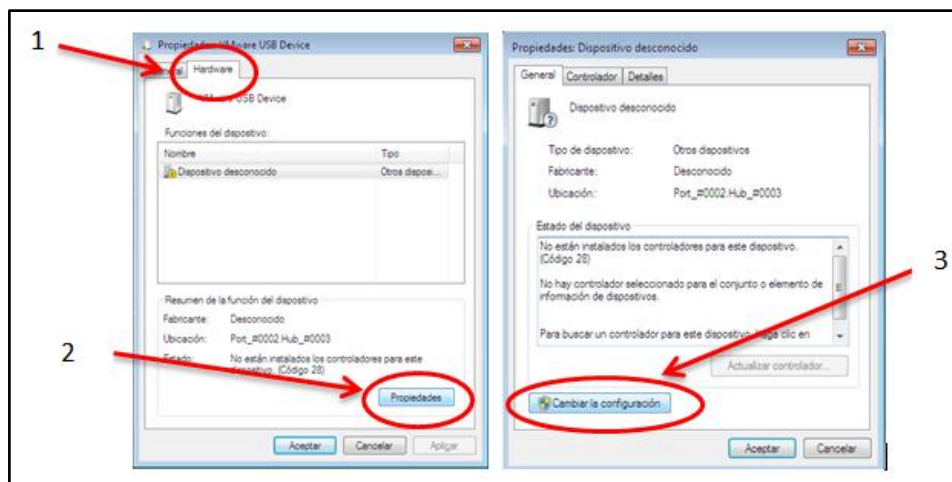
En este caso no se reconocerá el dispositivo conectado por lo que hay que hay que instalarlo de forma manual.



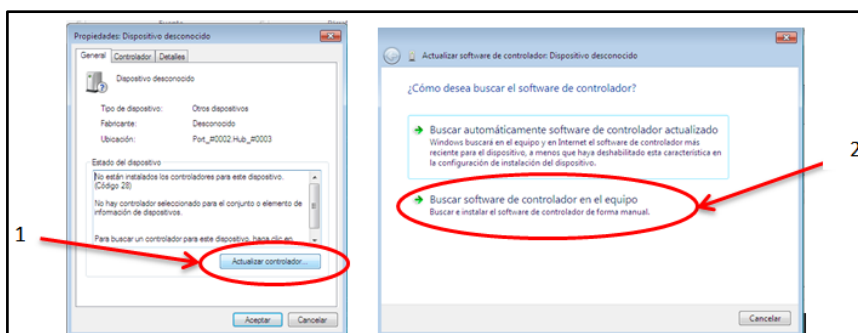
Para ello nos dirigimos a <inicio>, <Dispositivos e Impresoras>, ya en esta ventana seleccionaremos el dispositivo que esta con signo de admiración, el cual es el dispositivo que no se ha instalado bien.



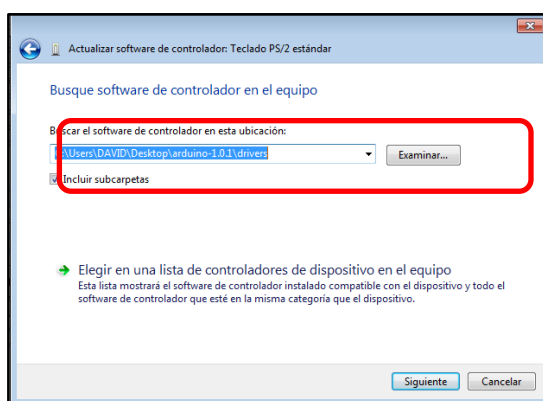
Sobre este dispositivo hacemos clic derecho y accedemos a propiedades, y escogemos la pestaña “Hardware”, ya en esta ventana escogemos “Propiedades”, y ejecutamos la instrucción “Cambiar la configuración”.



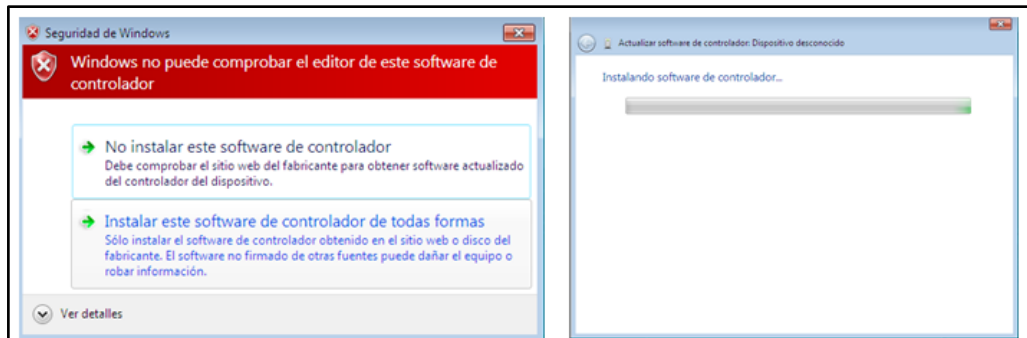
Ya en las propiedades del dispositivo, hacemos clic en “Actualizar controlador”, aparecerá el asistente “Actualizar software del controlador”, del cual seleccionamos “Buscar software de controlador en el equipo”.



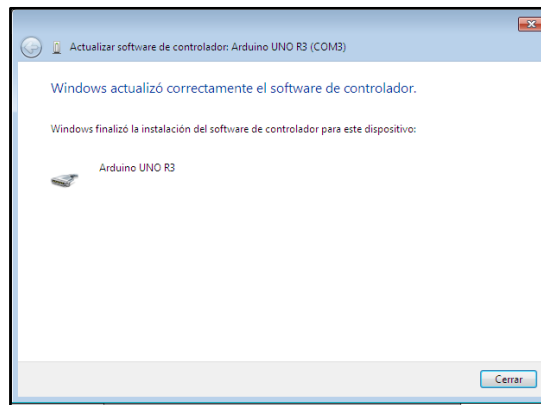
En la barra de direcciones hay que buscar la ubicación del “Software Instalador” y escogemos la carpeta “DRIVERS_CNC” y aceptamos, ver gráfica.



Inmediatamente se mostrara la ventana de seguridad de Windows, y se debe escoger la opción “Instalar este software de controlador de todas formas” e inmediatamente se procederá a instalar este recurso, vea la siguiente gráfica.

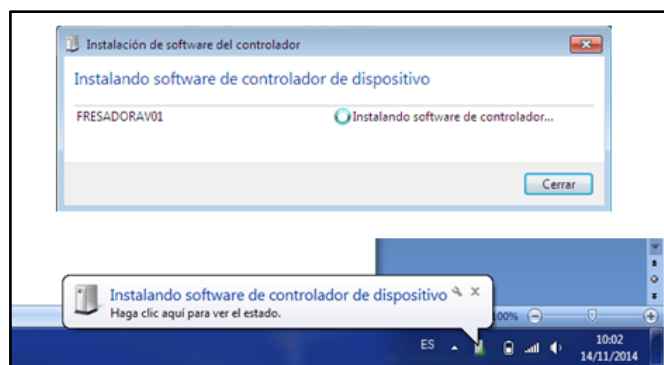


Terminada la instalación aparecerá la siguiente gráfica, con la cual se confirmará la instalación del dispositivo controlador de la máquina fresadora CNC.

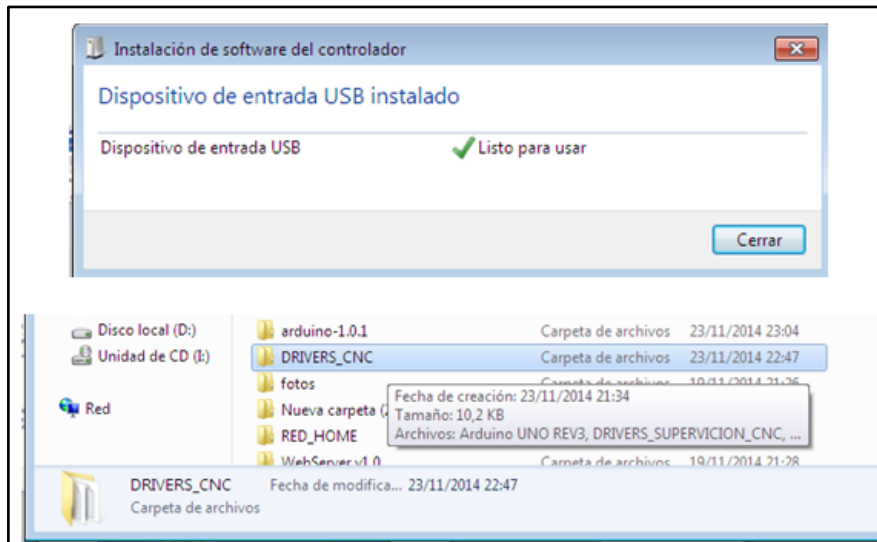


INSTALACIÓN DRIVER DE SUPERVISIÓN “FRESADORAV01”

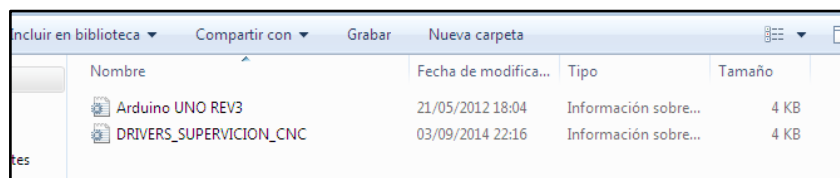
Para la instalación de éste dispositivo es necesario tener encendida la máquina Fresadora CNC y conectada por medio del cable de comunicación USB a la PC de control, el diálogo en la PC identificará que el dispositivo está siendo reconocido, en la siguiente gráfica se ilustra la identificación del dispositivo supervisor.



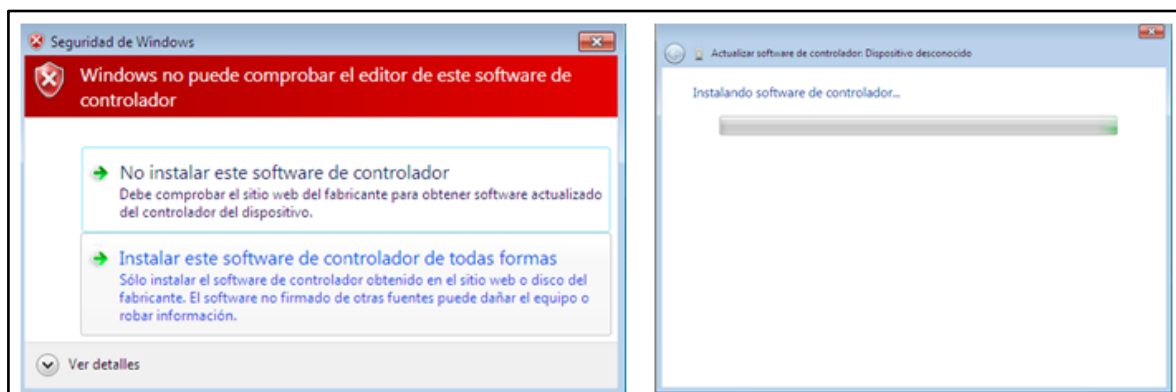
La PC reconoce a éste dispositivo como “Dispositivo de entrada USB” sin que haya ningún conflicto; Ha este dispositivo se le debe actualizar el driver controlador y para ello nos dirigimos a donde se encuentra ubicado el “Software Instalador” y escogemos la carpeta “DRIVERS_CNC” ver siguiente gráfica.



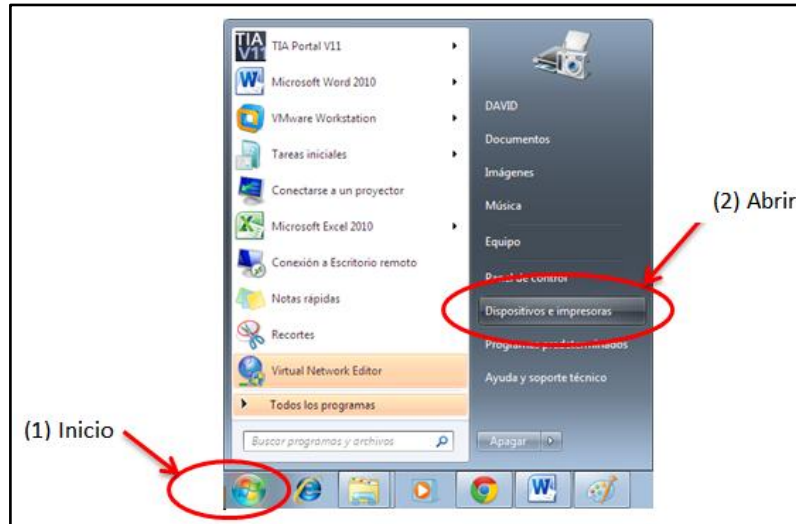
Ya abierta la carpeta se encontraran dos archivos, seleccionamos el archivo “DRIVERS_SUPERVCIÓN_CNC”, hacemos sobre este archivo clic derecho y seleccionamos ejecutar como administrador, con el cual podremos instalar este recurso.



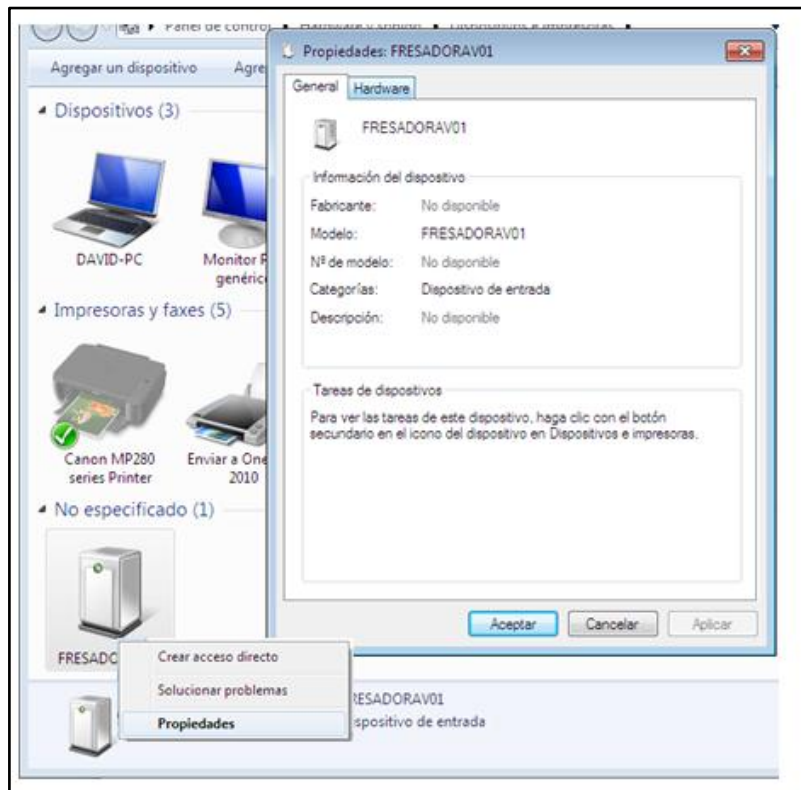
Inmediatamente se mostrara la ventana de seguridad de Windows, y se debe escoger la opción “Instalar este software de controlador de todas formas” e inmediatamente se procederá a instalar este recurso, vea la siguiente gráfica.



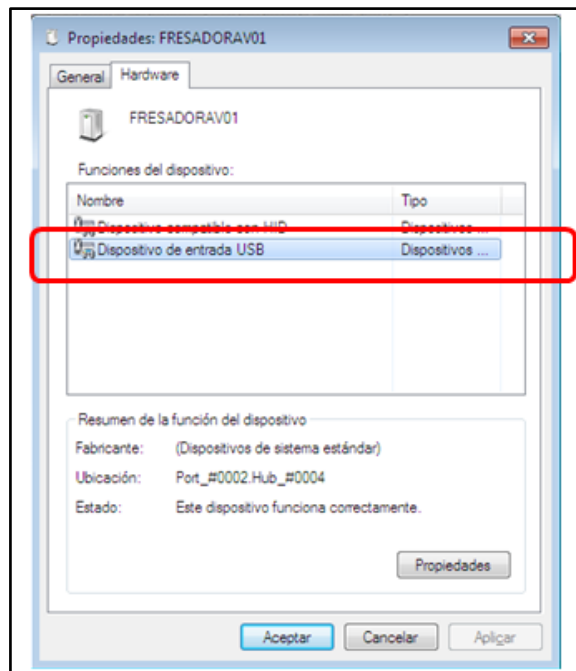
Para que la PC reconozca este dispositivo como el “dispositivo supervisor de la máquina CNC”, hay que actualizar el driver y para ello hay que dirigirse a la dirección “administrador de dispositivos”, para ello hacemos clic en <Inicio>, <Equipo>, <Dispositivos e impresoras> como se indica en la siguiente gráfica.



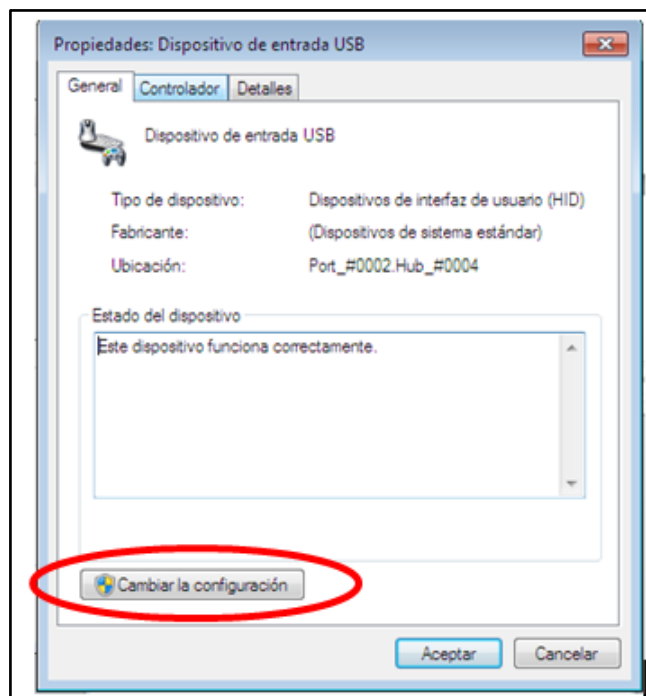
Al abrir la ventana “Dispositivos e impresoras”, nos aparecerá la siguiente ventana mostrando el dispositivo USB denominado “FRESADORAV01”; Abrimos las propiedades de éste recurso como se indica en la siguiente gráfica.



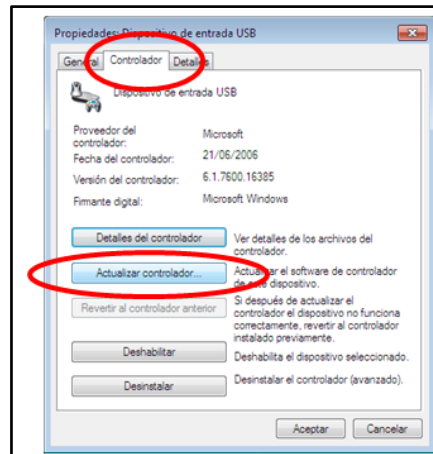
Ya en la ventana de propiedades seleccionamos “Hardware”, y escogemos en funciones del dispositivo “Dispositivo de entrada USB” haciendo doble clic, como se indica en la siguiente gráfica.



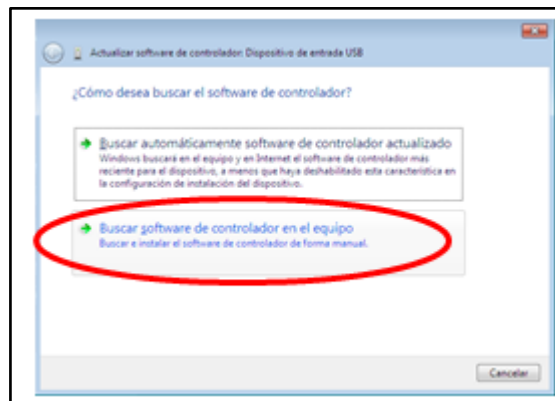
Ya en la ventana de “Propiedades Dispositivo de entrada USB”, seleccionamos “Cambiar la configuración”.



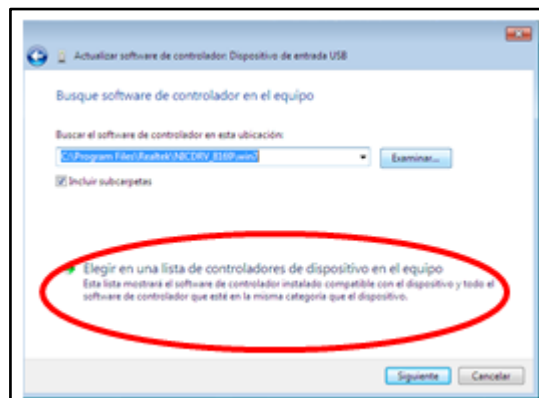
Con la cual podremos cambiar e instalar el controlador apropiado para este dispositivo, ya en la ventana seleccionaremos “Controlador”, y en ella hacemos clic en “Actualizar controlador”, como se indica en la siguiente gráfica.



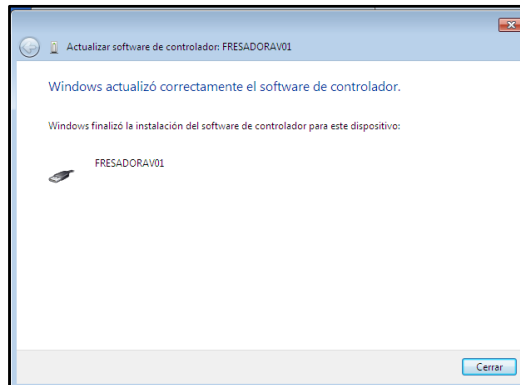
En la siguiente ventana aparecerá el asistente para instalar el software de control del dispositivo USB conectado, escogemos “Buscar software de controlador en el equipo”.



En la siguiente ventana hay que escoger “Elegir en una lista de controladores de dispositivos en el equipo”.



En la lista escogemos “FRESADORA V01” y aceptamos y lograremos que la máquina CNC este en línea, como se indica en la siguiente gráfica.



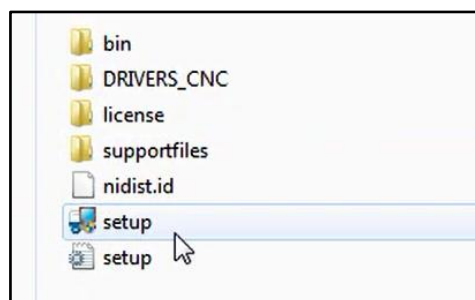
INSTALACIÓN DEL SOFTWARE

Siga estas instrucciones para instalar el software de control para la máquina Fresadora CNC en su ordenador (PC):

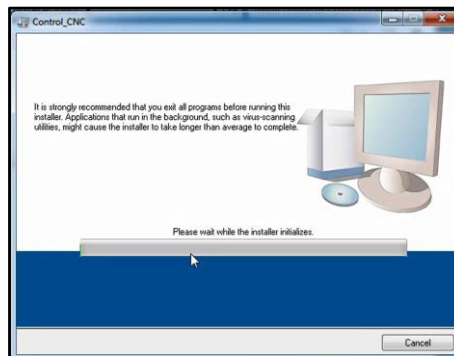
- 1) Encienda el ordenador e inicie Windows XP sp-3, 7, 8.
- 2) Inserte el CD-ROM, o el dispositivo de almacenamiento con los archivos de instalación para la máquina de fresado.
- 3) Abra el contenido del Archivo de instalación del software.



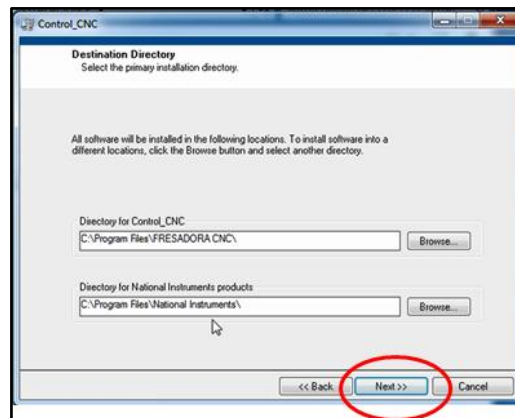
- 4) En el contenido de archivos necesarios para la instalación se mostrarán el archivo denominado "Setup.exe" para iniciar el programa de instalación.



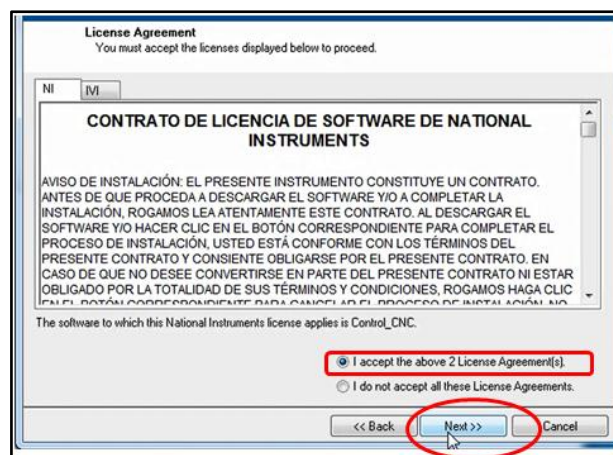
5) Haga clic derecho sobre el archivo “Setup.exe” y seleccione “Ejecutar como administrador” para proceder con la instalación.



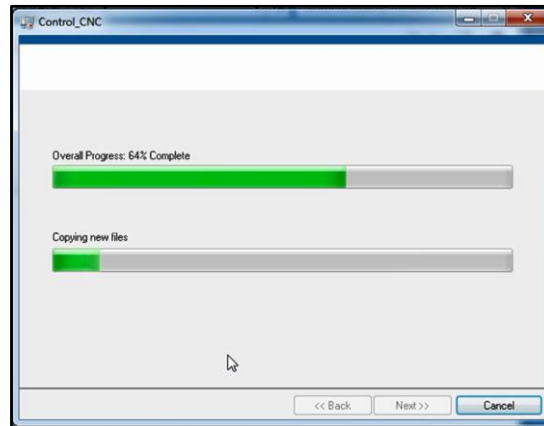
6) En el proceso de instalación se debe escoger la ruta o dirección en donde se desea instalar el software de control para la máquina fresadora CNC y presionar “Next”.



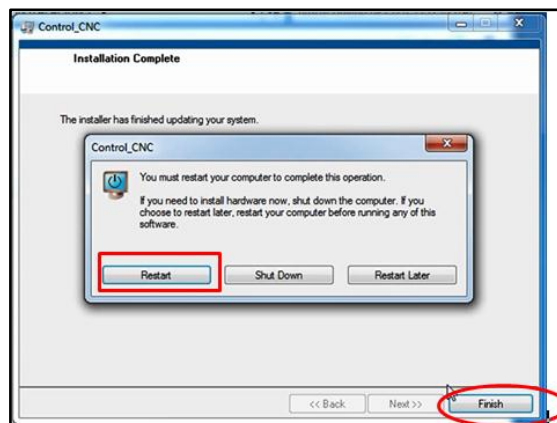
7) Al ser un software de licencia es necesario aceptar los términos y condiciones para el uso de este paquete de instalación caso contrario no se podrá hacer uso de este archivo.



8) El tiempo de instalación dependerá de la velocidad del equipo, pero es aproximadamente 10 minutos tiempo en el cual se copiará y configurará el software de control en la PC.



9) Al terminar la instalación hay que hacer clic en "Finish" y se pedirá que el equipo debe reiniciar "Restart" para completar el proceso y actualización del software de control de la máquina Fresadora CNC.



Sección 5

Inicio y configuración del software

VENTANA “LOGO”

Al ejecutar el software de control ya instalado aparecerá como ventana de presentación inicial “LOGO”, en la cual se mostrarán dos botones “INICIO” con el cual arrancaremos el control de máquina Fresadora CNC, y el botón de “SALIR” cuya función es cerrar la aplicación.



Cuando se presiona el botón de “INICIO” el software ejecutará el proceso para buscar y verificar que la máquina este encendida y conectada a la PC, en la ventana “BUSCAR_CNC” se encuentra el botón denominado “VERIFICAR COMUNICACION” al presionar éste momento empezará a buscar los dispositivos de control y supervisión del equipo CNC, si no hay ningún problema en la búsqueda aparecerá en la ventana de diálogo el mensaje “Sistema en Línea..!”, en éste momento se podrá realizar dos opciones la primera “COMPROBAR CÓDIGO” o “MAQUINAR”.

VENTANA “BUSCAR_CNC”



En esta ventana “BUSCAR_CNC” el software buscará los dispositivos de control físicos de la máquina CNC sin los cuales no podrá ejecutar ninguna acción, a continuación se describirá la función de cada uno de los elementos que conforman ésta ventana.

1. “VERIFICAR COMUNICACIÓN”, botón para iniciar la búsqueda y actualización de los dispositivos de control físico de la máquina CNC.
2. “VOLVER MENU”, botón para regresar a la pantalla “LOGO”.
3. “COMPROBAR CÓDIGO”, botón para iniciar la aplicación que verificará el código G con el cual se va a trabajar.
4. “MAQUINAR”, con éste botón se podrá ingresar sin verificar el código G de trabajo directamente para la ejecución.
5. “SALIR”, botón por el cual se podrá cerrar la aplicación de control de la máquina Fresadora CNC.
6. Pantalla de visualización de contenido con el cual indica el proceso que está realizando.
 - “Sistema en línea...!!”, indica que la máquina está lista para el trabajo.

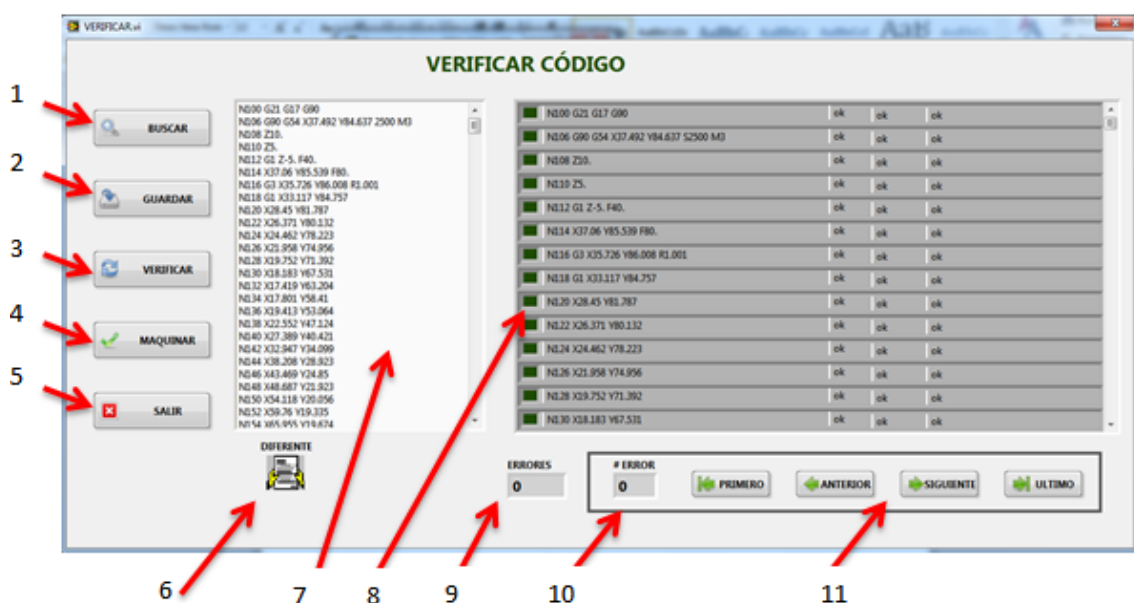
- “ERROR ARDUINO...!!”, indica que el controlador Arduino no está conectado, probablemente ésta apagado o no lo reconoce.
 - “ERROR PIC...!!”, indica que el controlador de Supervisión no está conectado, probablemente ésta apagado o no lo reconoce.
7. Indica la dirección del puerto de comunicación con el cual lo reconoció al dispositivo Supervisor de la máquina Fresadora CNC.
 8. Indica la dirección del puerto de comunicación con el cual lo reconoció al dispositivo Controlador Arduino de la máquina Fresadora CNC.

NOTA:

Al presionar el botón “VERIFICAR COMUNICACIÓN”, el software buscará los dispositivos, si uno o los dos dispositivos no son reconocidos en 20 segundos se mostrará un mensaje indicando cuál de los componentes no ha sido reconocido, con lo cual el usuario debe verificar que la máquina está encendida y conectada por medio del cable de comunicación a la PC.

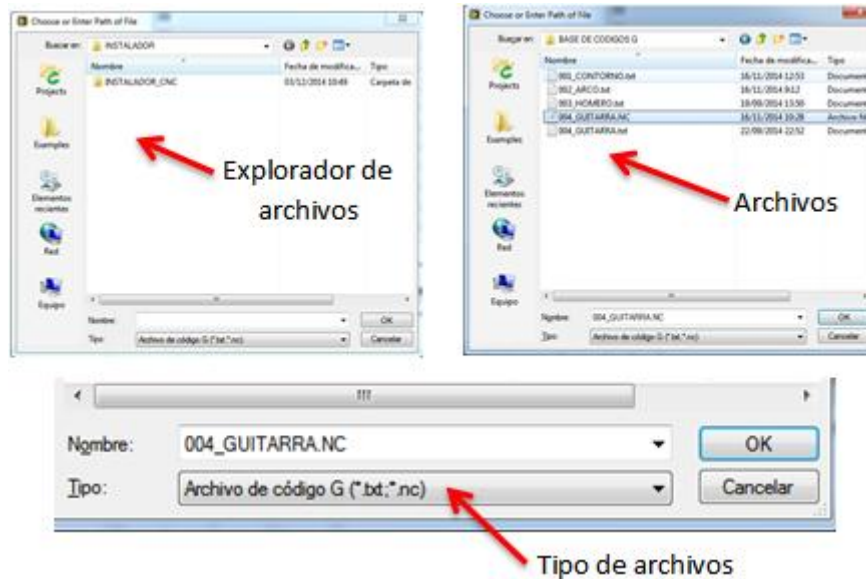
VENTANA “VERIFICAR”

La ventana “VERIFICAR” permite realizar una verificación del código G previa al la operación de maquinado, es decir revisará si el código G es compatible con la máquina CNC o si en su estructura tiene errores de contenido. En el caso de encontrar errores se indicara el tipo de error y la línea en la que se encuentra.



En la siguiente lista se indicara la función de cada uno de los elementos que contiene esta ventana.

1. “BUSCAR”, con éste botón abre el cuadro de búsqueda en el cual podemos buscar el código que deseamos verificar para un maquinado, los archivos de código G aceptados en el software de control de la máquina fresadora CNC son de tipo (*.txt; *.nc).



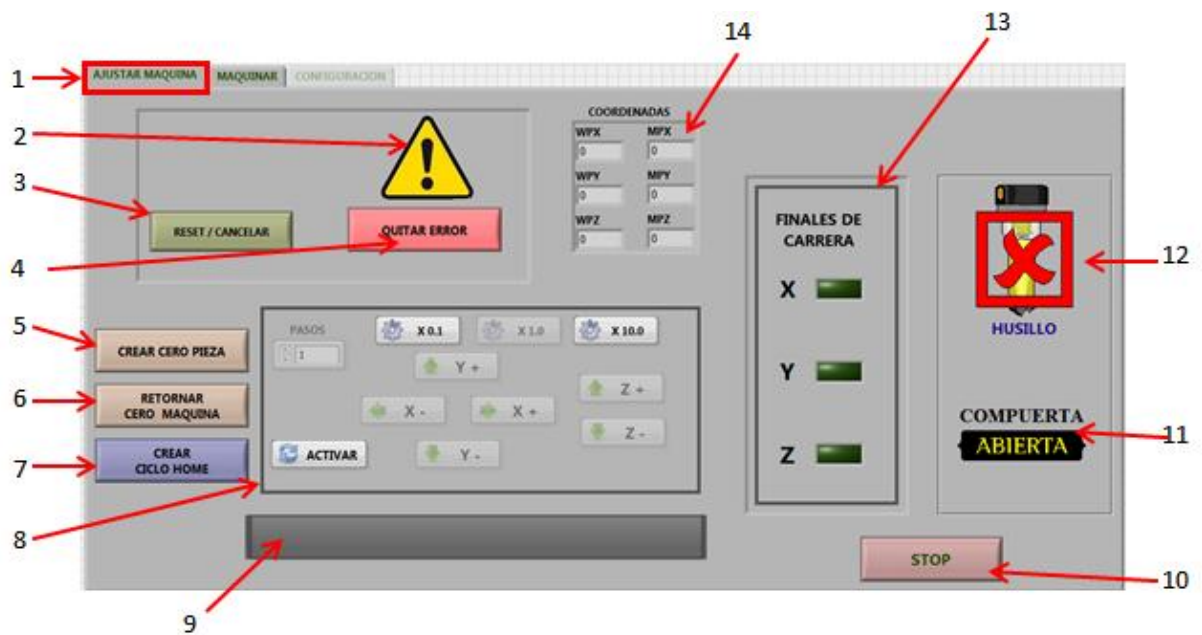
Una vez encontrado el archivo de código G se presiona el botón “OK” y se cargará y mostrará en la ventana de código.

2. “GUARDAR”, por medio de este botón se puede guardar los cambios realizados en el código G producto de una modificación o edición si se llegara a necesitar. Cuando se realiza algún cambio en el código se mostrará en la parte inferior un icono con el nombre “DIFERENTE” el cual indica que el código G ha sido cambiado por lo que debe guardar o deshacer los cambios.
3. “VERIFICAR”, con éste botón se realiza el proceso de verificación del código, este proceso puede tardar algunos segundos, como forma de indicar que el proceso está en curso se indica en un contador numérico las líneas de código G verificadas.
4. “MAQUINAR”, una vez verificado el código G con éste botón se puede proceder a maquinar.

5. “SALIR”, con éste botón se puede cerrar la aplicación si no se desea realizar ninguna operación.
6. Icono “DIFERENTE”, es un indicador y aparece solo cuando el código G ha sido modificado o editado.
7. Ventana de código, en ésta ventana se carga el código G con el cual se desea trabajar.
8. Tabla, en esta tabla se muestra luego de la verificación si ha sido aceptado por la máquina o no, muestra también si se presenta algún error en su estructura o si existe alguna violación a los códigos G admitidos.
9. Número de errores, indica el número de errores encontrados en el proceso de verificación.
10. Número de error, indica el error y ayuda como guía para realizar correcciones.
11. Botones de desplazamiento, son botones que permiten navegar por lo errores encontrados y ayuda como guía para realizar correcciones.

VENTANA “MAQUINAR” - <AJUSTAR MÁQUINA>

En esta ventana se tiene todo el proceso de posicionamiento y movimientos iniciales antes de proceder a realizar un proceso de maquinado.



1. “AJUSTAR MÁQUINA”, en esta pestaña se puede hacer operaciones de posicionamiento inicial antes de iniciar un proceso de maquinado.
2. “ADVERTENCIA”, es un icono que se visualiza para indicar que no se podrá hacer ningún tipo de movimiento por las siguientes circunstancias: porque ha ocurrido un error en el proceso, se ha activado un paro de emergencia, o cuando se enciende la máquina luego de un corte de energía.

NOTA:

Al encender la máquina CNC y controlarla con la PC siempre iniciará con señal de alarma, y esto es por seguridad ya que la máquina no tiene sistemas de encoder como para almacenar la posición luego de un corte de energía o cuando se apaga.

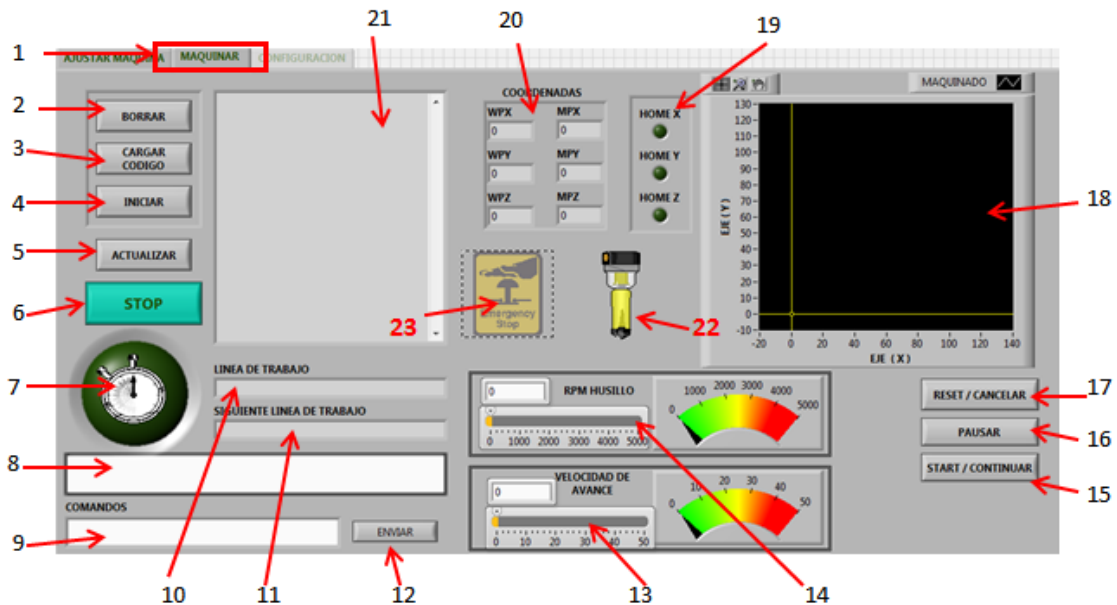
Siempre que se enciende la máquina CNC arrancará con error por lo que hay que revisar que no haya ningún tipo de objeto que obstaculice los movimientos de desplazamiento de los ejes en la máquina CNC.

3. “RESET/CANCELAR”, sirve para dar al controlador de la máquina CNC un reset para actualizar el estado del equipo, para borrar o detener un proceso de maquinado por emergencia.
4. “QUITAR ERROR”, sirve para desbloquear el sistema de error, poder realizar las operaciones de posicionamiento, regreso a home o realizar un maquinado.
1. “CREAR CERO PIEZA”, con este botón podemos seleccionar un nuevo cero pieza, luego de un proceso de posicionamiento manual utilizando los botones de navegación de cada uno de los ejes.
5. “RETORNAR CERO MÁQUINA”, regresa los ejes de desplazamiento de mesa y husillo al cero máquina, siempre y cuando se haya realizado un ciclo home.
6. “CREAR CICLO HOME”, con esto activamos el proceso de regreso a home y escoger el cero máquina, cuando se enciende la máquina o se restaura luego de un corte de energía la máquina CNC no sabrá en qué posición se encuentra por lo que realizar un ciclo HOME ayuda a tener un eje del cual partir para realizar todos los desplazamientos.
7. Panel de control para posicionamiento, sirve para dar instrucciones de movimiento para un desplazamiento controlado, éstos botones de dirección sirven para posicionar y seleccionar en la máquina fresadora CNC el cero pieza.

8. Pantalla de visualización, en el indica que proceso se está haciendo o el estado en el que se encuentra. La lista de los mensaje son los siguientes:
- “Error revise sistema..!”, ocurre cuando se presiona un paro de emergencia.
 - “Activado modo chequeo”, se activa para enviar el código G sin realizar ningún movimiento.
 - “Maquinando..!”, indica que se encuentra en un proceso de maquinado.
 - “Sistema listo..!”, indica que el controlador está listo para recibir un comando o una secuencia de comandos de códigos G.
 - “Esperando Star..!”, indica que se producido un pause en el proceso de maquinado.
 - “Deteniendo maquinado...!”, indica que el operario está realizando un proceso de “PAUSE” por lo que la maquina ira disminuyendo la velocidad antes de detenerse..
 - “Regresando a Home..!”, indica que se está realizando un proceso de HOME AUTOMÁTICO.
9. “STOP”, sirve para detener el proceso de maquinado y salir de la aplicación.
10. “PUERTA”, este icono se activa para indica que la puerta está abierta o cerrada según el estado.
11. “HUSILLO”, cuando el icono está con una X roja, indica que el husillo está apagado y bloqueado.
12. “FINALES DE CARRERA”, son indicadores gráficos que muestran si alguno de los finales de carrera están activos por lo que hay que detener el proceso de maquinado por seguridad siempre y cuando no se encuentre realizando un ciclo home.

VENTANA “MAQUINAR” - <MAQUINAR>

En esta pestaña de la ventana “MAQUINAR” se realiza las operaciones fundamentales de maquinado como son selección del cero pieza, carga de archivos G para maquinado, monitoreo en tiempo real de dispositivos como finales de carrera, puerta, y estado del husillo entre otras funciones básicas para un proceso de maquinado.



1. “MAQUINAR”, pestaña principal para operaciones de maquinado.
2. “BORRAR”, borra el archivo de instrucciones cargadas y refresca la memoria del software y lo deja listo para un nuevo proceso de maquinado.
3. “CARGAR CÓDIGO”, con este botón permite abrir el explorador de archivos para escoger un código G de un diseño para su posterior maquinado.
4. “INICIAR”, ya cargada la lista de códigos G este botón permite iniciar el proceso de maquinado de forma automática.
5. “ACTUALIZAR”, actualiza la posición y el estado en que se encuentra la máquina y la tarjeta controladora.
6. “STOP”, detiene el proceso de maquinado y cierra la aplicación.
7. “RUN”, éste icono indica que está ejecutando una instrucción de maquinado o que el sistema ésta esperando una nueva instrucción.
8. Pantalla de visualización, en él se indica que proceso está haciendo o en qué estado se encuentra la máquina CNC. La lista de los mensajes son los siguientes:

- “Error revise sistema..!”, ocurre cuando se presiona un paro de emergencia.
 - “Activado modo chequeo”, se activa para enviar el código G sin realizar ningún movimiento.
 - “Maquinando..!”, indica que se encuentra en un proceso de maquinado.
 - “Sistema listo..!”, indica que el controlador está listo para recibir un comando o una secuencia de comandos de códigos G.
 - “Esperando Start..!”, indica que se producido un pause en el proceso de maquinado.
 - “Deteniendo maquinado...!”, indica que el operario está realizando un proceso de “PAUSE” por lo que la maquina ira disminuyendo la velocidad antes de detenerse.
 - “Regresando a Home..!”, indica que se está realizando un proceso de HOME AUTOMÁTICO.
9. Barra de comandos, en ella se pueden escribir instrucciones individuales o secuencia de códigos G para maquinar.
10. “LÍNEA DE TRABAJO”, indica que instrucción se encuentra realizando en él proceso de maquinado.
11. “SIGUIENTE LÍNEA DE TRABAJO”, indica la siguiente instrucción de código G a maquinar.
12. “ENVIAR”, ya escrito el comando de código G a maquinar en la barra de comandos, este botón permite enviar para que el controlador ejecute la acción.
13. “VELOCIDAD DE AVANCE”, en ella se visualiza la velocidad de maquinado en mm/min la velocidad de maquinado recomendado es de (5 a 50) mm/min.
14. “VELOCIDAD DE HUSILLO”, en ella se puede configurar desde el software la velocidad en RPM a la que se desea que trabaje el husillo.
- NOTA: Ésta función esta deshabilitada en la caja eléctrica de la máquina CNC, si se desea activarla hay que recurrir a los planos eléctricos y conectar los canales análogos del driver del husillo a la tarjeta controladora. Ésta operación debe ser realizada por personal autorizado y calificado.
15. “START/CONTINUAR”, con este botón se puede reanudar el proceso de maquinado luego de un ciclo “PAUSE”.

16. “PAUSAR”, con este botón se puede detener el proceso de maquinado sin perder la posición en la que se quedó.
17. “RESET/CANCELAR”, con éste botón se puede borrar una alarma producto de un error, se puede refrescar la memoria, o detener un proceso de maquinado.
18. Visualizador 2D, aquí se dibujara en dos dimensiones las trayectorias realizar en un proceso de maquinado.
19. Indicadores de finales de carrera activos, indica si alguno de los terminales de carrera están activos por lo cual se detendrá el proceso de maquinado por seguridad del equipo.
20. “COORDENADAS”, indica en qué posición se encuentra tanto en coordenadas de maquina como de dibujo.
21. Ventana de códigos G, aquí se visualiza la lista de comando de códigos G a maquinar.
22. “HUSILLO”, cuando éste icono se activa, indica que el husillo de la máquina fresadora CNC está encendida.
23. “PUERTA”, cuando éste icono se activa indica que la puerta está abierta y por tanto no se activara el botón de “INICIO”, y no podrá encender el husillo.

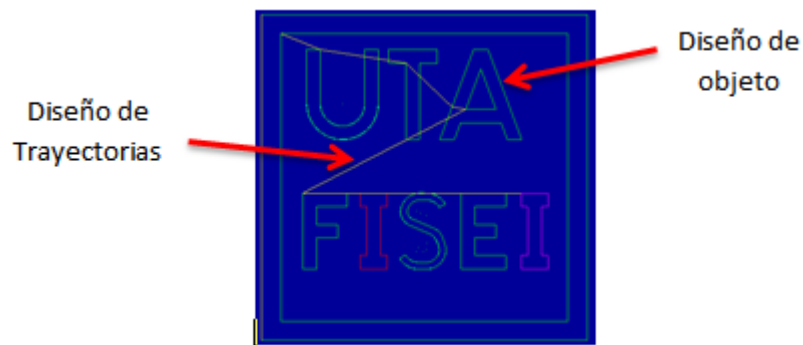
Sección 6

Ejemplo de estructura de código G

Para extraer códigos G de un diseño se recomienda utilizar el software MastercamX en el cual se puede diseñar un objeto en 2D, 3D y crear las respectivas rutas de maquinado para su posterior extracción del archivo de códigos CNC.

Paso para sacar el código G.

1. Diseñar el objeto en MastercamX con sus respectivas rutas de movimiento de herramienta.

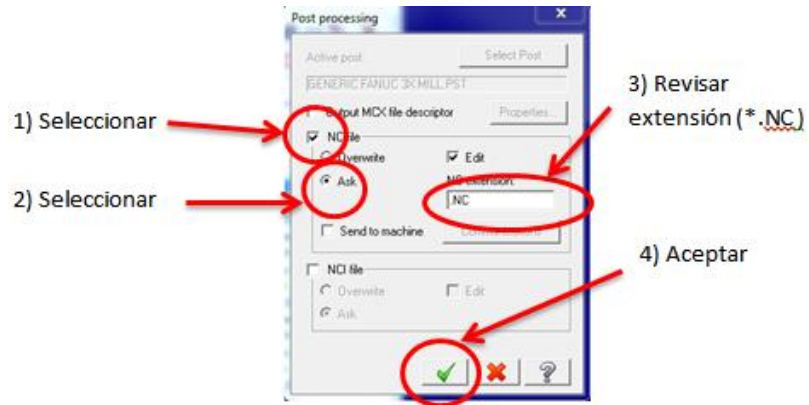


2. Extraer código CNC

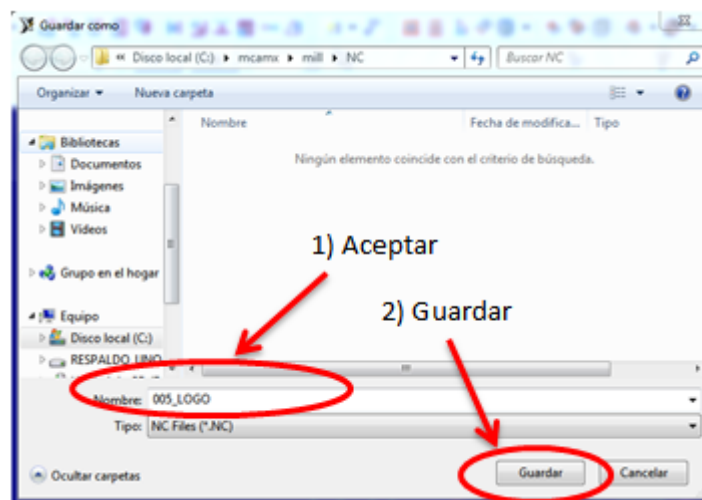
- a. En la barra de diseño de trayectorias y maquinado hay que escoger 1 activar todas las operaciones de maquinado, 2 crear código G.



- b. En la ventana siguiente se escogerá las opciones para crear el archivo de código G, en éste caso se escogerá la extensión de archivo (*.NC) y se acepta los cambios.



- c. En la ventana del explorador de archivos se ubica la dirección donde se va a guardar el archivo y el nombre con el cual se lo va a identificar.



- d. Cuando se haya guardado el archivo se abrirá el editor de texto del MastercamX en el cual hay que realizar unas modificaciones con el fin de que la máquina CNC los admita sin inconveniente alguno.

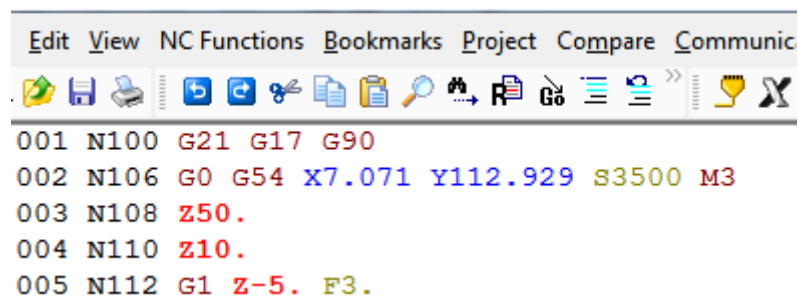
```
Mastercam X Editor - [C:\USERS\COMPUEDIT\DESKTOP\BASE DE CODIGOS G\005_LOGO.NC]
File Edit View NC Functions Bookmarks Project Compare Communications Tools Window Help
New...
001
002 O0000 (005_LOGO)
003 (DATE=DD-MM-YY - 11-12-14 TIME=HH:MM - 09:36)
004 (MCM FILE - C:\USERS\COMPUEDIT\DESKTOP\BASE DE CODIGOS G\005_LOGO.MCM)
005 (NC FILE - C:\USERS\COMPUEDIT\DESKTOP\BASE DE CODIGOS G\005_LOGO.NC)
006 (MATERIAL - ALUMINUM M1 - 2024)
007 ( T212 | 3. FLAT ENDMILL | H0 )
008 N100 G21
009 N102 G0 G17 G40 G49 G80 G90
010 N104 T212 M6
011 N106 G0 G90 G54 X7.071 Y112.929 Z3500 M3
012 N108 G43 H0 Z50.
013 N110 Z10.
014 N112 G1 Z-5. F3.
015 N114 X112.929
016 N116 Y7.071
017 N118 X7.071
018 N120 Y112.929
019 N122 Z5.
020 N124 G0 Z50.
021 N126 X21.564 Y107.063
022 N128 Z10.
023 N130 G1 Z-5.
024 N132 Y7.063
025 N134 G5 X21.566 Y86.903 R8.201
026 N136 X37.964 Y87.263 R8.201
027 N138 G1 Y107.063
028 N140 X42.964
029 N142 Y87.311
```

3. Ya con el código G se procese a editar la estructura de programación. Para ello hay que tener en cuenta ciertas reglas:

- Borrar los signos de porcentaje de inicio y fin de programa [%]
- Borrar encabezado de proyecto, se caracteriza por no tener códigos de maquinado CNC y no tener el descriptor de número de línea de programación además son de color verde.

```
003 (MCX FILE - C:\USERS\COMPUEDIT\DE
004 (NC FILE - C:\USERS\COMPUEDIT\DES
005 (MATERIAL - ALUMINUM MM - 2024)
006 ( T212 | 3. FLAT ENDMILL | H0 )
007 N100 G21
008 N102 G0 G17 G40 G49 G80 G90
009 N104 T212 M6
```

- Quitar parámetros de configuración de herramienta
T__ M6
- Quitar compensación por radio de curvatura.
G40
- Quitar porcentaje de velocidad de avance
G49
- Quitar compensación de longitud de herramienta
G43
- Quitar la anulación de ciclo finito
G80
- Quitar compensación de herramienta de corte.
H0
- Al final la estructura de encabezado para el maquinado en la maquina CNC queda de la siguiente manera.



```
001 N100 G21 G17 G90
002 N106 G0 G54 X7.071 Y112.929 S3500 M3
003 N108 Z50.
004 N110 Z10.
005 N112 G1 Z-5. F3.
```

En donde:

G21 = inicio de unidades métricas

G17 = trabajar en el plano XY

G90 = inicio de unidades absolutas

G00 = activa movimiento de avance rápido

G54 = guarda esta coordenada como punto de origen

4. Configuración recomendable de encabezado de código CNC para máquina Fresadora CNC.

- Seleccione el tipo de unidades en las que se va a trabajar (en MastercamX viene por defecto).

G20	Modo en pulgadas
G21	Modo en milímetros

- Seleccione el plano en el que va a trabajar (en MastercamX viene por defecto).

G17	Trabajo en plano XY
G18	Trabajo en plano XZ
G19	Trabajo en plano YZ

- Guarde la coordenada del cero pieza, (en MastercamX viene por defecto).

G54	X__ Y__ Z__
-----	-------------

- Seleccione el modo de sistema coordenadas a utilizar, (en MastercamX viene por defecto).

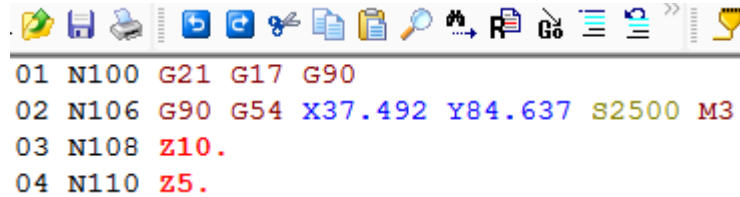
G90	Todas las cotas serán tomadas como absolutas.
G91	Todas las cotas serán tomadas como incrementales.

MANUAL DE USUARIO

Máquina Fresadora CNC

ANEXO 18

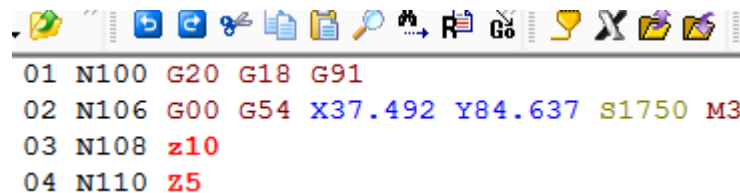
Ejemplos de encabezados de código G para máquina fresadora CNC Denford, a continuación se muestra ejemplos de secuencias de programación.



```
01 N100 G21 G17 G90
02 N106 G90 G54 X37.492 Y84.637 S2500 M3
03 N108 z10.
04 N110 z5.
```

[G21, G17, G90, G54 X__ Y__ Z__ S____ M3]

Se maquina con unidades en milímetros G21, en el plano XY G17, en coordenadas absolutas G90, y se tomara como cero pieza de trabajo la coordenada **G54 X__ Y__ Z__**, con una velocidad de rotación de husillo de S____ en RPM, en este momento encender husillo M3.



```
01 N100 G20 G18 G91
02 N106 G00 G54 X37.492 Y84.637 S1750 M3
03 N108 z10
04 N110 z5
```

[G20, G18, G91, G00, G54 X__ Y__ Z__ S____ M3]

Se maquina con unidades en pulgadas G20, en el plano XZ G18, en coordenadas incrementales G91, posicionarse a velocidad máxima G00 y se tomara como cero pieza de trabajo la coordenada **G54 X__ Y__ Z__**, el husillo rotará a velocidad en RPM de S____, en este momento encender husillo M3.

Sección 7

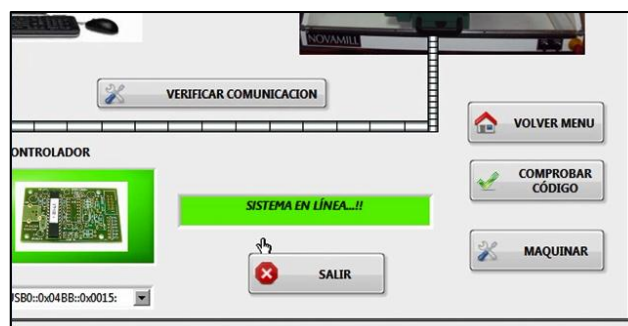
Ejemplo de maquinado CNC

Como ejemplo de maquinado CNC se procederá con una secuencia de pasos programados para que el usuario tenga facilidad de operación hasta adquirir experiencia en la elaboración de diseños de maquinado.

1. Obtener el código G.
2. Abrir el software instalado en la PC y ejecutar, con esta acción se presentara la ventana “LOGO”, en ella hay que hacer clic en el botón de “INICIAR”.



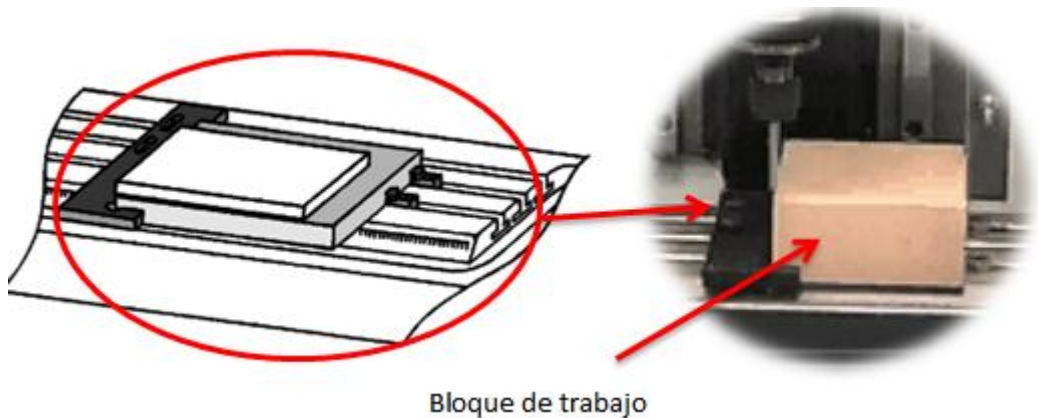
3. Verificamos comunicaciones y se debe visualizar el mensaje “SISTEMA EN LÍNEA...!!” y hacemos clic en “COMPROBAR CÓDIGO”, con lo cual se procederá a verificar y asegurarse que el código G a maquinar se encuentra libre de errores.



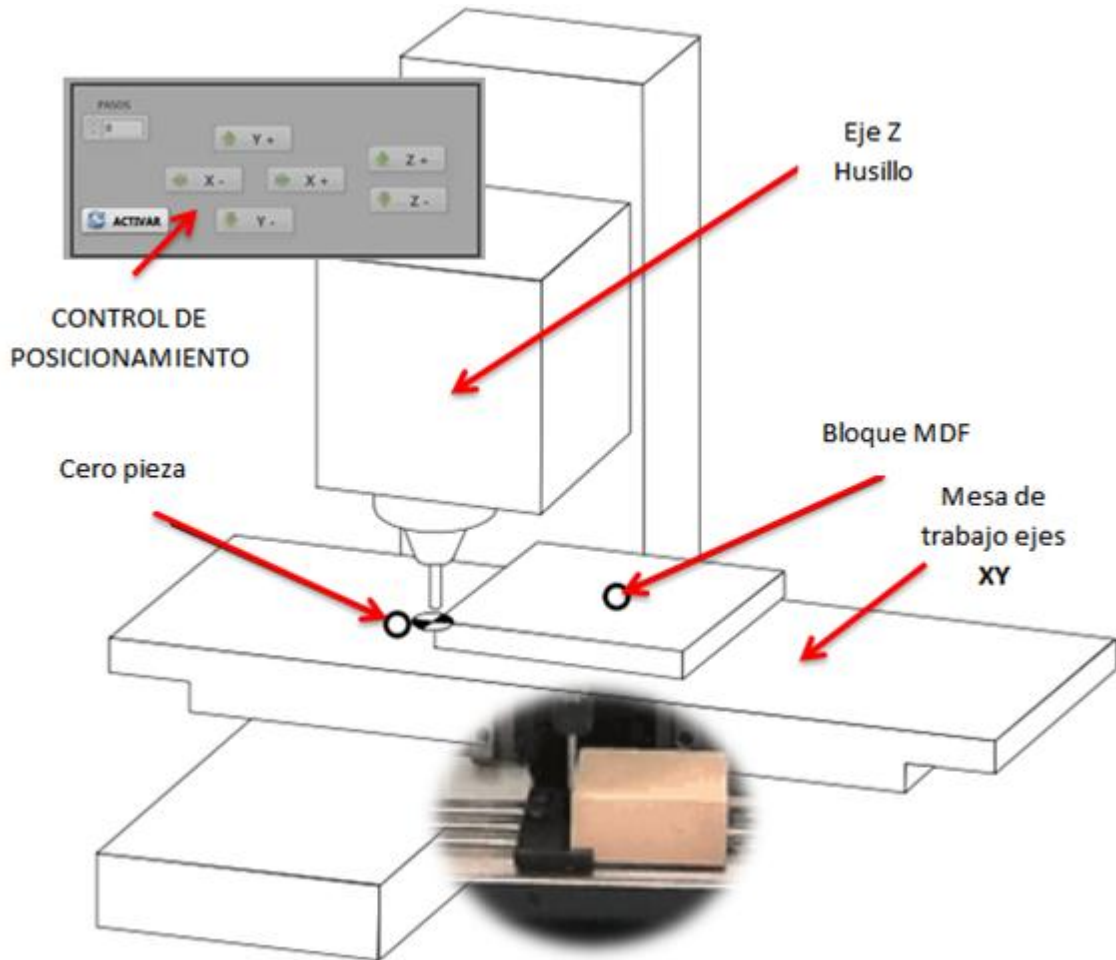
4. Ya en la ventana de verificación en “BUSCAR” se seleccionará el archivo de código G y luego se presionará el botón de “VERIFICAR”; para poder maquinar presione el botón “MAQUINAR” siempre u cuando no haya errores en la verificación.



5. Ya en la ventana de control para la máquina Fresadora CNC hay que colocar el bloque de madera a trabajar en la mesa de la máquina CNC y fijar con las terminales de ajuste. Nota: Utilizar herramientas adecuadas no improvisar para evitar daños en el equipo.



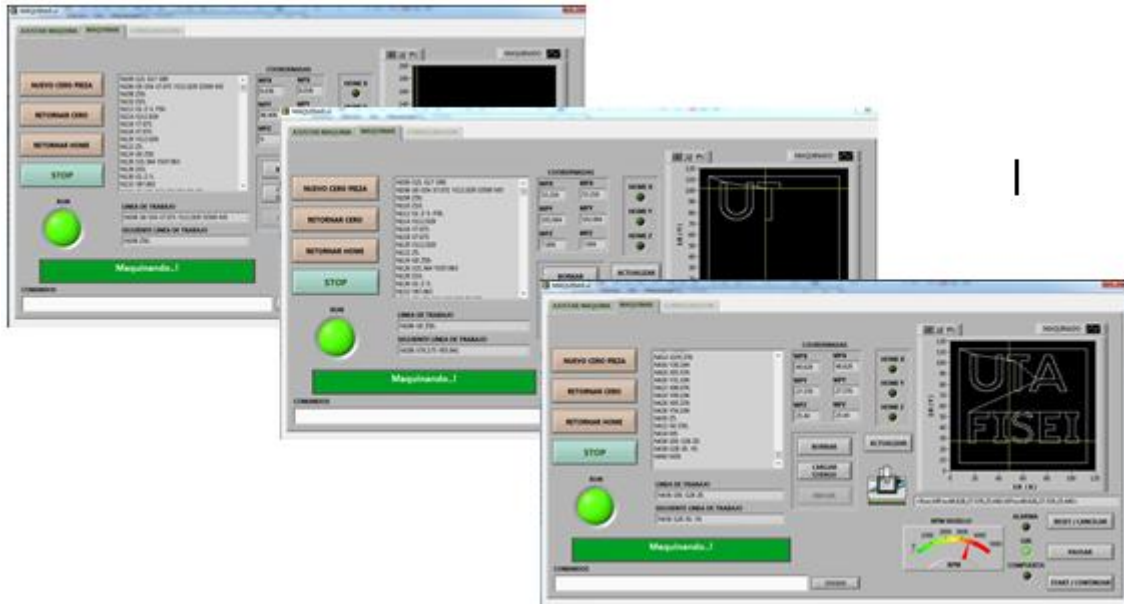
6. Ya ubicada y sujeta la pieza de trabajo hay que seleccionar el cero pieza y es la coordenada principal de la cual va a partir todas las trayectorias del maquinado. Para ello hay que presionar en la ventana del software “POSICIONAR MÁQUINA” los botones de desplazamiento según el caso hasta posicionar la herramienta de corte en el punto del cero pieza. Nota: Hay que tener mucho cuidado en este procedimiento revisando que no haya colisiones en los desplazamientos.



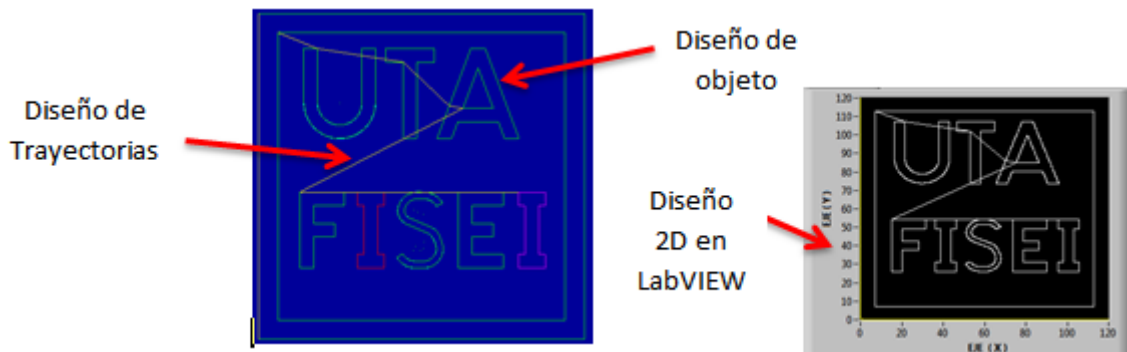
7. Ya ubicado el cero pieza se debe presionar en la ventana del software de control el botón “NUEVO CERO PIEZA” para guardar ésta coordenada. Ya guardada la coordenada hay que presionar el botón “RETORNAR HOME” y la máquina ésta lista para maquinar.



8. En este momento hay que presionar el botón “INICIAR”, siempre y cuando el código a maquinar ya éste previamente cargado y verificado. Ya en el proceso de maquinado solo hay que vigilar que no haya algún inconveniente. Nota: En la programación de maquinado hay que verificar que se utilicen la velocidad de avance y la velocidad de giro del husillo adecuadas para evitar colisiones.



9. Cuando el proceso de maquinado termine se obtendrá el bloque maquinado con el diseño.



Sección 8

Mantenimiento del equipo CNC

La máquina fresadora CNC debe estar siempre limpia, en cada proceso de maquinado se acumula viruta o material despendido producto del maquinado, para realizar labores de limpieza hay que tener precaución y por seguridad hay que presionar el paro de emergencia y si es posible apagarla, con una brocha hay que limpiar los rieles de los tres ejes de movimientos.



Una vez limpia la maquina hay que lubricar los rieles de los tres ejes para ello con un engrasador hay que inyectar aceite en los terminales de lubricación, por cada eje hay dos rieles cada uno de estos tiene una terminal de engrasado.



En el eje Y también hay dos rieles, éstos deben estar limpios y lubricados.



Cada uno de los ejes tiene un tornillo sin fin, este debe estar constantemente lubricado y bajo ningún motivo debe maquinarse en seco ya que se sobrecalentaría los motores.



ANEXO 19

Código de programación desarrollada en software WINAVR con
lenguaje C para controlador Arduino UNO R2

Programmer's Notepad - main.c

```
/*
PROYECTO:CONTROL MAQUINA CNC DENFORD
AUTOR DEL CODIGO: Copyright (c) 2012-2014 Simen Svale Skogsrud, Sungeun K. Jeon
LICENCIA DE CODIGO: LICENCIA LIBRE CODIGO ABIERTO
VERSION DEL CODIGO: GRBL V9.0
MODIFICACION: 14/09/2014 POR JESUS RODRIGUEZ
EMPRESA: UNIVERSIDAD TECNICA DE AMBATO
CONSULTA: <http://www.gnu.org/licenses/>.
*/

#include "system.h"
#include "serial.h"
#include "settings.h"
#include "protocol.h"
#include "gcode.h"
#include "planner.h"
#include "stepper.h"
#include "spindle_control.h"
#include "coolant_control.h"
#include "motion_control.h"
#include "limits.h"
#include "probe.h"
#include "report.h"

// Declarar sistema de estructura variable global
system_t sys;
int main(void)
{
    // sistema nitialize cuando se enciende.
    serial_init(); // Configuración de la velocidad de transmisión de serie y las
interrupciones
    settings_init(); // Cargar ajustes grbl de EEPROM
    stepper_init(); // Configurar pines paso a paso y contadores de tiempo de interrupción
    system_init(); // Configure pasadores de patillas y pin-cambio de interrupción
    memset(&sys, 0, sizeof(sys)); // Borrar todas las variables del sistema
    sys.abort = true; // Set abortar para completar la inicialización
    sei(); // Habilitar interrupciones

    /*Compruebe el encendido y programar una alarma del sistema si homing está
habilitada para forzar ciclo homing
Al establecer el estado de alarma de Grbl. Alarma bloquea todos los comandos de
código G, incluyendo la
Los scripts de inicio, pero permite el acceso a los ajustes y comandos internos.
Sólo un homing
Ciclo '$ H' o matar cerraduras de alarma '$ X' desactivará la alarma.
Nota: El script de inicio se ejecutará después de completar con éxito el ciclo de
homing, pero
No después de desactivar los bloqueos de alarma. Bloques de inicio impide el
movimiento de estrellarse en
Cosas incontrolablemente. Muy mal.*/
#ifdef HOMING_INIT_LOCK
    if (bit_istrue(settings.flags,BITFLAG_HOMING_ENABLE)) { sys.state = STATE_ALARM; }
#endif
    /*Grbl inicialización del bucle cuando se enciende o un aborto de sistema. Para este
último, todos los procesos
Volveré a este bucle para ser limpiamente re-inicializado.*/
    for(;;) {
        /* TODO: tarea de configuración independiente que requieren interrupciones que se
desactiven, especialmente sobre
un sistema de abortar y la garantía de las interrupciones activas se restablecen
limpiamente.
*/
        // Restablecer Grbl sistemas primarios
        serial_reset_read_buffer(); // Búfer de lectura en serie Claro
        gc_init(); // Set G-código analizador por defecto estado
        spindle_init();
    }
}
```

```
coolant_init();
limits_init();
probe_init();
plan_reset(); // Tampón bloque y planificador de variables claras
st_reset(); // Borrar las variables del subsistema de paso a paso.

// Sincronizar aclaró posiciones gcode y planificador a la posición actual del
sistema.
plan_sync_position();
gc_sync_position();

// Restablecer las variables del sistema.
sys.abort = false;
sys.execute = 0;
if (bit_istrue(settings.flags,BITFLAG_AUTO_START)) { sys.auto_start = true; }
else { sys.auto_start = false; }

// Inicia bucle principal Grbl. Procesos de insumos de los programas y los ejecuta.
protocol_main_loop();

}
return 0; /* Nunca alcanzado */
}
```

```
/*
PROYECTO:CONTROL MAQUINA CNC DENFORD
AUTOR DEL CODIGO: Copyright (c) 2012-2014 Simen Svale Skogsrud, Sungeun K. Jeon
LICENCIA DE CODIGO: LICENCIA LIBRE CODIGO ABIERTO
VERSION DEL CODIGO: GRBL V9.0
MODIFICACION: 14/09/2014 POR JESUS RODRIGUEZ
EMPRESA: UNIVERSIDAD TECNICA DE AMBATO
CONSULTA: <http://www.gnu.org/licenses/>.
*/

#ifndef system_h
#define system_h

// Definir los archivos de cabecera y bibliotecas del sistema estándar utilizados por Grbl
#include <avr/io.h>
#include <avr/pgmspace.h>
#include <avr/interrupt.h>
#include <avr/wdt.h>
#include <util/delay.h>
#include <math.h>
#include <inttypes.h>
#include <string.h>
#include <stdlib.h>
#include <stdint.h>
#include <stdbool.h>

// Definir configuración Grbl y archivos de cabecera compartidos
#include "config.h"
#include "defaults.h"
#include "cpu_map.h"
#include "nuts_bolts.h"

/*
// Definir ejecutor sistema de mapas de bits. Se usa internamente por protocolo de
tiempo de ejecución como banderas de comandos de tiempo de ejecución,
// Que notifica al programa principal para ejecutar el comando de ejecución
especificado de forma asíncronica.
// NOTA: El sistema ejecutor utiliza una variable volátil sin signo de 8 bits El valor
predeterminado (8 límite bandera.)
// Las banderas son siempre falsas, por lo que el protocolo de tiempo de ejecución sólo
tiene que comprobar un valor distinto de cero a
// Saber cuándo hay un comando de ejecución para ejecutar.
*/
#define EXEC_STATUS_REPORT bit(0) // máscara de bits 00000001
#define EXEC_CYCLE_START bit(1) // máscara de bits 00000010
#define EXEC_CYCLE_STOP bit(2) // máscara de bits 00000100
#define EXEC_FEED_HOLD bit(3) // máscara de bits 00001000
#define EXEC_RESET bit(4) // máscara de bits 00010000
#define EXEC_ALARM bit(5) // máscara de bits 00100000
#define EXEC_CRIT_EVENT bit(6) // máscara de bits 01000000
// #define bit(7) // máscara de bits 10000000
*/
/// Definir sistema de mapa de bits de estado. La variable de estado almacena
principalmente las funciones individuales
// De Grbl para gestionar cada sin superponerse. También se usa como una bandera de
mensajería para
// Los eventos críticos. */

#define STATE_IDLE 0 // Debe ser cero. No hay banderas.
#define STATE_ALARM bit(0) // En estado de alarma. Bloquea todos los procesos de
g-código. Permite el acceso ajustes.
#define STATE_CHECK_MODE bit(1) // Modo de verificación de código G. Bloquea
planificador y sólo movimiento.
#define STATE_HOMING bit(2) // Realizar ciclo homing
#define STATE_QUEUED bit(3) // Indica bloques almacenados temporalmente, a la
espera del inicio del ciclo.
#define STATE_CYCLE bit(4) // Ciclo está ejecutando
#define STATE_HOLD bit(5) // Ejecución de retención de alimentación
// #define STATE_JOG bit(6) // Modo de jogging es único como mensajera.

// Definir las variables globales del sistema
typedef struct {
uint8_t abort; // Sistema abortar bandera. Fuerzas salir de nuevo a
bucle principal para el rearme.

```

```
uint8_t state; // Rastrea el estado actual de Grbl.
volatile uint8_t execute; // Global ejecutor tiempo del sistema bitflag
variable. Ver máscaras de bits EXEC.
uint8_t homing_axis_lock;
int32_t position[N_AXIS]; // Máquina de tiempo real (aka casa) vector de
posición en los pasos. // NOTA: Esto puede tener que ser una variable
volátil, si surgen problemas.
uint8_t auto_start; // Planificador de la bandera de inicio automático.
Desactivados durante bodega alimentación. Fallido por los ajustes.
volatile uint8_t probe_state; // Sondeo valor de estado. Se utiliza para coordinar
el ciclo de palpación con ISR paso a paso.
int32_t probe_position[N_AXIS]; // Última posición de la sonda en coordenadas de
máquina y pasos.
} system_t;
extern system_t sys;

// Inicializar el protocolo serie
void system_init();

// Ejecuta un comando del sistema interno, que se define como una cadena que comienza
con un "$"
uint8_t system_execute_line(char *line);

// Los cheques y ejecuta una orden de ejecución en varios puntos de parada en el
programa principal
void system_execute_runtime();

// Ejecutar las líneas de script de inicio almacenado en la EEPROM en la inicialización
void system_execute_startup(char *line);

#endif
```



```
/*
PROYECTO:CONTROL MAQUINA CNC DENFORD
AUTOR DEL CODIGO: Copyright (c) 2012-2014 Simen Svale Skogsrud, Sungeun K. Jeon
LICENCIA DE CODIGO: LICENCIA LIBRE CODIGO ABIERTO
VERSION DEL CODIGO: GRBL V9.0
MODIFICACION: 14/09/2014 POR JESUS RODRIGUEZ
EMPRESA: UNIVERSIDAD TECNICA DE AMBATO
CONSULTA: <http://www.gnu.org/licenses/>.
*/

#include "system.h"
#include "settings.h"
#include "eeprom.h"
#include "protocol.h"
#include "report.h"
#include "limits.h"
#include "stepper.h"

settings_t settings;

// Método para almacenar líneas de arranque en EEPROM
void settings_store_startup_line(uint8_t n, char *line)
{
    uint32_t addr = n*(LINE_BUFFER_SIZE+1)+EEPROM_ADDR_STARTUP_BLOCK;
    memcpy_to_eeprom_with_checksum(addr,(char*)line, LINE_BUFFER_SIZE);
}

// Método para almacenar construir información en la EEPROM
void settings_store_build_info(char *line)
{
    memcpy_to_eeprom_with_checksum(EEPROM_ADDR_BUILD_INFO,(char*)line, LINE_BUFFER_SIZE);
}

// Método para almacenar parámetros de datos en la EEPROM coord
void settings_write_coord_data(uint8_t coord_select, float *coord_data)
{
    uint32_t addr = coord_select*(sizeof(float)*N_AXIS+1) + EEPROM_ADDR_PARAMETERS;
    memcpy_to_eeprom_with_checksum(addr,(char*)coord_data, sizeof(float)*N_AXIS);
}

// Método para almacenar Grbl configuración global estructura y número de versión en la EEPROM
void write_global_settings()
{
    eeprom_put_char(0, SETTINGS_VERSION);
    memcpy_to_eeprom_with_checksum(EEPROM_ADDR_GLOBAL, (char*)&settings, sizeof(settings_t)
);
}

// Método para restaurar Grbl configuración global EEPROM-salvado a los valores predeterminados.
void settings_restore_global_settings() {
    settings.pulse_microseconds = DEFAULT_STEP_PULSE_MICROSECONDS;
    settings.stepper_idle_lock_time = DEFAULT_STEPPER_IDLE_LOCK_TIME;
    settings.step_invert_mask = DEFAULT_STEPPING_INVERT_MASK;
    settings.dir_invert_mask = DEFAULT_DIRECTION_INVERT_MASK;
    settings.status_report_mask = DEFAULT_STATUS_REPORT_MASK;
    settings.junction_deviation = DEFAULT_JUNCTION_DEVIATION;
    settings.arc_tolerance = DEFAULT_ARC_TOLERANCE;
    settings.homing_dir_mask = DEFAULT_HOMING_DIR_MASK;
    settings.homing_feed_rate = DEFAULT_HOMING_FEED_RATE;
    settings.homing_seek_rate = DEFAULT_HOMING_SEEK_RATE;
    settings.homing_debounce_delay = DEFAULT_HOMING_DEBOUNCE_DELAY;
    settings.homing_pulloff = DEFAULT_HOMING_PULLOFF;

    settings.flags = 0;
    if (DEFAULT_REPORT_INCHES) { settings.flags |= BITFLAG_REPORT_INCHES; }
    if (DEFAULT_AUTO_START) { settings.flags |= BITFLAG_AUTO_START; }
    if (DEFAULT_INVERT_ST_ENABLE) { settings.flags |= BITFLAG_INVERT_ST_ENABLE; }
    if (DEFAULT_INVERT_LIMIT_PINS) { settings.flags |= BITFLAG_INVERT_LIMIT_PINS; }
    if (DEFAULT_SOFT_LIMIT_ENABLE) { settings.flags |= BITFLAG_SOFT_LIMIT_ENABLE; }
```

```
if (DEFAULT_HARD_LIMIT_ENABLE) { settings.flags |= BITFLAG_HARD_LIMIT_ENABLE; }
if (DEFAULT_HOMING_ENABLE) { settings.flags |= BITFLAG_HOMING_ENABLE; }

settings.steps_per_mm[X_AXIS] = DEFAULT_X_STEPS_PER_MM;
settings.steps_per_mm[Y_AXIS] = DEFAULT_Y_STEPS_PER_MM;
settings.steps_per_mm[Z_AXIS] = DEFAULT_Z_STEPS_PER_MM;
settings.max_rate[X_AXIS] = DEFAULT_X_MAX_RATE;
settings.max_rate[Y_AXIS] = DEFAULT_Y_MAX_RATE;
settings.max_rate[Z_AXIS] = DEFAULT_Z_MAX_RATE;
settings.acceleration[X_AXIS] = DEFAULT_X_ACCELERATION;
settings.acceleration[Y_AXIS] = DEFAULT_Y_ACCELERATION;
settings.acceleration[Z_AXIS] = DEFAULT_Z_ACCELERATION;
settings.max_travel[X_AXIS] = (-DEFAULT_X_MAX_TRAVEL);
settings.max_travel[Y_AXIS] = (-DEFAULT_Y_MAX_TRAVEL);
settings.max_travel[Z_AXIS] = (-DEFAULT_Z_MAX_TRAVEL);

write_global_settings();
}

// Función de ayuda a despejar el espacio EEPROM contiene datos de parámetros.
void settings_clear_parameters() {
    uint8_t idx;
    float coord_data[3];
    memset(&coord_data, 0, sizeof(coord_data));
    for (idx=0; idx < SETTING_INDEX_NCOORD; idx++) { settings_write_coord_data(idx,
coord_data); }
}

// Función de ayuda a despejar el espacio EEPROM que contiene las líneas de inicio.
void settings_clear_startup_lines() {
    #if N_STARTUP_LINE > 0
    eeprom_put_char(EEPROM_ADDR_STARTUP_BLOCK, 0);
    #endif
    #if N_STARTUP_LINE > 1
    eeprom_put_char(EEPROM_ADDR_STARTUP_BLOCK+(LINE_BUFFER_SIZE+1), 0);
    #endif
}

// Función de ayuda a despejar el espacio EEPROM que contiene la información de
construcción cadena de usuario.
void settings_clear_build_info() { eeprom_put_char(EEPROM_ADDR_BUILD_INFO , 0); }

// Lee la línea de arranque de la EEPROM. Actualizado datos de cadena de línea apuntada.
uint8_t settings_read_startup_line(uint8_t n, char *line)
{
    uint32_t addr = n*(LINE_BUFFER_SIZE+1)+EEPROM_ADDR_STARTUP_BLOCK;
    if (!(memcpy_from_eeprom_with_checksum((char*)line, addr, LINE_BUFFER_SIZE))) {
        // Restablecer línea con el valor por defecto
        line[0] = 0; // línea vacía
        settings_store_startup_line(n, line);
        return(false);
    }
    return(true);
}

// Lee la línea de arranque de la EEPROM. Actualizado datos de cadena de línea apuntada.
uint8_t settings_read_build_info(char *line)
{
    if (!(memcpy_from_eeprom_with_checksum((char*)line, EEPROM_ADDR_BUILD_INFO,
LINE_BUFFER_SIZE))) {
        // Restablecer línea con el valor por defecto
        line[0] = 0; // línea vacía
        settings_store_build_info(line);
        return(false);
    }
    return(true);
}

// Leer coordenadas seleccionado datos de la EEPROM. Actualizaciones señaló valor
coord_data.
```

```

uint8_t settings_read_coord_data(uint8_t coord_select, float *coord_data)
{
    uint32_t addr = coord_select*(sizeof(float)*N_AXIS+1) + EEPROM_ADDR_PARAMETERS;
    if (!memcpy_from_eeprom_with_checksum((char*)coord_data, addr, sizeof(float)*N_AXIS
))) {
        // Restablecer con defecto vector cero
        clear_vector_float(coord_data);
        settings_write_coord_data(coord_select, coord_data);
        return(false);
    }
    return(true);
}

// Lee Grbl configuración global estructura de la EEPROM.
uint8_t read_global_settings() {
    // Compruebe la versión de bytes de EEPROM
    uint8_t version = eeprom_get_char(0);
    if (version == SETTINGS_VERSION) {
        // Leer configuración a grabar y verificar la suma de comprobación
        if (!memcpy_from_eeprom_with_checksum((char*)&settings, EEPROM_ADDR_GLOBAL, sizeof(
settings_t)))) {
            return(false);
        }
    } else {
        return(false);
    }
    return(true);
}

// Un método de ayuda para establecer la configuración de línea de comandos
uint8_t settings_store_global_setting(uint8_t parameter, float value) {
    if (value < 0.0) { return(STATUS_NEGATIVE_VALUE); }
    if (parameter >= AXIS_SETTINGS_START_VAL) {
        // Configuración de ejes tienda. Secuencia Eje numeración establecido por
        // AXIS_SETTING define.
        // NOTA: Asegúrese de que el índice de ajuste corresponde a la impresión ajustes
        // report.c.
        parameter -= AXIS_SETTINGS_START_VAL;
        uint8_t set_idx = 0;
        while (set_idx < AXIS_N_SETTINGS) {
            if (parameter < N_AXIS) {
                // Eje Configuración válida encontraron.
                switch (set_idx) {
                    case 0: settings.steps_per_mm[parameter] = value; break;
                    case 1: settings.max_rate[parameter] = value; break;
                    case 2: settings.acceleration[parameter] = value*60*60; break; // Convertir
                    // en mm / min ^ 2 para uso interno grbl.
                    case 3: settings.max_travel[parameter] = -value; break; // Guarde como
                    // negativas para uso interno grbl.
                }
                break; // Salir mientras bucle después de ajuste se ha configurado y procederá
                // a la convocatoria de EEPROM.
            } else {
                set_idx++;
                // Si eje índice mayor que N_AXIS o la creación de índice mayor que el número
                // de configuración del eje, error fuera.
                if ((parameter < AXIS_SETTINGS_INCREMENT) || (set_idx == AXIS_N_SETTINGS)) {
                    return(STATUS_INVALID_STATEMENT);
                }
                parameter -= AXIS_SETTINGS_INCREMENT;
            }
        }
    } else {
        // Guarde los ajustes no-eje Grbl
        uint8_t int_value = trunc(value);
        switch(parameter) {
            case 0:
                if (int_value < 3) { return(STATUS_SETTING_STEP_PULSE_MIN); }
                settings.pulse_microseconds = int_value; break;
            case 1: settings.stepper_idle_lock_time = int_value; break;
            case 2:
                settings.step_invert_mask = int_value;
                st_generate_step_dir_invert_masks(); // Regenerar paso y portuarias dirección
                // máscaras invertidas.
                break;
        }
    }
}

```

```

    case 3:
        settings.dir_invert_mask = int_value;
        st_generate_step_dir_invert_masks(); // Regenerar paso y portuarias dirección
máscaras invertidas.
        break;
    case 4: // Restablecer para asegurar el cambio. Inmediata re-init puede causar
problemas.
        if (int_value) { settings.flags |= BITFLAG_INVERT_ST_ENABLE; }
        else { settings.flags &= ~BITFLAG_INVERT_ST_ENABLE; }
        break;
    case 5: // Restablecer para asegurar el cambio. Inmediata re-init puede causar
problemas.
        if (int_value) { settings.flags |= BITFLAG_INVERT_LIMIT_PINS; }
        else { settings.flags &= ~BITFLAG_INVERT_LIMIT_PINS; }
        break;
    case 6: // Restablecer para asegurar el cambio. Inmediata re-init puede causar
problemas.
        if (int_value) { settings.flags |= BITFLAG_INVERT_PROBE_PIN; }
        else { settings.flags &= ~BITFLAG_INVERT_PROBE_PIN; }
        break;
    case 10: settings.status_report_mask = int_value; break;
    case 11: settings.junction_deviation = value; break;
    case 12: settings.arc_tolerance = value; break;
    case 13:
        if (int_value) { settings.flags |= BITFLAG_REPORT_INCHES; }
        else { settings.flags &= ~BITFLAG_REPORT_INCHES; }
        break;
    case 14: // Restablecer para asegurar el cambio. Inmediata re-init puede causar
problemas.
        if (int_value) { settings.flags |= BITFLAG_AUTO_START; }
        else { settings.flags &= ~BITFLAG_AUTO_START; }
        break;
    case 20:
        if (int_value) {
            if (bit_isfalse(settings.flags, BITFLAG_HOMING_ENABLE)) { return(
STATUS_SOFT_LIMIT_ERROR); }
            settings.flags |= BITFLAG_SOFT_LIMIT_ENABLE;
        } else { settings.flags &= ~BITFLAG_SOFT_LIMIT_ENABLE; }
        break;
    case 21:
        if (int_value) { settings.flags |= BITFLAG_HARD_LIMIT_ENABLE; }
        else { settings.flags &= ~BITFLAG_HARD_LIMIT_ENABLE; }
        limits_init(); // Re-init para cambiar inmediatamente NOTA :: Es agradable
tener pero podría ser problemático más tarde.
        break;
    case 22:
        if (int_value) { settings.flags |= BITFLAG_HOMING_ENABLE; }
        else {
            settings.flags &= ~BITFLAG_HOMING_ENABLE;
            settings.flags &= ~BITFLAG_SOFT_LIMIT_ENABLE; // Fuerza desactivar plazos
suaves.
        }
        break;
    case 23: settings.homing_dir_mask = int_value; break;
    case 24: settings.homing_feed_rate = value; break;
    case 25: settings.homing_seek_rate = value; break;
    case 26: settings.homing_debounce_delay = int_value; break;
    case 27: settings.homing_pulloff = value; break;
    default:
        return(STATUS_INVALID_STATEMENT);
    }
}
write_global_settings();
return(STATUS_OK);
}

```

// Inicializar el subsistema de configuración

```

void settings_init() {
    if(!read_global_settings()) {
        report_status_message(STATUS_SETTING_READ_FAIL);

        settings_restore_global_settings();
    }
}

```

// Forzar líneas claras de inicio y construir datos información de los usuarios.
Los parámetros deben estar bien.

```
// TODO: Para la próxima versión, eliminar estos borra sólo aquí porque buffer de
línea aumentó ..
settings_clear_startup_lines();
settings_clear_build_info();

report_grbl_settings();
}

// Revise todos los datos de los parámetros en una variable ficticia. Si el error,
pone a cero, de lo contrario no hacer nada.
float coord_data[N_AXIS];
uint8_t i;
for (i=0; i<=SETTING_INDEX_NCOORD; i++) {
    if (!settings_read_coord_data(i, coord_data)) {
        report_status_message(STATUS_SETTING_READ_FAIL);
    }
}
//NOTA: Las líneas de inicio están controlados y ejecutados por protocol_main_loop al
final de la inicialización.
//TODO :: Build información debe comprobarse aquí, pero esperará hasta v1.0 para
abordar esta bien por ahora.
}

// Devoluciones paso máscara pin según Grbl eje interno de indexación.
uint8_t get_step_pin_mask(uint8_t axis_idx)
{
    if ( axis_idx == X_AXIS ) { return((1<<X_STEP_BIT)); }
    if ( axis_idx == Y_AXIS ) { return((1<<Y_STEP_BIT)); }
    return((1<<Z_STEP_BIT));
}

// Devuelve máscara dirección pin según Grbl eje interno de indexación.
uint8_t get_direction_pin_mask(uint8_t axis_idx)
{
    if ( axis_idx == X_AXIS ) { return((1<<X_DIRECTION_BIT)); }
    if ( axis_idx == Y_AXIS ) { return((1<<Y_DIRECTION_BIT)); }
    return((1<<Z_DIRECTION_BIT));
}

// Devoluciones limitan máscara pin según Grbl eje interno de indexación.
uint8_t get_limit_pin_mask(uint8_t axis_idx)
{
    if ( axis_idx == X_AXIS ) { return((1<<X_LIMIT_BIT)); }
    if ( axis_idx == Y_AXIS ) { return((1<<Y_LIMIT_BIT)); }
    return((1<<Z_LIMIT_BIT));
}
```

```
/*
PROYECTO:CONTROL MAQUINA CNC DENFORD
AUTOR DEL CODIGO: Copyright (c) 2012-2014 Simen Svale Skogsrud, Sungeun K. Jeon
LICENCIA DE CODIGO: LICENCIA LIBRE CODIGO ABIERTO
VERSION DEL CODIGO: GRBL V9.0
MODIFICACION: 14/09/2014 POR JESUS RODRIGUEZ
EMPRESA: UNIVERSIDAD TECNICA DE AMBATO
CONSULTA: <http://www.gnu.org/licenses/>.
*/

#include <avr/interrupt.h>
#include "system.h"
#include "serial.h"
#include "motion_control.h"
#include "protocol.h"

uint8_t serial_rx_buffer[RX_BUFFER_SIZE];
uint8_t serial_rx_buffer_head = 0;
volatile uint8_t serial_rx_buffer_tail = 0;

uint8_t serial_tx_buffer[TX_BUFFER_SIZE];
uint8_t serial_tx_buffer_head = 0;
volatile uint8_t serial_tx_buffer_tail = 0;

#ifdef ENABLE_XONXOFF
    volatile uint8_t flow_ctrl = XON_SENT; // Flujo variable de estado de control
#endif

// Devuelve el número de bytes utilizados en la memoria intermedia serie RX.
uint8_t serial_get_rx_buffer_count()
{
    uint8_t rtail = serial_rx_buffer_tail; // Copia para limitar varias llamadas a volátil
    if (serial_rx_buffer_head >= rtail) { return(serial_rx_buffer_head-rtail); }
    return (RX_BUFFER_SIZE - (rtail-serial_rx_buffer_head));
}

// Devuelve el número de bytes utilizados en la memoria intermedia serie TX.
// NOTA: No se usa, excepto para la depuración y la garantía de no hay cuellos de
botella TX.

uint8_t serial_get_tx_buffer_count()
{
    uint8_t ttail = serial_tx_buffer_tail; // Copy to limit multiple calls to volatile
    if (serial_tx_buffer_head >= ttail) { return(serial_tx_buffer_head-ttail); }
    return (TX_BUFFER_SIZE - (ttail-serial_tx_buffer_head));
}

void serial_init()
{
    // Establezca la velocidad de transmisión
    #if BAUD_RATE < 57600
        uint16_t UBRR0_value = ((F_CPU / (8L * BAUD_RATE)) - 1)/2 ;
        UCSR0A &= ~(1 << U2X0); // baudios doblador off - Sólo necesitaba sobre el Uno XXX
    #else
        uint16_t UBRR0_value = ((F_CPU / (4L * BAUD_RATE)) - 1)/2;
        UCSR0A |= (1 << U2X0); // baudios doblador por altas velocidades de transmisión,
es decir, 115.200
    #endif
    UBRR0H = UBRR0_value >> 8;
    UBRR0L = UBRR0_value;

    // permitir rx y tx
    UCSR0B |= 1<<RXEN0;
    UCSR0B |= 1<<TXEN0;

    // habilitar interrupción en la recepción completa de un byte
    UCSR0B |= 1<<RXCIE0;

    // por defecto es de 8 bits, sin paridad, 1 bit de parada
}
```

```
// Escribe un byte en la memoria intermedia serie TX. Llamado por programa principal.
// TODO: Compruebe si podemos acelerar este proceso para escribir cadenas, en lugar de
bytes individuales.

void serial_write(uint8_t data) {
    // Calcular próximo jefe
    uint8_t next_head = serial_tx_buffer_head + 1;
    if (next_head == TX_BUFFER_SIZE) { next_head = 0; }

    // Espere hasta que haya espacio en la memoria intermedia
    while (next_head == serial_tx_buffer_tail) {
        // TODO: Restructure st_prep_buffer() llamadas a ser ejecutado aquí durante una
        impresión larga.
        if (sys.execute & EXEC_RESET) { return; } // sólo comprobar abortar para evitar un
        bucle sin fin.
    }

    // Almacenar datos y la cabeza antelación
    serial_tx_buffer[serial_tx_buffer_head] = data;
    serial_tx_buffer_head = next_head;

    // Habilitar registro de datos vacía de interrupción para asegurarse tx-transmisión
    está en marcha
    UCSR0B |= (1 << UDRIE0);
}

// Registro de datos vacía manejador de interrupción
ISR(SERIAL_UDRE)
{
    uint8_t tail = serial_tx_buffer_tail; // Serial_tx_buffer_tail Temporal (para
    optimizar volátil)

    #ifdef ENABLE_XONXOFF
        if (flow_ctrl == SEND_XOFF) {
            UDR0 = XOFF_CHAR;
            flow_ctrl = XOFF_SENT;
        } else if (flow_ctrl == SEND_XON) {
            UDR0 = XON_CHAR;
            flow_ctrl = XON_SENT;
        } else
    #endif
    {
        // Enviar un byte de la memoria intermedia
        UDR0 = serial_tx_buffer[tail];

        // Actualización de posición de la cola
        tail++;
        if (tail == TX_BUFFER_SIZE) { tail = 0; }

        serial_tx_buffer_tail = tail;
    }

    // Apague el registro de datos vacía de interrupción para detener tx-streaming de si
    esto concluye la transferencia
    if (tail == serial_tx_buffer_head) { UCSR0B &= ~(1 << UDRIE0); }
}

// Obtiene el primer byte en el búfer de lectura en serie. Llamado por programa
principal.
uint8_t serial_read()
{
    uint8_t tail = serial_rx_buffer_tail; // Serial_rx_buffer_tail Temporal (para
    optimizar volátil)
    if (serial_rx_buffer_head == tail) {
        return SERIAL_NO_DATA;
    } else {
        uint8_t data = serial_rx_buffer[tail];

        tail++;
        if (tail == RX_BUFFER_SIZE) { tail = 0; }
        serial_rx_buffer_tail = tail;
    }

    #ifdef ENABLE_XONXOFF
```

```

    if ((serial_get_rx_buffer_count() < RX_BUFFER_LOW) && flow_ctrl == XOFF_SENT) {
        flow_ctrl = SEND_XON;
        UCSR0B |= (1 << UDRIE0); // Fuerza TX
    }
}
#endif

return data;
}
}

ISR(SERIAL_RX)
{
    uint8_t data = UDR0;
    uint8_t next_head;

    /*Recogida de caracteres de comandos de tiempo de ejecución directa de la corriente
de serie. Estos personajes son
No pasó en el búfer, pero estos bits de la bandera del estado del sistema conjunto
para su ejecución en tiempo de ejecución.*/
    switch (data) {
        case CMD_STATUS_REPORT: bit_true_atomic(sys.execute, EXEC_STATUS_REPORT); break; //
Establecer como verdadera
        case CMD_CYCLE_START: bit_true_atomic(sys.execute, EXEC_CYCLE_START); break; //
Establecer como verdadera
        case CMD_FEED_HOLD: bit_true_atomic(sys.execute, EXEC_FEED_HOLD); break; //
Establecer como verdadera
        case CMD_RESET: mc_reset(); break; // Reajuste del control de movimiento de
llamada.
        default: // Escribe carácter para amortiguar
            next_head = serial_rx_buffer_head + 1;
            if (next_head == RX_BUFFER_SIZE) { next_head = 0; }

            // Escribir datos para amortiguar a menos que esté lleno.
            if (next_head != serial_rx_buffer_tail) {
                serial_rx_buffer[serial_rx_buffer_head] = data;
                serial_rx_buffer_head = next_head;

                #ifdef ENABLE_XONXOFF
                    if ((serial_get_rx_buffer_count() >= RX_BUFFER_FULL) && flow_ctrl == XON_SENT)
                {
                    flow_ctrl = SEND_XOFF;
                    UCSR0B |= (1 << UDRIE0); // Fuerza TX
                }
            }
            #endif

            }
        //TODO: otra alarma en caso de desbordamiento?
    }
}

void serial_reset_read_buffer()
{
    serial_rx_buffer_tail = serial_rx_buffer_head;

    #ifdef ENABLE_XONXOFF
        flow_ctrl = XON_SENT;
    #endif
}

```


ANEXO 20

Código de programación para microcontrolador PIC 18F2550 desarrollado en
PIC C Compiler con lenguaje C

```
C:\Users\Compuedit\Desktop\USB_HID\Codigo\USBHID.c1: #include
<18F2550.h>
2: #fuses HSPLL, NOWDT, NOPROTECT, NOLVP, NODEBUG, USBDIV, PLL5, CPUDIV1, VREGEN3: #use
delay(clock=48M)
4: #DEFINE USB_HID_DEVICE TRUE
5: //Encender EP1 para IN bulk / transferencias de interrupción
6: #define USB_EP1_TX_ENABLE USB_ENABLE_INTERRUPT
7: #define USB_EP1_TX_SIZE 8
8: //encender EP1 para OUT BULK transferencias de interrupción
9: #define USB_EP1_RX_ENABLE USB_ENABLE_INTERRUPT
10: #define USB_EP1_RX_SIZE 8
11: #include <pic18_usb.h>
12: // funciones de bajo nivel(hardware) para la serie
13: // IC 18Fxx5x que servirán en usb.c
14: #include <usb_desc_hid.h>
15: // ^qui es donde van las descripciones de
16: // este dispositivo (como lo reconocerá windows)
17: #include <usb.c>
18: // librería para el manejo del usb
19: #define OFF output_low
20: #define ON output_high
21: // CONFIGURACION DE PUERTO DIGITALES DE SALIDA
22: #define HOME_OUT_X PIN_B0
23: #define HOME_OUT_Y PIN_B1
24: #define HOME_OUT_Z PIN_B2
25: #define RESET_ABORT PIN_B3
26: #define FEED_HOLD PIN_B4
27: #define START_RESUME PIN_B5
28: // CONFIGURACION DE PUERTO DIGITAL DE ENTRADA
29: #define HOME_IN_X PIN_A1
30: #define HOME_IN_Y PIN_A2
31: #define HOME_IN_Z PIN_A3
32: #define HOME_IN_W PIN_A4
33: #define HOME_IN_V PIN_A5
34: #define CLOSE PIN_C0
35: //DECLARACION DE VARIABLES
36: int8 Salida[8], Entrada[8];
37: INT b,c,d,e,f;
38: void main() {
39: //CONFIGURACION CANALES ANALOGOS
40: setup_adc_ports(AN0);
41: setup_adc(ADC_CLOCK_INTERNAL);
42: //CONFIGURACION MODO PWM
43: //setup_ccp1(ccp_pwm);
44: //setup_timer_2(t2_div_by_4, 249, 1);
45: set_tris_b(0x00);
46: output_b(0x00);
47: //LED_ON(LED_R);
48: //LED_OFF(LED_V);
49: usb_init();
50: //Monitorea el estado de la conexión conectándose y
51: // desconectándose automáticamente
52: usb_task();
53: //espera infinitamente hasta que el
54: //dispositivo fue enumerado
55: usb_wait_for_enumeration();
56: output_b(LED_V);
57: //LED_OFF(LED_R);
58: while (TRUE)
59: {
60: usb_task();
61: if (usb_enumerated())
62: {
63: //CONSTRUCCION CANALES ANALOGOS
64: set_adc_channel(0);
65: Salida[0]=read_adc();
66: //INVERSION DE PINES DIGITALES
67: //HOME_X, Y, Z START
68: if (input(HOME_IN_X)==1)
```

```

C:\Users\Compuedit\Desktop\USB_HID\Codigo\USBHID.c
69: {OFF(HOME_OUT_X);a=0;}else{ON(HOME_OUT_X);a=1;}
70: if (input(HOME_IN_Y)==1)
71: {OFF(HOME_OUT_Y);b=0;}else{ON(HOME_OUT_Y);b=1;}
72: if (input(HOME_IN_Z)==1)
73: {OFF(HOME_OUT_Z);c=0;}else{ON(HOME_OUT_Z);c=1;}
74: if (input(STOP)==1)
75: {d=0;}else{d=1;}
76: if (input(GIR)==1)
77: {e=0;}else{e=1;}
78: if (input(CLOSE)==1)
79: {f=0;}else{f=1;}
80: Salida[1]=a+2*b+4*c+8*d+16*e+32*f;
81: //ENVIO DE PAQUETE DE DATOS USB
82: usb_put_packet(1, Salida, 2, USB_DTS_TOGGLE);
83: //RESECCION DE DATOS POR PUERTO USB
84: if (usb_kbhit(1))
85: {
86:
87: usb_get_packet(1, Entrada, 1);
88:
89: switch (Entrada[0])
90: {
91: //ojo con off enciendo el pin
92: case 0:
93: ON(RESET_ABORT);
94: ON(FEED_HOLD);
95: ON(START_RESUME);
96: break;
97: case 1:
98: OFF(RESET_ABORT);
99: ON(FEED_HOLD);
100: ON(START_RESUME);
101: break;
102: case 2:
103: ON(RESET_ABORT);
104: OFF(FEED_HOLD);
105: ON(START_RESUME);
106: break;
107: case 3:
108: OFF(RESET_ABORT);
109: OFF(FEED_HOLD);
110: ON(START_RESUME);
111: break;
112: case 4:
113: ON(RESET_ABORT);
114: ON(FEED_HOLD);
115: OFF(START_RESUME);
116: break;
117: case 5:
118: OFF(RESET_ABORT);
119: ON(FEED_HOLD);
120: OFF(START_RESUME);
121: break;
122: case 6:
123: ON(RESET_ABORT);
124: OFF(FEED_HOLD);
125: OFF(START_RESUME);
126: break;
127: case 7:
128: OFF(RESET_ABORT);
129: OFF(FEED_HOLD);
130: OFF(START_RESUME);
131: break;
132: default:
133: ON(RESET_ABORT); 138: }//FIN RECEPCION DE DATOS
134: ON(FEED_HOLD); 139: }// FIN ENUMERACION USB
135: ON(START_RESUME);140: }// FIN WHILE TRUE
136: break; 141: }// FIN MAIN

```

ANEXO 21

Código de programación PIC 16F628A programado en MICROCODE
para el control de motores paso a paso

```
C:\Users\Compuedit\Desktop\code_16F628A_enable_motor\prueba_portb\main.c
1: #include <main.h>
2: long tiempo=0;
3: #int_RB
4: void RB_isr(void)
5: {
6:     disable_interrupts(INT_RB);
7:     output_high(pin_a1); //activar enable
8:     tiempo=0;
9: }
10: void main()
11: {
12:     while(true){
13:         set_tris_a(0x00);
14:         set_tris_b(0xff);
15:         enable_interrupts(INT_RB);
16:         //ext_int_edge(L_to_H);
17:         enable_interrupts(GLOBAL);
18:         if(tiempo>=5){
19:             output_low(pin_a1); //apagar enable
20:         }
21:         else
22:         {
23:             tiempo++;
24:             delay_ms(1000);
25:         }
26:         if (tiempo>150)
27:         {
28:             tiempo=50;
29:         }
30:     }
31: }
```

ANEXO 22

Lista de códigos admitidos en la máquina fresadora CNC

Códigos G & M admitidos para control de máquina Fresadora CNC	
CÓDIGO	DESCRIPCIÓN
G00	Posicionamiento rápido (sin maquinar)
G01	Interpolación lineal (maquinando)
G02	Interpolación circular (horaria)
G03	Interpolación circular (anti-horaria)
G04	Tiempo de espera
G17	Selección del plano XY, para maquinado circular (G02 o G03)
G18	Selección del plano XZ, para maquinado circular (G02 o G03)
G19	Selección del plano YZ, para maquinado circular (G02 o G03)
G20	Comienzo de uso de unidades imperiales (pulgadas)
G21	Comienzo de uso de unidades métricas
G28	Volver al home de la máquina
G30	Ir a Posición predefinida
G53	Movimiento en coordenadas absolutas
G54-G59	Trabajo Sistemas de Coordenadas
G80	Cancelar Modo Movimiento
G90	Comandos de posicionamiento absoluto (Grupo 03).
G91	Comandos de posicionamiento incremental (Grupo 03).
G92	Establecer valor de cambio de sistema de coordenadas de trabajo
G93	Modo de avance de tiempo inverso
G94	Modo avance por minuto
M0	Detener programa
M2	Fin de programa
M30	Fin de programa y reset
M3	Activa husillo con giro en sentido horario
M4	Activa husillo con giro en sentido anti-horario
M5	Detener husillo
M8	Activa refrigerante
M9	Apaga refrigerante

Código de cabecera y configuración para maquinado CNC.

N100 G21 G17 G90

N106 G90 G54

G21: Selección de unidades métricas.

G17: Selección de maquinado en el plano XY.

G90: Uso de coordenadas absolutas.