

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE INGENIERÍA ELÉCTRICA

**IMPLEMENTACIÓN DE UN SISTEMA DE CORTE BIDIMENSIONAL
CONTROLADO POR COMPUTADORA**

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

VERÓNICA CAROLINA NOLIVOS CISNEROS

DIRECTOR: ING. FERNANDO FLORES

QUITO, FEBRERO DEL 2006

DECLARACIÓN

Yo, Verónica Carolina Nolivos Cisneros, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

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



Verónica Carolina Nolivos Cisneros

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Verónica Carolina Nolivos Cisneros, bajo mi supervisión.


Ing. Fernando Flores
DIRECTOR DE PROYECTO

AGRADECIMIENTO

A Dios, por darme la fortaleza y sabiduría para superar cada uno de los obstáculos que se me presentaron en el transcurso y culminación de mi carrera, pero sobretodo por no abandonarme nunca.

A mis padres, Blanca y Manuel, por su apoyo y amor incondicional, ya que han sido mi fortaleza durante todos éstos años en los que necesité de su calor y comprensión.

A mis hermanos, Mónica y José, por guiarme a través de sus consejos, cuidados y comprensión en el transcurso de ésta etapa tan importante como es la culminación de mi carrera.

A mi enamorado, Antonio, quien con paciencia y cariño estuvo siempre a mi lado ayudándome y luchando por hacer más suave mi camino.

A esos ángeles que son mis abuelitos; Carmen, Luzmila, Humberto y José por su compañía y protección.

Al Ingeniero Fernando Flores por aportar con su tiempo, con sus conocimientos y comprensión en todo el proceso de elaboración del Proyecto de Titulación.

Al Ingeniero Antonio Redín y su familia, quienes me brindaron todo su cariño y comprensión; pero sobretodo, me dieron esa mano amiga que tanto necesité.

MUCHAS GRACIAS

Verónica Carolina No

DEDICATORIA

A Dios, por acompañarme ayer, hoy y siempre:

A la mejor mujer del mundo; mi madre, quien con sus cuidados, abnegación y cariño supo guiarme y comprenderme durante todos éstos años.

A mi padre, quien me enseñó a través de su ejemplo a luchar con honestidad por mis sueños y a levantarme después de cada tropiezo.

A mis hermanos, por todas sus atenciones y cuidados; por nunca abandonarme y por recordarme día a día que siempre estaremos juntos como la familia que somos apoyándonos, ayudándonos y fortaleciéndonos.

A mi enamorado, quien estuvo a mi lado siempre, apoyándome y dándome fuerzas para seguir luchando, pero en especial, porque en cada tropiezo estuvo ahí para levantarme.

“SOLO VENCIÉNDOTE VENCERAS”

Los quiero mucho.

CONTENIDO

RESUMEN	X
PRESENTACIÓN	XI

INDICE GENERAL

CAPÍTULO 1

FUNDAMENTOS TEÓRICOS.

CAPÍTULO 1.....	1
FUNDAMENTOS TEÓRICOS.....	1
1.1 INTRODUCCIÓN.....	1
1.2 CONTROL NUMÉRICO (CN).....	2
1.2.1 DEFINICIÓN.....	3
1.2.2 ELEMENTOS DE UN SISTEMA DE CONTROL NUMÉRICO.....	4
1.2.3 FORMAS DE CONTROL NUMÉRICO.....	6
1.2.3.1 Control numérico (CN).....	6
1.2.3.2 Control numérico por computadora (CNC).....	7
1.2.3.3 Control numérico adaptativo (CNA).....	7
1.2.3.4 Control numérico directo o distribuido (DNC).....	8
1.2.3.5 Controladores lógicos programables.....	9
1.2.3.6 Programación del control numérico.....	9
1.2.3.7 Máquinas Herramientas de Control Numérico (MHCN).....	10
1.3 SISTEMA DE POSICIONAMIENTO.....	10
1.3.1 SISTEMA DE COORDENADAS CARTESIANAS.....	11
1.3.2 EJES DE COORDENADAS EN CNC.....	13
1.3.2.1 Ejes Principales.....	14
1.3.2.2 Sistemas de Coordenadas de Dos y Tres Ejes.....	14
1.3.2.3 Ejes Complementarios.....	15
1.3.2.4 Desplazamientos Lineales y Circulares.....	16
1.3.3 MOVIMIENTO Y CONTROL DE MÁQUINAS HERRAMIENTAS. ...	19

1.3.4	SISTEMA DE REFERENCIA Y CONTROL DE TRAYECTORIA DE LOS CONTROLES NUMÉRICOS.....	20
1.3.4.1	Según el Sistema de Referencia.....	21
1.3.4.2	Según el control de las trayectorias.....	21
	Punto a Punto.....	22
	Paraxial.....	23
	De contorneado.....	23
	Interpolación.....	25
1.3.5	MECANISMOS DE POSICIONAMIENTO.....	26
1.3.5.1	Sistema de Bucle Cerrado.....	27
1.3.5.2	Sistema de Bucle Abierto.....	27
1.4	MOTOR PASO A PASO (PAP).....	28
1.4.1	PRINCIPIO DE FUNCIONAMIENTO.....	30
1.4.2	TIPOS DE MOTORES A PASO SEGÚN SU CONSTRUCCIÓN.....	31
1.4.2.1	Motores a paso de Reluctancia variable (VR).....	31
1.4.2.2	Motores a paso de Imán permanente (PM).....	31
1.4.2.3	Motores a paso Híbridos (HB).....	31
1.4.3	TIPOS DE BOBINADOS.....	33
1.4.3.1	Motores de Reluctancia Variable.....	33
1.4.3.2	Motores Unipolares.....	33
1.4.3.3	Motores Bipolares.....	34
1.4.3.4	Motores Bifilares.....	34
1.4.3.5	Motores Multifase.....	34
1.4.4	MODO DE TRABAJO.....	36
1.4.4.1	Paso Completo.....	36
1.4.4.2	Medio Paso.....	37
1.4.4.3	Micropasos.....	37
1.5	SISTEMAS DE CORTE.....	38
1.6	SISTEMAS DE TRANSMISIÓN.....	39
1.6.1	TRANSMISIÓN SERIAL.....	39
1.6.2	TRANSMISIÓN PARALELA.....	40

CAPÍTULO 2

DESCRIPCIÓN DEL SISTEMA MECÁNICO NEURACTOR CNC-III.

CAPÍTULO 2.....	42
DESCRIPCIÓN DEL SISTEMA MECÁNICO NEURACTOR CNC-III.....	42
2.1 INTRODUCCIÓN.....	42
2.2 REQUERIMIENTOS DEL SISTEMA.....	43
2.3 DISEÑO DEL SISTEMA MECÁNICO.....	44
2.3.1 CONSTRUCCIÓN DEL SISTEMA MECÁNICO NEURACTOR CNC-III. 45	
2.3.2 FUENTE DE PODER.....	46
2.3.3 FUNCIÓN DE LA TARJETA DE EFECTO DE CAMPO (HALL-EFFECT).....	47
2.3.4 TARJETA DE INTERFACE.....	49
2.3.5 PANEL DE CONTROL.....	52
2.4 MOTORES ELÉCTRICOS.....	53
2.4.1 MOTORES DE CORRIENTE DIRECTA (DC).	54
2.4.1.1 TIPOS DE MOTORES DE CORRIENTE CONTÍNUA.....	56
2.4.1.2 MOTOR DC SERIE.	57
2.4.1.3 MOTORES PARA USOS ESPECIALES.	58
2.4.2 MOTORES UNIVERSALES.....	60
2.4.2.1 CARACTERÍSTICAS DEL PAR DE LOS MOTORES UNIVERSALES.	61
2.4.2.2 PRINCIPIO DE FUNCIONAMIENTO DE LOS MOTORES UNIVERSALES.	61
2.4.3 MOTORES A PASOS.....	62
2.4.3.1 PRINCIPIO DE FUNCIONAMIENTO.....	63
2.4.3.2 TIPOS DE MOTORES PASO A PASO.....	64
2.5 DESCRIPCIÓN DE LOS MOTORES DE PASOS UTILIZADOS EN EL SISTEMA MECÁNICO NEURACTOR CNC-III.....	66

2.6	DESCRIPCIÓN DE LOS MICROCONTROLADORES CY-545	68
2.6.1	COMANDOS PARA INTERFASES.	70
2.6.1.1	Interfase Serial.	70
2.6.1.2	Interfase Paralelo.	71
2.6.1.3	Interfase del CY545 al CY233.	72
2.6.2	SUMARIO Y FORMATO DE LOS COMANDOS.....	72
2.6.2.1	Formato De Comandos ASCII y Binario.	73
2.6.2.2	Comandos de Control del Motor.....	73
2.7	CONTROLADOR DE REDES LOCALES CY-233.	74
2.7.1	MODOS DE OPERACIÓN.....	76
2.7.2	COMANDOS DEL CY-233.....	76
2.7.3	CONEXIÓN AL COMPUTADOR.....	78
2.8	DESCRIPCIÓN DEL PROGRAMA CIMCO EDIT V.5.....	78
2.8.1	FUNCIONES.....	79
2.8.2	MENÚ ARCHIVO.....	80
2.8.3	MENÚ EDICIÓN.....	80
2.8.4	MENÚ DE FUNCIONES NC.....	81
2.8.5	MENÚ DE TRANSMISIÓN.....	82
2.8.6	MENÚ PROGRAMACIÓN.....	82
2.8.6.1	Parámetros Del Programa General.....	83
2.8.7	MENÚ CNC-CALC.....	83
2.8.8	ARCHIVOS CDD Y TXT.....	85

CAPÍTULO 3. DESARROLLO DEL SOFTWARE.

3.1	INTRODUCCIÓN.....	87
3.2	DIAGRAMA DE FLUJO GENERAL.....	88
3.2.1	DIAGRAMA DE FLUJO DEL PROCESO DE CORTE EN MODO AUTOMÁTICO.....	89
3.2.1.1	Dibujo. frm.....	94
3.2.1.2	Sub Archivo_CBC_Click().....	98
3.2.1.3	Auto.frm.....	99
3.2.1.4	Sub Cortar.....	103
3.3	PROGRAMA PRINCIPAL.....	105
3.3.1	MENÚ PRINCIPAL.....	105
3.3.2	MANEJO MANUAL DEL CORTADOR.....	106
3.3.2.1	Controles del Display.....	108
3.3.2.2	Controles.....	109
3.3.3	MANEJO AUTOMÁTICO DEL CORTADOR.....	111
3.3.3.1	Parámetros.....	114
3.3.3.2	Porcentaje de Corte.....	115
3.3.4	MANEJO DE ARCHIVOS.....	115

CAPÍTULO 4. RESULTADOS EXPERIMENTALES.

4.1	INTRODUCCIÓN.....	117
4.2	RECTANGULO. CDD.....	118
4.2.1	RECTANGULO.TXT	119
4.2.2	RECTANGULO.CBC	120
4.3	ESTRELLA. CDD.	120
4.3.1	ESTRELLA.TXT.....	121
4.3.2	ESTRELLA.CBC.....	122
4.4	CIRCULO_CORTE.CDD.	123
4.4.1	CIRCULO_CORTE.TXT	124
4.4.2	CIRCULO_CORTE.CBC.....	124
4.5	OCHO.CDD.....	125
4.5.1	OCHO.TXT	126
4.5.2	OCHO.CBC	126
4.6	R.CDD.....	128
4.6.1	R.TXT.....	128
4.6.2	R.CBC	129
4.7	CAMPANA.CDD.....	131
4.7.1	CAMPANA.TXT	132
4.7.2	CAMPANA.CBC	133
4.8	RESULTADOS GENERALES.....	134

CAPÍTULO 5. CONCLUSIONES Y RECOMENDACIONES.

5.1	CONCLUSIONES GENERALES.....	138
5.2	RECOMENDACIONES.....	140

INDICE DE FIGURAS

CAPÍTULO 1

FIG. 1. 1 ELEMENTOS BÁSICOS DE UN SISTEMA DE CONTROL NUMÉRICO.....	5
FIG. 1. 2 COMPONENTES DE UN SISTEMA CNC.	5
FIG. 1. 3 LOS MÉTODOS DE CONTROL QUE PROPORCIONAN (A) POSICIÓN O CONTROL DE PUNTO A PUNTO O (B) CONTROL SOBRE EL MOVIMIENTO DE LA PIEZA O DE LA HERRAMIENTA EN UN CORTE RECTO O (C) A LO LARGO DE UN CONTORNO.	6
FIG. 1. 4 MUCHAS MÁQUINAS DE CONTROL NUMÉRICO (CN), ACTUALMENTE SE CONTROLAN POR (A) UNA MICROCOMPUTADORA DEDICADA (CNC) O (B) UNA JERARQUÍA DE COMPUTADORAS (DNC).	8
FIG. 1. 5 (A) LÍNEAS CRUZADAS FORMANDO EL ORIGEN; (B) PLANOS DE TRES DIMENSIONES USADO EN CNC.....	12
FIG. 1. 6 LOS CUADRANTES FORMADOS CUANDO LOS EJES X Y Y SE CRUZAN SE USAN PARA LOCALIZAR PUNTOS CORRECTAMENTE DESDE EL CERO XY O PUNTO ORIGEN.	13
FIG. 1. 7 DESPLAZAMIENTOS-EJE DE UNA FRESADORA	14
FIG. 1. 8 PIEZA EN UN SISTEMA DE COORDENADAS EN 2-D	14
FIG. 1. 9 COORDENADAS 3-D.....	15
FIG. 1. 10 (A) MESA GIRATORIA Y CABEZAL BASCULANTE; (B) CENTRO DE MECANIZADO DE 6 EJES	16
FIG. 1. 11 PLANOS FUNDAMENTALES	17
FIG. 1. 12 DESPLAZAMIENTOS CIRCULARES: (A) EN DOS DIMENSIONES 2D, (B) EN EL PLANO XY Y (C) EN TRES DIMENSIONES 3D.	17
FIG. 1. 13 SISTEMA DIRECTO PARA LA MEDICIÓN DE UNA POSICIÓN	18
FIG. 1. 14 SISTEMA INDIRECTO PARA LA MEDICIÓN DE UNA POSICIÓN	18
FIG. 1. 15 LAS HERRAMIENTAS Y PIEZAS DE TRABAJO PUEDEN MOVERSE Y EL CONTROL SE PUEDE EJERCER A LO LARGO DE (A) UNO, DOS, (B) TRES, O (C) VARIOS EJES.....	19
FIG. 1. 16(A) CAVIDAD DE PAREDES RECTAS FÁCIL DE PRODUCIR, (B) SE PUEDE NECESITAR UN ÁNGULO PARA RETIRAR LA HERRAMIENTA. (C), (D) FORMAS SEGADAS REQUIEREN CONTROL DE EJES MÚLTIPLES O HERRAMIENTAS COMPLEJAS.	20
FIG. 1. 17 EL CAMINO SEGUIDO POR EL POSICIONAMIENTO PUNTO A PUNTO PARA ALCANZAR VARIOS PUNTOS PROGRAMADOS EN LOS EJES XY.	22
FIG. 1. 18 CONTROL PARAXIAL.....	23
FIG. 1. 19 TIPOS DE CONTORNO. (A) CONTORNO SIMPLE Y (B) CONTORNO COMPLEJO.....	24
FIG. 1. 20 UN EJEMPLO DE INTERPOLACIÓN LINEAL EN DOS EJES.	25
FIG. 1. 21 PARA LA INTERPOLACIÓN CIRCULAR BIDIMENSIONAL DEL MCU SE SUMINISTRAN LOS VALORES DEL RADIO, PUNTO DE INICIO, PUNTO FINAL Y DIRECCIÓN DE CORTE.....	26
FIG. 1. 22 (A) MOTOR A PASO Y (B) LA TENSIÓN EXTERNA APLICADA A LOS EXTREMOS DE LAS BOBINAS DEL INDUCIDO, CONECTADAS A LAS DOS PARTES DEL COLECTOR, SE INTRODUCE POR LAS ESCOBILLAS.	29
FIG. 1. 23 EL CONMUTADOR GIRATORIO SELECCIONA LA BOBINA QUE ATRAVESARÁ LA CORRIENTE.	30
FIG. 1. 24 (A) ESTRUCTURAS MECÁNICAS DEL MOTOR DE RELUCTANCIA VARIABLE, (B) MOTOR DE IMÁN PERMANENTE Y (C) OFF-SET EN LOS DIENTES DEL MOTOR HÍBRIDO.....	32
FIG. 1. 25 BOBINADO DE LOS MOTORES DE RELUCTANCIA VARIABLE, (B) MOTOR DE IMÁN PERMANENTE O HÍBRIDO BIPOLAR, (C) MOTOR DE IMÁN PERMANENTE O HÍBRIDO UNIPOLAR, (D) MOTOR BIFILAR Y (E) MOTOR MULTIFASE.	36
FIG. 1. 26 (A) MANEJO DEL MOTOR DE PASOS POR OLAS, (B) MANEJO DEL MOTOR DE PASOS POR DOS FASES.	37

CAPÍTULO 2

FIG. 2. 1 ESQUEMA DEL NEURACTOR CNC-III.	44
FIG. 2. 2 NEURACTOR CNC-III.....	46
FIG. 2. 3 DETALLES DE LA FUENTE DE PODER.	47
FIG. 2. 4 TARJETA HALL EFFECT.	48
FIG. 2. 5 ESQUEMA DE LA TARJETA DE INTERFASE.	50
FIG. 2. 6 CABLE DE LA TARJETA DE INTERFASE PARA LA TRANSMISIÓN PARALELA.....	50
FIG. 2. 7 TARJETA DE INTERFASE COMPLETA Y CONECTADA AL CABLE PARA LA TRANSMISIÓN PARALELA.	51
FIG. 2. 9 PANEL DE CONTROL.	52
FIG. 2. 10 MOTORES DE CORRIENTE DIRECTA.	54
FIG. 2. 11 (A) ESTATOR Y (B) ROTOR.	55
FIG. 2. 12 (A) MOTOR SHUNT, (B) MOTOR SERIE, (C) MOTOR COMPOUND, (D) MOTOR SERIE CON INDUCTOR PARTIDO.	56
FIG. 2. 13 MÁQUINA UNIVERSAL CONECTADA EN SERIE.	60
FIG. 2. 14 CURVA PAR-VELOCIDAD DE UN MOTOR UNIVERSAL (A) NO COMPENSADO DE POLOS SALIENTES DE 1/4CV, 8000 RPM; Y (B) COMPENSADO DE 1/4CV, 4000 RPM.	61
FIG. 2. 15 (A) IMAGEN DEL ROTOR, (B) IMAGEN DE UN ESTATOR DE 4 BOBINAS.	63
FIG. 2. 16 EFECTO DE LA RAPIDEZ DE LA SECUENCIA DE LOS IMPULSOS SOBRE EL PAR ÚTIL DE LOS MOTORES ORDINARIOS A PASOS.	64
FIG. 2. 17 MOTORES A PASOS BIPOLARES.	65
FIG. 2. 18 MOTOR A PASOS UNIPOLAR.	65
FIG. 2. 19 MOTOR A PASOS U.S. CYBERLAB.....	68
FIG. 2. 20 VELOCIDAD SERIAL FIJA Y ADAPTATIVA.	71
FIG. 2. 21 INTERFASE DEL CY545 AL CY233.....	72
FIG. 2. 22 (A) RED DE ANILLO HOST- EL HOST ESTÁ EN EL ANILLO; (B) REDES DE ANILLO PUNTO A PUNTO- TODOS LOS DISPOSITIVOS ESTÁN EN EL ANILLO.	75
FIG. 2. 23 (A) COMANDOS PARALELOS Y SERIALES; (B) COMANDOS LAN.....	77
FIG. 2. 24 CONEXIÓN DEL MAX233 AL CY233.	78
FIG. 2. 25 PANTALLA PRINCIPAL CON LAS FUNCIONES DEL PROGRAMA CIMCO EDIT V.5.	79
FIG. 2. 26 BARRA DE HERRAMIENTAS DEL MENÚ ARCHIVO.	80
FIG. 2. 27 BARRA DE HERRAMIENTAS DE EDICIÓN.	81
FIG. 2. 28 BARRA DE HERRAMIENTAS NC.	81
FIG. 2. 29 BARRA DE HERRAMIENTAS DNC DE ENVIAR/RECIBIR.	82
FIG. 2. 30 PROGRAMACIÓN GENERAL.....	83
FIG. 2. 31 HERRAMIENTAS DEL MENÚ CNC_CALC.....	84
FIG. 2. 32 ARCHIVO MUJER.CDD.	85

CAPÍTULO 3

FIG. 3. 1 DIAGRAMA DE OPCIONES PARA EL MANEJO DEL EQUIPO.....	88
FIG. 3.2 (A) DIAGRAMA DE FLUJO: (A) DEL PROCESO ARCHIVO, (B) DEL PROCESO DE CONSTRUCCIÓN DEL ARCHIVO CBC, (C) DEL PROCESO DE CORTE AUTOMÁTICO, (D) DEL PROCESO DE CORTAR.	93
FIG. 3. 3 MENÚ PRINCIPAL.....	106
FIG. 3. 4 MANEJO MANUAL DEL CORTADOR.....	107
FIG. 3. 5 MANEJO AUTOMÁTICO DEL CORTADOR ANTES DE INICIALIZAR EL EQUIPO, Y AL SELECCIONAR UN ARCHIVO PARA INICIAR EL CORTE.....	112
FIG. 3. 6 MANEJO DE ARCHIVOS.....	115

CAPÍTULO 4

FIG. 4. 1 RECTANGULO.CDD.	119
FIG. 4. 2 ESTRELLA.CDD.....	121
FIG. 4. 3 CIRCULO_CORTE.CDD.....	123
FIG. 4. 4 OCHO.CDD.....	125
FIG. 4. 5 R.CDD.	128
FIG. 4. 6 CAMPANA.CDD.	131

ANEXOS

ANEXOS CAPÍTULO 1

ANEXOS CAPÍTULO 2

ANEXOS CAPÍTULO 3

RESUMEN

El presente Proyecto de Titulación, "Implementación de un Sistema de Corte Bidimensional Controlado por Computadora", consta de cinco capítulos en su contenido.

El Primer Capítulo, hace referencia a aquellos principios básicos referentes al Control Numérico, elementos, tipos, generalidades, etc., Sistemas de Posicionamiento, Sistemas de Referencia, Motores a Pasos, Sistemas de Corte y Sistemas de Transmisión; que son necesarios para la implementación del Sistema de Corte. Los temas tratados en éste capítulo llevan una breve explicación acerca de sus características, clasificaciones, usos, ventajas y desventajas.

El Segundo Capítulo, descripción del sistema mecánico utilizado para la Implementación del Sistema de corte; cuya máquina herramienta lleva el nombre de NEURACTOR CNC-III. Dentro de éste capítulo, el lector encontrará la descripción para la construcción del equipo, los requerimientos del sistema, una descripción más amplia de los motores utilizados por el equipo, los microcontroladores utilizados, con sus características principales y modos de comunicación, entre ellos, el equipo y el computador.

El Tercer y Cuarto Capítulo, hacen referencia del programa utilizado para controlar al NEURACTOR CNC-III, que en éste caso se refiere al sistema operativo Visual Basic. El Tercer Capítulo desarrolla el programa, mientras que el Cuarto Capítulo mencionan todas las pruebas realizadas con el equipo, referentes a los tipos de corte que realiza el equipo según el trazo que conforma la figura a recortar, así como las características y parámetros óptimos que se recomiendan conservar para obtener un trabajo final agradable para el usuario..

En cambio, en el Quinto Capítulo, se dan las Conclusiones y Recomendaciones que se adquirieron durante todo el proceso de elaboración del Proyecto de Titulación. Éste capítulo dará mayor información al lector acerca de la Implementación del Sistema de Corte, sus problemas y soluciones.

PRESENTACIÓN

En el transcurso de los dos últimos siglos la manufactura y automatización han sido los motores del desarrollo económico, debido a que estos procesos se encargan de la fabricación de bienes y artículos, controlados y operados por maquinarias (La manufactura es una actividad humana que se ha difundido en todas las fases de nuestra vida).

Con el desarrollo de la manufactura aparecen procesos de fabricación de bienes y artículos realizados a mano cuyo grado de precisión era bajo y que requería una gran inversión de tiempo. La necesidad de crear piezas con mayor precisión, rapidez, seguridad y bajos costos de producción ha llevado a investigadores y operadores a diseñar sistemas y componentes que permiten automatizar éstos procesos de manufactura.

Es necesario mencionar a los sistemas de corte como una aplicación de los procesos de manufactura y automatización, ya que es una actividad humana que se difunde en todas las fases de nuestra vida. Muchos sistemas de corte utilizan motores a paso, sistemas de posicionamiento y control numérico, debido a que estos factores facilitan el buen desenvolvimiento de la máquina herramienta.

El NEURACTOR CNC-III es uno de éstos dispositivos electromecánicos que aparecen con el desarrollo de la automatización, diseñado para la producción de figuras exactas, perforación de agujeros con variación de profundidades, aplicación de spray o pegamento con movimientos en tres dimensiones, cortado de materiales con diferentes tipos de herramientas, corte de artículos de cuero o tejidos siguiendo un patrón, estampado de hojas de metal, automatización del movimiento de una fotografía en un stand, control de movimiento para fotografía y otros tipos de aplicaciones que se los realiza con alta exactitud y rapidez.

CAPÍTULO 1

FUNDAMENTOS TEÓRICOS.

CAPÍTULO 1.

FUNDAMENTOS TEÓRICOS.

1.1 INTRODUCCIÓN.

En el transcurso de los dos últimos siglos la manufactura y automatización han sido los motores del desarrollo económico, debido a que estos procesos se encargan de la fabricación de bienes y artículos, controlados y operados por maquinarias.

La manufactura es una actividad humana que se ha difundido en todas las fases de nuestra vida. "La palabra manufactura se deriva del latín manus=mano, factus=hecho, y en los diccionarios se lo encuentra como la fabricación de bienes y artículos a mano o especialmente por maquinaria, frecuentemente en gran escala y con división del trabajo"¹.

"La palabra automático en cambio se deriva del griego, y significa automotor o auto pensante. La palabra automatización se acuñó para indicar aspectos de manufactura en los que la producción, el movimiento y la inspección se realizan o controlan por máquinas que se operan a sí mismas sin la intervención humana"². El movimiento de materiales, piezas y herramientas es un elemento importante en todas las operaciones de manufactura, que abarca la producción de piezas y su ensamblaje.

Es necesario mencionar a los sistemas de corte como una aplicación de los procesos de manufactura y automatización, ya que es una actividad humana que se difunde en todas las fases de nuestra vida. Muchos sistemas de corte utilizan motores a paso, sistemas de posicionamiento y control numérico que más

¹ SCHEY, John A., Procesos de Manufactura , McGraw-Hill, Tercera Edición, pág.3

² SCHEY, John A., Procesos de Manufactura , McGraw-Hill, Tercera Edición, pág. 34

adelante se explicarán con mayor precisión, debido a que tienen una influencia profunda en los controles necesarios del proceso. Esto es más obvio en el maquinado, en el cual la herramienta de corte sigue una trayectoria exacta para generar la forma requerida, en donde las herramientas y piezas de trabajo pueden moverse y el control se puede ejercer a lo largo de uno, dos, tres o varios ejes.

Así, el objetivo de éste primer capítulo no es el de proporcionar información detallada, sino impartir un conocimiento de los principios básicos que se pueden utilizar para considerar las capacidades del sistema, y sus implicaciones en el diseño de máquinas.

1.2 CONTROL NUMÉRICO (CN).

Los cambios bruscos en lo tecnológico, económico y político que se dan en la actualidad, obligan a sociedades como la nuestra buscar soluciones como la libre competencia o la adecuación de las industrias a fin de satisfacer los retos de los próximos años.

Una opción frente a esto es la reconversión de las industrias introduciendo la automatización, de modo que se pueda absorber gradualmente la nueva tecnología en un tiempo adecuado; todo esto sin olvidar los factores de rendimiento de la inversión y capacidad de producción.

El control numérico es un ejemplo de automatización programable que se lo realiza desde 1957 y se diseñó para adaptar las variaciones en la configuración de los productos. Su principal aplicación se centra en volúmenes de producción bajos y medios, permitiendo la mecanización de piezas muy complejas.

A continuación se presenta brevemente el origen y desarrollo del control numérico con los diferentes tipos de controles abiertos.

- (1725) Máquinas de tejer construidas en Inglaterra, controladas por tarjetas perforadas.
- (1863) M. Forneaux- primer piano que tocó automáticamente.
- (1870-1890) Eli Whitney- desarrollo de plantillas y dispositivos.
- (1880) Introducción de herramientas para el maquinado de metales.
- (1940) Introducción de los controles hidráulicos, neumáticos y electrónicos.
- (1945) Comienzo de la investigación y desarrollo del control numérico.
- (1955) Las herramientas automatizadas comenzaron a aparecer en las plantas de producción para la Fuerza Aérea de los Estados Unidos:
- (1956) Concentración en la investigación y desarrollo del control numérico.
- (1960) Hasta la actualidad
 - Se crean varios y nuevos sistemas de control numérico.
 - Se perfeccionaron las aplicaciones a la producción de una gama más grande de procedimientos de maquinado de metales.
 - Se idearon aplicaciones a otras actividades diferentes del maquinado de metales.
 - Se utilizan insumos computarizados de control numérico.
 - Se utilizan documentos computarizados de planeación gráficos por control numérico.
 - Se han desarrollado procedimientos computarizados de trazo de curvas de nivel por control numérico, a bajo costo.
 - Se han establecido centros de maquinado para utilización general.

1.2.1 DEFINICIÓN.

"Control numérico es todo dispositivo capaz de dirigir posicionamientos de un órgano mecánico móvil, en el que las órdenes relativas a los desplazamientos del móvil son elaboradas a partir de las instrucciones codificadas en un programa, y que realiza un mando mediante números, haciendo que las máquinas desarrollen su trabajo automáticamente mediante la introducción en su memoria de un

programa en el que se definen las operaciones a realizar por medio de combinaciones de letras y números (códigos G)".³

Aplicando un sistema de Control Numérico a una herramienta de corte, se consigue su automatización, que consiste en el control de sus acciones, entre las que se encuentran los movimientos de los carros y cabezal, el valor y el sentido de las velocidades de avance y de corte, las condiciones de funcionamiento de la máquina, la coordinación, etc.

1.2.2 ELEMENTOS DE UN SISTEMA DE CONTROL NUMÉRICO.

Al tener una idea acerca del Control Numérico, se indican a continuación los elementos más comunes y básicos que conforman éste sistema:

- **El programa**, contiene la información para el desarrollo de diferentes tareas, que se escribe en un lenguaje especial (códigos G) compuesto por letras y números y se graba en un soporte físico (cd, disquete, etc.) o se envía directamente a través del RS-232.⁴
- **El control numérico (CN)**, interpreta las instrucciones contenidas en el programa, convirtiéndolas en señales que accionan los dispositivos de las máquinas comprobando así su resultado.
- **El equipo de procesado**, es el que realiza el trabajo útil, compuesto por la mesa de trabajo, máquinas-herramientas, motores y controles de movimiento.

³ SABATER, J., Control Numérico, Universitas Miguel Hernández, pág. 3

⁴ Estándar de EIA que especifica las características eléctricas de las interconexiones de la velocidad entre terminales y computadoras. Aún cuando el estandar más utilizado es RS232C.

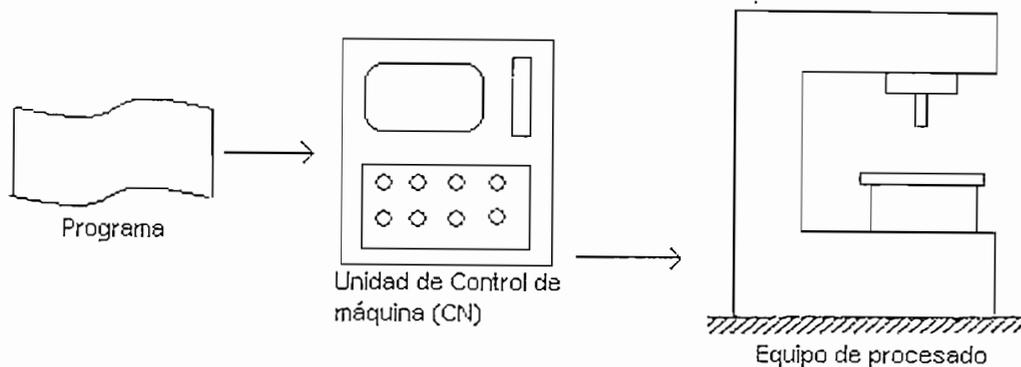


Fig. 1. 1 Elementos básicos de un sistema de control numérico.

El corazón de un sistema de Control Numérico por Computadora es el computador que se encarga de realizar los cálculos necesarios y las conexiones lógicas, siendo de ésta forma el puente de unión entre el operador y la máquina-herramienta que requiere de dos interfaces o traductores, Fig. 1.2:

- El interfaz del operador esta formado por el panel de control y los dispositivos periféricos encargados del almacenamiento (casete, disqueteras, etc) o de la impresión de la información.
- El interfaz de control de la máquina-herramienta esta subdividido en múltiples conexiones de control que afectan a los actuadores de los ejes, husillo principal, etc., y que llegan al sistema auxiliar de alimentación de energía.

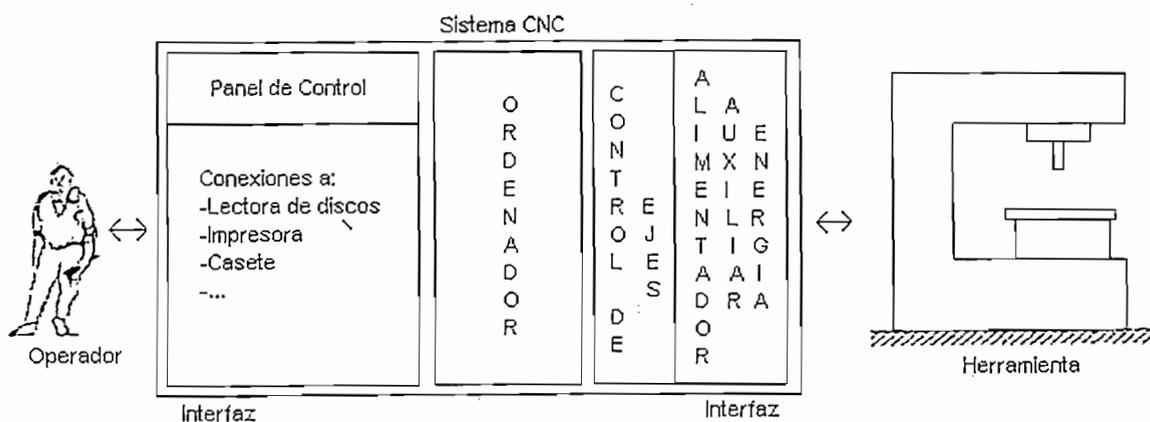


Fig. 1. 2 Componentes de un sistema CNC⁵.

⁵ Control Numérico por Computadora (CNC).

1.2.3 FORMAS DE CONTROL NUMÉRICO.

Se han desarrollado varias formas de CN para el control automático de procesos mediante instrucciones codificadas, las mismas que se indican a continuación:

1.2.3.1 Control numérico (CN).

El hardware para el CN incluye *servomotores*, *dispositivos de retroalimentación* (control lazo cerrado) y la *unidad de control de la máquina* (UCM) que contiene la lógica para traducir la información a una acción apropiada esperando que la máquina herramienta se mueva de un punto a otro. Su principal característica es la de trabajar sin memoria, con una cinta perforada como medio de introducción del programa que se ejecuta secuencialmente, siendo leído en la UCM.

1. Si el dispositivo mecánico está conformado por dos servomotores colocados en las coordenadas x-y, la UCM mueve primero el servomotor x y luego el y de acuerdo con las distancias prescritas, manteniendo un sistema punto a punto o de posicionamiento (Fig. 1.3 (a)), que realiza la operación cuando se alcanza la posición programada. Un sistema más complejo cuenta con un control de velocidad o sistema de corte directo Fig. 1.3 (b), y mueve el servomotor en una dirección y luego en otra, mientras se da una operación.
2. Cuando un *sistema de contorno* es complejo la UMC se programa para realizar la interpolación entre los puntos extremos de un segmento y descomponer los contornos en partes más pequeñas, Fig. 1.3 (c).

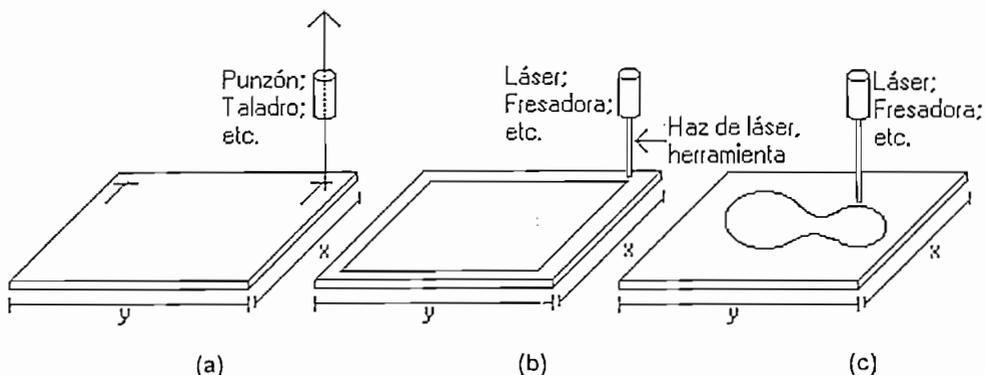


Fig. 1.3 Los métodos de control que proporcionan (a) posición o control de punto a punto o (b) control sobre el movimiento de la pieza o de la herramienta en un corte recto o (c) a lo largo de un contorno.⁶

⁶ SCHEY, John A., Procesos de Manufactura , McGraw-Hill, Tercera Edición, pág. 35

1.2.3.2 Control numérico por computadora (CNC).

El control numérico por computadora no es una tecnología nueva, ya que existe desde 1957, siendo uno de los factores importantes para la fabricación de productos en todo el mundo.

La computadora asume las funciones de la UMC, Fig. 1.4 (a), cuyo programa es leído por la memoria, se obtiene flexibilidad de operación, dando la posibilidad de trazar curvas complejas sin romper la continuidad. Los CNC incluyen una memoria interna que permite el almacenamiento del programa de la pieza, de los datos de la máquina y de las compensaciones de las herramientas. Incorporan un teclado que facilita la comunicación y el grado de interactividad con el operario y permiten la ruptura de la secuencia de los programas. De esta forma, se tiene una programación estructurada y fácil de aprender.

Tanto el CN como el CNC elevan la productividad, aumentando de ésta forma la precisión, calidad y confiabilidad del producto final. El CNC minimiza los errores introducidos por el lector de cinta y reduce los gastos generales relativos al CN.

1.2.3.3 Control numérico adaptativo (CNA).

Es la tendencia actual de los controles, donde el controlador detecta las características del mecanizado que está realizando y en función de ellas optimiza las velocidades de corte y los avances. Para ello, hace uso de sistemas sensoriales de fuerza y deformación en la herramienta, temperatura de corte, vibraciones, potencia, etc.

Las razones de la introducción del CNA residen en la variación de las condiciones de corte durante el mecanizado por varios motivos:

- Geometría variable de la sección de corte (profundidad y anchura) por la complejidad de la superficie a mecanizar.
- Variaciones en la dureza y en la maquinabilidad de los materiales.

- Desgaste de las herramientas, incrementándose el esfuerzo de corte.

1.2.3.4 Control numérico directo o distribuido (DNC).

El *control numérico directo* se caracteriza por que cada máquina-herramienta tiene su propia computadora, mientras que la central se ocupa de funciones de almacenamiento, administración, monitoreo, etc.

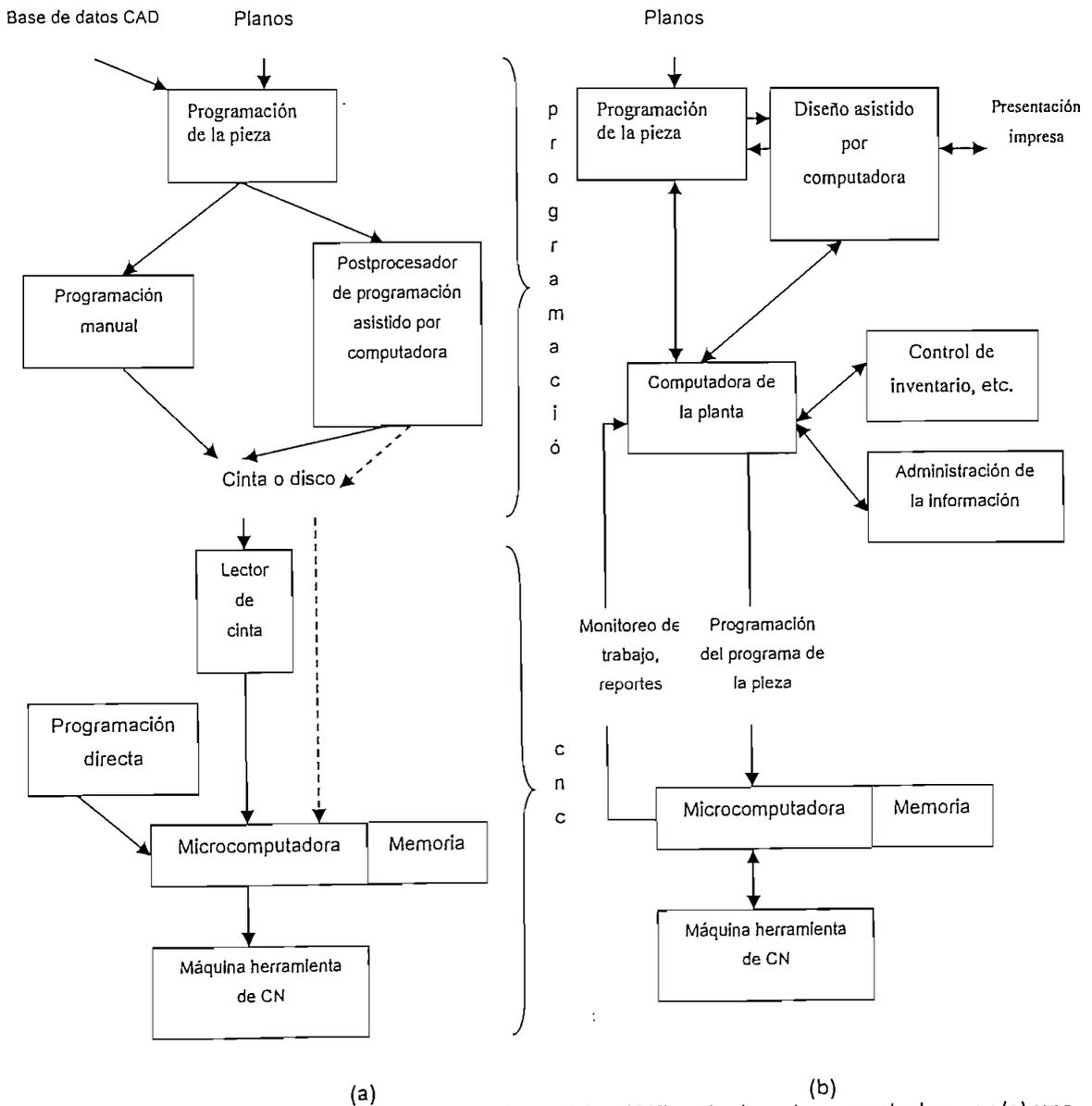


Fig. 1. 4 Muchas máquinas de control numérico (CN), actualmente se controlan por (a) una microcomputadora dedicada (CNC) o (b) una jerarquía de computadoras (DNC).

⁷ SCHEY, John A., Procesos de Manufactura , McGraw-Hill, Tercera Edición, pág. 36

1.2.3.5 Controladores lógicos programables.

El control de muchos procesos requiere funciones de secuencia, sincronización, conteo, lógica y aritmética, las que se satisfacen con circuitos lógicos relevadores. Actualmente, su lugar ha sido ocupado con controladores lógicos programables (PLC), cuya ventaja es que la memoria se puede reprogramar con un tablero de programación o una computadora.

1.2.3.6 Programación del control numérico.

La programación comienza definiendo la secuencia óptima de operaciones y las condiciones del proceso. Las características geométricas de la pieza se usan para calcular la trayectoria de la herramienta. El programa resultante puede ser muy general y se debe convertir, con la ayuda de un programa en una forma aceptable para el control particular de la máquina herramienta, cuya salida es cualquier medio de almacenamiento. Existen cuatro tipos básicos de programación del control numérico, los que se presentan a continuación:

1. **Programación manual:** los elementos del programa se calculan por un programador que los pone en instrucciones generales. La programación es laboriosa y limitada en gran parte a programas punto a punto.
2. **Programación asistida por computadora:** el programador se comunica con un sistema de software cuyo lenguaje de programación traduce la información de entrada para que ésta sea entendible para la computadora, y así realizar los cálculos necesarios. La verificación de la cinta se debe hacer en la máquina herramienta o en una máquina de dibujo.
3. **CAD/CAM:** Cuando las piezas se diseñan con CAD, la base de datos puede ser usada para generar el programa en la terminal de gráficas, cuya programación es rápida y relativamente barata.
4. **Entrada manual de datos:** Algunas máquinas herramientas de CNC están equipadas con una pantalla de VDT⁸ y un software poderoso que prepara el programa de la pieza. El operador introduce información para definir la

⁸ Video Display Terminal

geometría de la pieza, el material y las herramientas. Se usan palabras estándares y en inglés. La técnica es muy económica ya que permite la programación mientras otra aplicación está ocurriendo.

Con la aparición del Control Numérico (CNC) y mediante la entrada manual de datos, la programación de la máquina herramienta depende solo del operador; mientras que en una programación convencional asistida por computadora y CAD/CAM se lo realiza en los departamentos de programación.

1.2.3.7 Máquinas Herramientas de Control Numérico (MHCN).

Modalidad de automatización flexible conformada por máquinas herramientas programadas para fabricar lotes de pequeño y medio tamaño de piezas de formas complicadas; los programas de software controlan sus velocidades, temperatura, vibración, control adaptativo, condición del material, desgaste de la herramienta, etc., permitiendo los reajustes necesarios. En ocasiones las máquinas están dispuestas en forma semicircular para que un robot se encargue de manejar los materiales. Cuando una máquina de control numérico actúa de forma independiente, necesita de la presencia de un operario, quien se ocupa de la carga y descarga de las piezas a procesar, los programas y las herramientas.

Se cree que, en un futuro, las máquinas de Control Numérico harán el trabajo de precisión, mientras que los robots se limitarán a la carga, descarga y ensamblaje. En los casos de producción de gran volumen, la automatización será rígida, sencilla y barata.

1.3 SISTEMA DE POSICIONAMIENTO.

“En un ensamble, es necesario que las partes diseñadas encajen una con otra; por lo que las desviaciones permisibles en las dimensiones o tolerancias dimensionales, deben ser específicas, siendo necesario usar técnicas y procedimientos de medición”.⁹

⁹ SCHEY, John A., Procesos de Manufactura , McGraw-Hill, Tercera Edición, pág.43

La forma de una pieza la dicta ante todo su función, y con frecuencia la complejidad de la misma determina los procesos a seguir para fabricarla. Al elevarse la complejidad de la forma de la pieza se reduce el rango de procesos aplicables incrementando por tal razón el costo del diseño y manufactura. Por tal razón, se recomienda mantener la forma de la pieza lo más sencilla que se pueda para reducir el número de procesos a seguir, o en caso contrario, pueden usarse formas complejas si aquello permite la consolidación de varias partes y/o la eliminación de uno o más pasos de manufactura.

Para que una pieza funcione adecuadamente respecto a otros componentes, es necesario colocar restricciones adicionales a la localización o posición de las características geométricas. Esta información se transmite en los dibujos de ingeniería mediante el método de dimensionamiento por coordenadas, que es muy rápido y parecería ser idealmente adecuado para el maquinado en herramientas de control numérico.

Cabe mencionar que inicialmente se hace un análisis meramente geométrico y básico en el desarrollo de los Sistemas de Posicionamiento; en lo referente al análisis mecánico propiamente dicho se hablará con mayor detalle más adelante en éste mismo capítulo.

1.3.1 SISTEMA DE COORDENADAS CARTESIANAS.

El sistema de coordenadas cartesiano o rectangular fue inventado por el matemático filósofo francés Rene 'Descartes, que consiste en que cualquier punto puede describirse en términos matemáticos a lo largo de los ejes perpendiculares. Éste concepto encaja con las herramientas de la máquina ya que su construcción se basa en un movimiento por los ejes X, Y, Z y de rotación. En una máquina de fresado vertical, el eje X está en movimiento horizontal (derecho o izquierdo), el eje Y tiene un movimiento cruzado por la mesa (hacia o lejos de la columna), y el eje Z es el movimiento vertical de la varilla.

Los sistemas CNC confían en el uso de coordenadas rectangulares porque el programador localiza cada punto con precisión, y es que con la intersección de dos líneas rectas (una vertical y una horizontal) se localizan los puntos sobre la pieza de trabajo; estas líneas deben estar en ángulo recto la una con la otra, y a éste punto de intersección se lo conoce como origen o punto cero (Fig. 1.5).

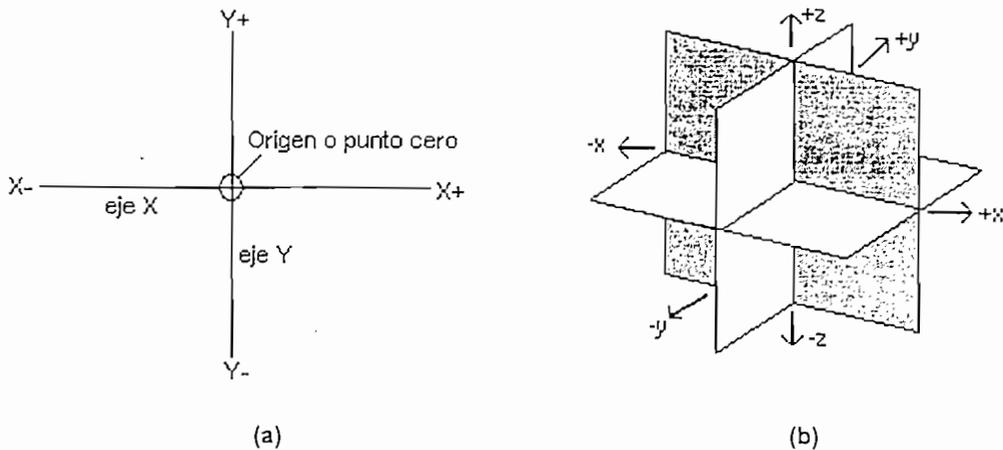


Fig. 1. 5 (a) Líneas cruzadas formando el origen; (b) Planos de tres dimensiones usado en CNC.¹⁰

Los ejes X y Y representan el movimiento horizontal de la mesa de la máquina, el plano Z representa el movimiento vertical de la herramienta. Los signos (+) o (-) dan la dirección desde el origen hacia los alrededores de los ejes de movimiento.

Los cuatro cuadrantes formados cuando los ejes XY se cruzan se numeran en dirección antihoraria (Fig. 1.6). Las posiciones localizadas en el primer cuadrante serán positivas en X y Y, en el segundo cuadrante las posiciones son en X negativo y en Y positivo, en el tercero serán negativas tanto en X como en Y, y en el cuarto las posiciones en Y son negativas y en X positivas. En la programación de CNC no se necesita indicar el valor "+" pues será asumido; el signo "-" es necesario indicarlo (la realización del presente Proyecto de Titulación trabaja únicamente en el primer cuadrante).

¹⁰ KRAR, Steve / GILL, Arthur, Computer Numerical Control Programming Basics, Machinery's Handbook, pág. 8

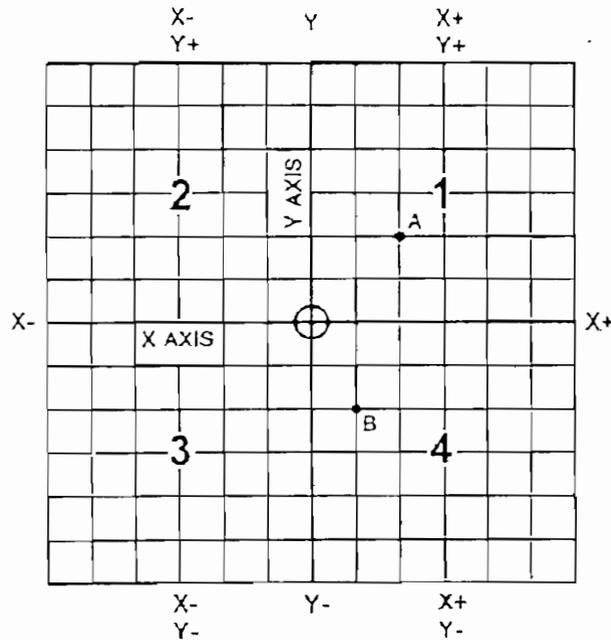


Fig. 1. 6 Los cuadrantes formados cuando los ejes X y Y se cruzan se usan para localizar puntos correctamente desde el cero XY o punto origen.¹¹

El sistema de control de la máquina herramienta convierte las coordenadas a través del programa en movimientos de la herramienta, siendo ésta la única que se mueva ya que en la operación de contorneado la pieza queda fija.

1.3.2 EJES DE COORDENADAS EN CNC.

Las herramientas de una máquina CNC pueden realizar ciertos movimientos según el tipo de máquina, por ejemplo, en un torno, los movimientos son del tipo longitudinales y transversales. Por tal razón, para controlar la herramienta de forma precisa durante estos movimientos, todos los puntos dentro del área de trabajo de la máquina deben permitir una definición clara y comprensible. Los sistemas de coordenadas se usan con este propósito, proporcionando una orientación al programador durante la confección de programas.

¹¹ KRAR, Steve / GILL, Arthur, Computer Numerical Control Programming Basics, Machinery's Handbook, pág. 9

1.3.2.1 Ejes Principales.

Generalmente las máquinas convencionales tienen de dos (X, Y) a tres ejes de coordenadas (X, Y, Z), como los tornos y las fresadoras respectivamente, pero en trabajos de mecanizado de formas complejas se requieren MHCN dotadas de más ejes de desplazamiento (Fig. 1.7).

Eje Y: desplazamiento transversal del carro portador

Eje Z: desplazamiento transversal de la mesa

Eje X: desplazamiento transversal de la mesa

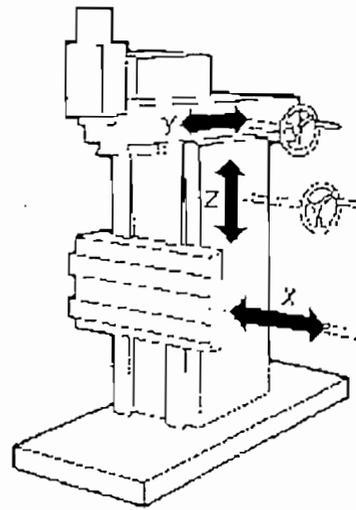


Fig. 1. 7 Desplazamientos-eje de una fresadora

1.3.2.2 Sistemas de Coordenadas de Dos y Tres Ejes.

Una forma simple para programar un sistema de coordenadas de control numérico consiste en dos ejes con intersección en ángulo recto (origen de coordenadas). Un sistema de coordenadas con dos ejes permite una descripción precisa de los puntos (vértices, centros de círculos, etc.) en el dibujo de la pieza (Fig. 1.8).

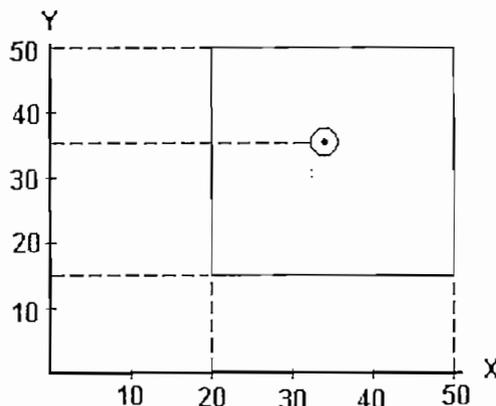


Fig. 1. 8 Pieza en un sistema de coordenadas en 2-D

Al mecanizar piezas con torno o fresadora, es necesario imaginar la pieza en 3-D, ya que el movimiento no puede describirse solo con las coordenadas X y Y, requiere una coordenada para la profundidad del agujero taladrado (eje Z). Para representar la pieza en 3-D se necesita un sistema de coordenadas con tres ejes (X, Y, Z), Fig. 1.9.

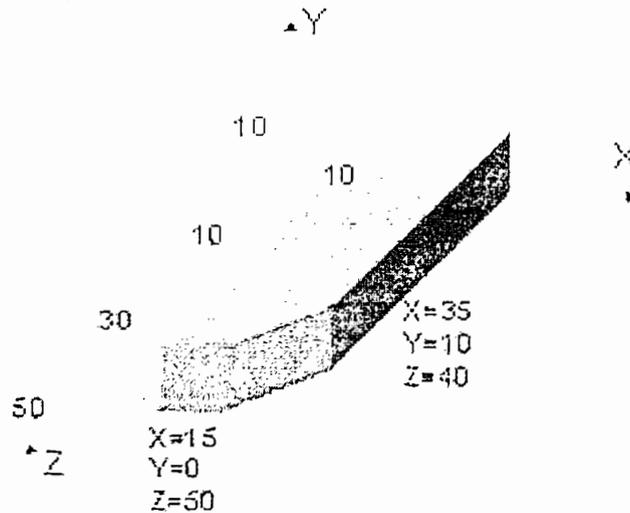


Fig. 1. 9 Coordenadas 3-D

Las coordenadas X, Y, Z de una pieza se obtienen estableciendo la posición de los puntos a escala en los tres ejes (Fig.1.9), formando siempre ángulos rectos entre sí. Cada eje tiene direcciones y valores negativos o positivos según la ubicación de la pieza con respecto al origen de coordenadas.

1.3.2.3 Ejes Complementarios.

Algunas MHCN disponen de mesas giratorias y/o cabezales para la orientación. En ellas la pieza puede ser mecanizada por diferentes planos y ángulos de aproximación.

Los ejes sobre los que giran estas mesas y cabezales se controlan de forma independiente y se conocen con el nombre de ejes complementarios de rotación, en los cuales la velocidad se regula en forma autónoma. Los ejes complementarios de rotación se designan en la programación CN como A, B, C.

Debido a las exigencias impuestas por la complejidad de ciertas piezas otras MHCN están dotadas de más de tres ejes de desplazamiento, donde los centros de mecanizado presentan usualmente un cuarto eje para la orientación del cabezal, un quinto para el giro de la mesa y hasta un sexto (W) de aproximación de la herramienta (Fig. 1.10).

En muchos casos el eje W opera sólo cuando el resto de los ejes permanecen fijos y generalmente se usa para trabajos menores de taladrado en cualquier dirección. Los ejes complementarios de desplazamiento se designan en la programación como U, V, W.

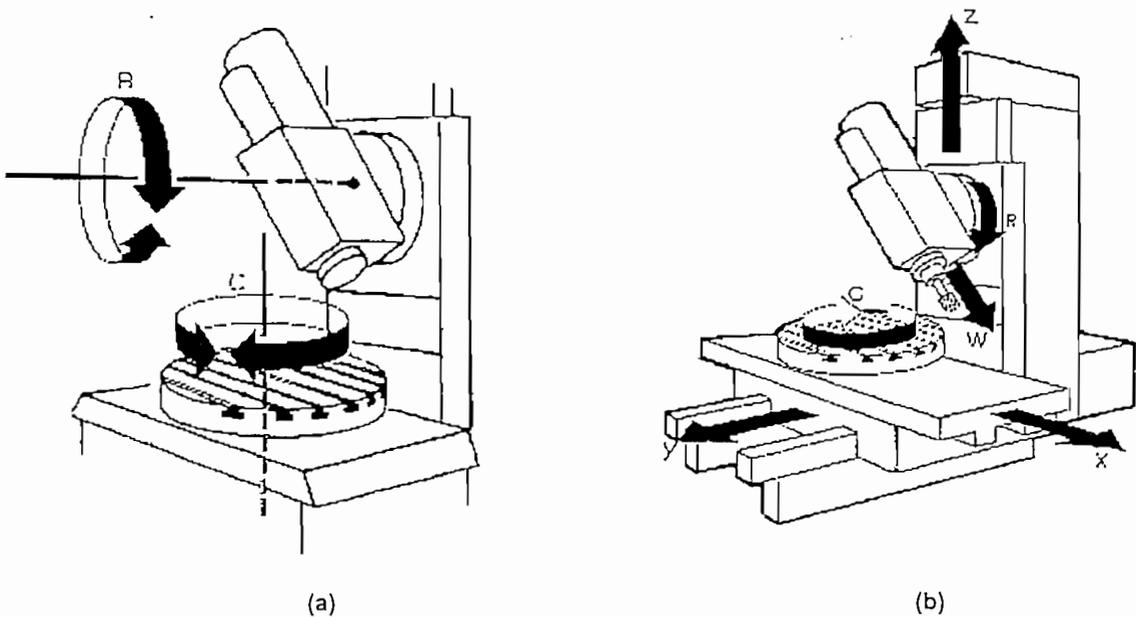


Fig. 1. 10 (a) Mesa giratoria y cabezal basculante; (b) Centro de mecanizado de 6 ejes

1.3.2.4 Desplazamientos Lineales y Circulares.

En un sistema de coordenadas tridimensional los 3 ejes X, Y y Z forman 3 diferentes planos fundamentales. Estos planos se caracterizan por el hecho de que el tercer eje, en cada caso, es perpendicular al plano (Fig. 1.11).

- Plano XY (figura a)
- Plano XZ (figura b)
- Plano YZ (figura c)

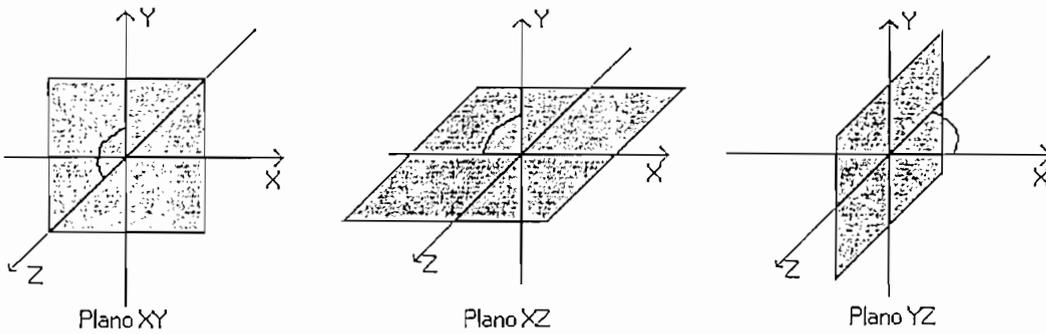


Fig. 1.11 Planos fundamentales

Para ser capaces de describir círculos en un sistema de coordenadas bidimensional se requiere establecer el centro del círculo y un radio (figura 1.12a). Para determinar círculos en un sistema de coordenadas tridimensional es necesario además especificar el plano del círculo (figuras 1.12b y 1.12c).

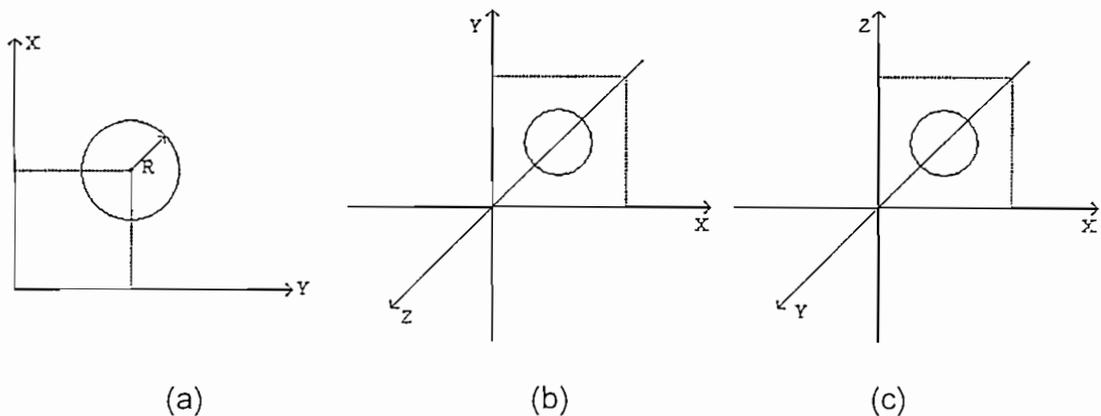


Fig. 1.12 Desplazamientos circulares: (a) en dos dimensiones 2D, (b) en el plano XY y (c) en tres dimensiones 3D.

Medida de los Desplazamientos.

Las posiciones de los elementos móviles de las MHCN se pueden medir mediante los sistemas directo e indirecto:

- El **sistema directo** usa una escala de medida ubicada en la guía de la mesa de la máquina. Las imprecisiones en el giro del sinfín o en su acoplamiento no afectan a éste método; se determina la posición por conteo directo en la regleta graduada y transforma la información a señales eléctricas para procesarla en la UCM (Fig. 1.13).

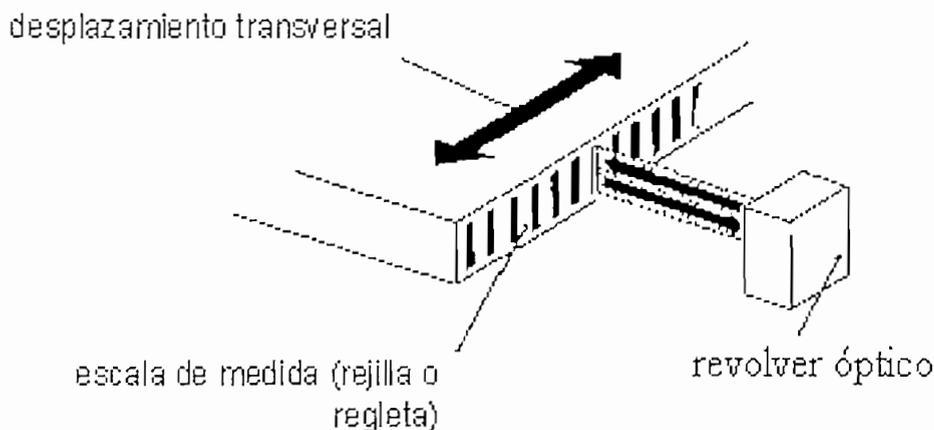


Fig. 1. 13 Sistema directo para la medición de una posición

- En el *sistema indirecto* la posición de la mesa se calcula por la rotación en el sinfín. En donde el revolver registra el movimiento del disco graduado que actúa con el sinfín; con lo que la UCM calcula la posición mediante el número de pasos o pulsos que se generan durante el desplazamiento (Fig. 1.14).

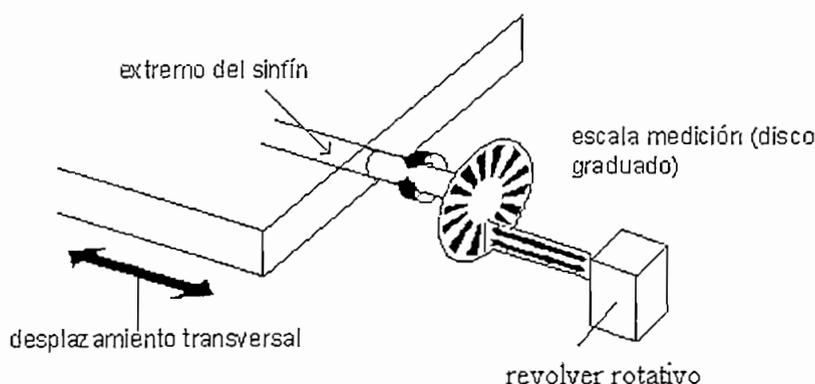


Fig. 1. 14 Sistema indirecto para la medición de una posición

Atendiendo el método y forma de lectura se puede distinguir entre mediciones de posición absoluta y por incrementos.

- **Medición de posición absoluta.**- para la medición de los desplazamientos las posiciones estimadas son independientes del estado puntual de la máquina o de su control al estar referidas a un punto invariante conocido como "origen absoluto".

- **Medición de posición por incrementos.**- se emplea para designar los movimientos relativos a algún punto significativo distinto del origen absoluto y que, además, puede variar. Durante el movimiento la UCM lleva a cabo un conteo del número de incrementos (divisiones) en las que la nueva posición difiere de la anterior.

Es importante que la lectura se la realice en todo el rango de desplazamiento del eje en cuestión. Para garantizar que la medida se realiza correctamente, inmediatamente después de inicializarse la UCM se mide la posición inicial respecto al origen. Tan pronto como la máquina a asignado el punto de referencia comienza a suministrarse posiciones relativas al último punto.

1.3.3 MOVIMIENTO Y CONTROL DE MÁQUINAS HERRAMIENTAS.

Los controles básicos para el movimiento de una máquina herramienta, dependen directamente de la complejidad en su forma de la pieza. Un ejemplo notorio es el maquinado, donde la herramienta de corte sigue una trayectoria exacta para generar así la forma requerida de la pieza. Cuando el movimiento de la pieza de trabajo se restringe a un eje, siendo éste el Z generalmente, se habla de movimiento o control de un solo eje, Fig. 1.15 (a); cuando el movimiento en la dirección Z es sólo de arranque y paro, y además avanza a una cierta velocidad prefijada, se trata del control de dos ejes y medio. El movimiento de la mesa de trabajo requiere el control de dos ejes, refiriéndose usualmente al X y Y, Fig. 1.15 (b); girar la herramienta añadiría los ejes cuarto y quinto, Fig. 1.15 (c).

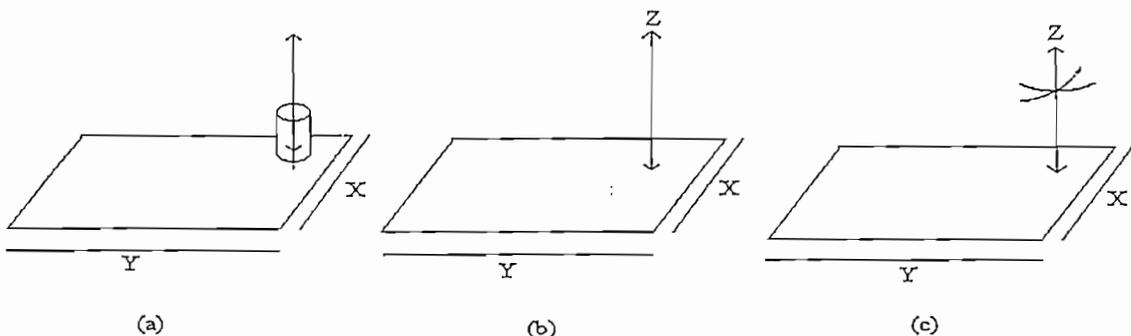


Fig. 1. 15 Las herramientas y piezas de trabajo pueden moverse y el control se puede ejercer a lo largo de (a) uno, dos, (b) tres, o (c) varios ejes.¹²

¹² SCHEY, John A., Procesos de Manufactura , McGraw-Hill, Tercera Edición, pág. 47

Existen algunas características en la forma de la pieza, que establecen ciertas limitaciones para el movimiento:

- La simetría axial es la más sencilla, ya que una forma bidimensional puede generarse girando la parte o la herramienta respecto al eje Z.
- Las piezas con simetría no rotacional exige un mínimo de dos ejes de control, aunque la curvatura espacial puede seguir con tres o más ejes.
- Una superficie paralela al movimiento de la herramienta, Fig. 1.16 (a), se genera con el control de un solo eje, y si la herramienta es difícil de retirar se utiliza un ángulo de salida, Fig. 1.16 (b). Las formas de corte segado Fig. 1.16 (c y d) requieren control en más de un eje, por lo que es común en la actualidad el uso de cinco ejes.

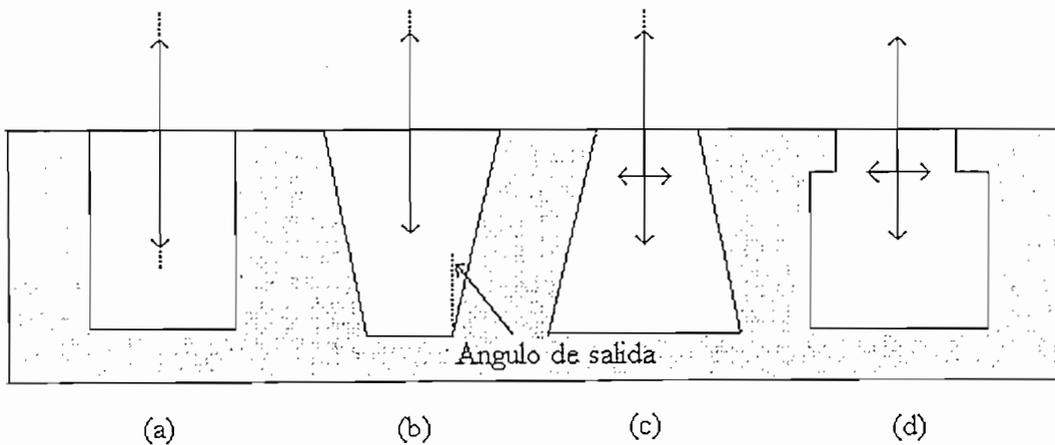


Fig. 1. 16(a) Cavidad de paredes rectas fácil de producir, (b) se puede necesitar un ángulo para retirar la herramienta. (c), (d) Formas segadas requieren control de ejes múltiples o herramientas complejas.¹³

1.3.4 SISTEMA DE REFERENCIA Y CONTROL DE TRAYECTORIA DE LOS CONTROLES NUMÉRICOS.

Para el movimiento de una máquina herramienta, se recomienda establecer un sistema de referencia y un sistema de control de trayectoria para tener cortes ordenados según la herramienta de corte y la forma de la pieza. A continuación se da una breve explicación de los sistemas antes mencionados para tener una idea básica acerca del tema.

¹³ SCHEY, John A., Procesos de Manufactura , McGraw-Hill, Tercera Edición, pág. 48

1.3.4.1 Según el Sistema de Referencia.

Para programar los sistemas de CN se recomienda un sistema de referencia estándar en el que puedan ser especificadas las diferentes posiciones relativas de la máquina herramienta con respecto al trabajo a realizar.

Para facilitar las cosas la pieza a ser maquinada se fija a una mesa de trabajo mientras que la máquina herramienta se mueve en torno a ella. De este modo el sistema de referencia se fija con respecto a la mesa de trabajo.

Sistemas de referencia fijos frente a sistemas de referencia flotantes.

El propósito de los sistemas de referencia es localizar la herramienta en relación con la pieza a ser maquinada. Dependiendo del tipo de máquina de CN el programador puede tener varias opciones para especificar esta localización.

En el caso de sistemas de referencia fijos, el origen siempre se localiza en la misma posición con respecto a la mesa de trabajo. Normalmente, esta posición es la esquina inferior de la izquierda de la mesa de trabajo y todas las posiciones se localizan a lo largo de los ejes XY positivos y relativos a ese punto fijo de referencia.

En el caso de sistemas de referencia flotantes (máquinas modernas de CN), se permite que el operador fije el origen del sistema en cualquier posición de la mesa de trabajo. A esta característica se le llama origen flotante. El programador es el que decide donde debe estar situado el origen. Esta decisión corresponde a la conveniencia de la parte de programación. La localización de esta referencia se realiza al principio de la tarea, donde el operador mueve la herramienta mediante un control manual al punto que se desea como origen del sistema y mediante algún dispositivo, se indica a la máquina la posición del nuevo origen.

1.3.4.2 Según el control de las trayectorias.

Dentro de los controles numéricos, encontramos varios tipos de controles de trayectoria, los mismos que se mencionan a continuación:

Punto a Punto.

El CN punto a punto controla el posicionamiento de la herramienta en los puntos donde debe ser realizada una operación de mecanizado utilizando desplazamientos en vacío según trayectorias paralelas a los ejes o formando un ángulo de 45 grados con los mismos. Se lo utiliza fundamentalmente en máquinas taladradoras, punzonadoras, punteadoras y en algunas mandrinadoras.

La coordinación entre ejes no es necesaria ya que lo importante es alcanzar un punto dado en un mínimo tiempo y con alta precisión. El camino seguido para ir de un punto a otro no importa con tal de que no existan colisiones.

El posicionamiento punto a punto se usa cuando se necesita precisión en la localización del huso, o las piezas de trabajo montadas sobre el tablero, para realizar los procesos de taladrar, perforar, horadar y remachar (Fig. 1.17).

El posicionamiento punto a punto es un proceso de ubicación o localización de una coordenada (XY) a otra, realizando la operación de maquinado, y continuando este modelo hasta completar las operaciones de localización programadas.

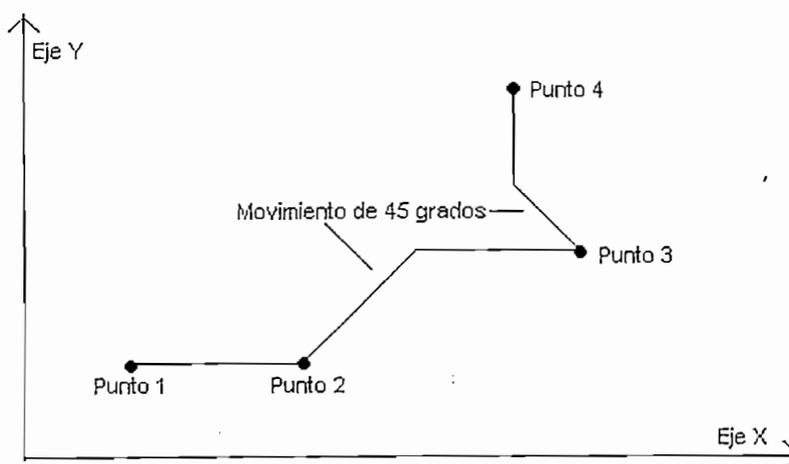


Fig. 1. 17 El camino seguido por el posicionamiento punto a punto para alcanzar varios puntos programados en los ejes XY.

En la Fig. 1.17 el punto 1 se une al punto 2 con una línea recta y los movimientos de la máquina sólo son a lo largo del eje X, pero los puntos 2 y 3 requieren el

movimiento a lo largo de los ejes X y Y. Como la distancia en la dirección de X es mayor que en la dirección de Y, Y alcanzará su posición primero, dejando que X viaje en línea recta por el resto de la distancia. Un movimiento similar tiene lugar entre los puntos 3 y 4.

Paraxial.

El CN paraxial controla la posición y trayectoria en el mecanizado del elemento desplazable, siempre que esta última sea paralela a los ejes de la máquina y, en algunos casos, a 45 grados. Es aplicable a cualquier tipo de máquina herramienta, aunque su uso se reduce al gobierno de taladradoras y fresadoras.

El control paraxial permite desplazamientos rápidos, avance de la herramienta en carga según trayectorias paralelas a los ejes básicos de la MHCN. En dichas trayectorias sólo actúa un único motor (ejecuta el desplazamiento en el eje) controlándose la distancia a recorrer y la velocidad del avance (Fig. 1.18).



Fig. 1. 18 Control paraxial.

De contorneado.

El CN de contorneado fue el primero en aparecer aunque quedó en segundo plano frente a los sistemas punto a punto y paraxiales. Estos sistemas controlan tanto la posición final de la herramienta como el movimiento en cada instante de los ejes, usando técnicas de interpolación lineal, circular y parabólica.

La denominación contorneado o de camino continuo viene dada por su capacidad de control continuo de la trayectoria de la herramienta durante el mecanizado, y la posibilidad de realizar trayectorias definidas matemáticamente, obtenidas por aproximación. Este tipo de control de contorneado se aplica a tornos, fresadoras, centros de mecanizado y a cualquier tipo de máquina que deba realizar mecanizados según una trayectoria más o menos compleja.

La modalidad de camino continuo, involucra el trabajo que se produce en el fresado, donde la herramienta de corte está en contacto con la pieza de trabajo cuando este viaja de un punto a otro. El posicionamiento de camino continuo tiene la habilidad de controlar los movimientos en dos o más ejes de la máquina guardando una constante relación con el cortador y la pieza de trabajo. La programación en CNC debe posicionar con precisión la herramienta de corte que va de un punto a otro, siguiendo un camino exacto y predefinido para producir el contorno requerido (Fig. 1.19).

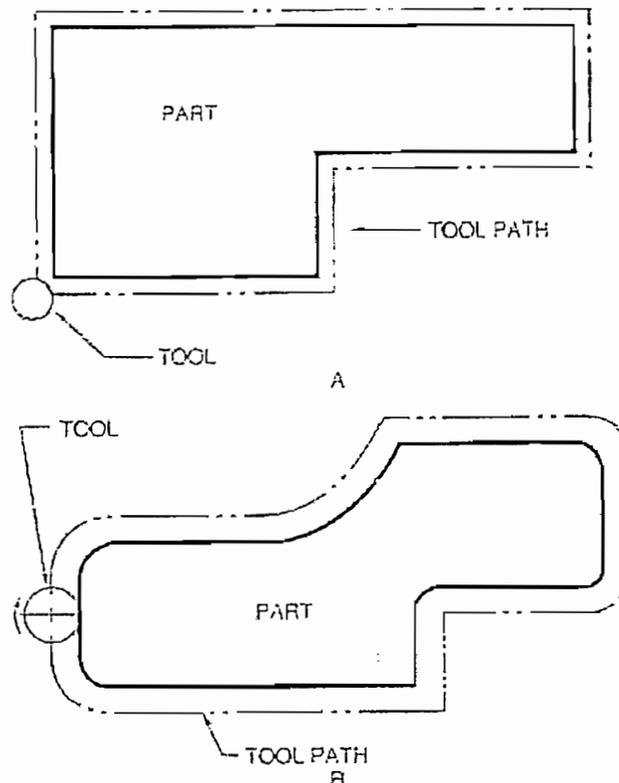


Fig. 1. 19 Tipos de contorneo. (a) Contorno simple y (b) Contorno complejo.¹⁴

¹⁴ KRAR, Steve / GILL, Arthur, Computer Numerical Control Programming Basics, Machinery's Handbook, pág. 20

Interpolación.

En este tipo de trayectoria el sistema CNC calcula un conjunto de posiciones intermedias a lo largo de un segmento recto definido entre dos puntos dados (Fig. 1.20). Durante el desplazamiento de una posición intermedia a otra, los movimientos en cada uno de los ejes afectados se corrigen continuamente de tal manera que la trayectoria no se desvía de la recta prefijada más allá de la tolerancia permitida.

Las curvas pueden ser producidas con la interpolación lineal interrumpiéndose por trazos pequeños de líneas rectas. Este método tiene limitaciones porque es necesario programar un número muy grande de puntos para describir la curva en orden para producir la forma del contorno.

La programación de un contorno con interpolación lineal requiere la coordenada de las posiciones para el inicio y fin de cada segmento de línea. Por consiguiente, el punto extremo de una línea o segmento se convierte en el punto de salida del siguiente segmento, y así sucesivamente, a lo largo de todo el programa.

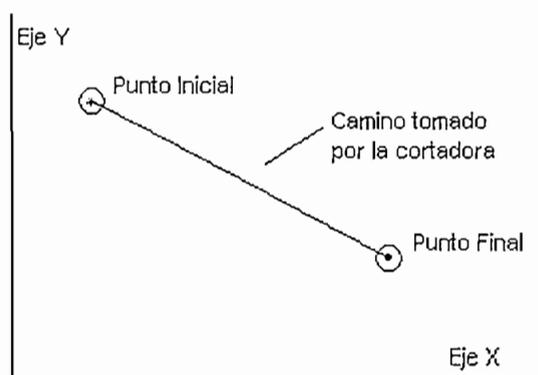


Fig. 1. 20 Un ejemplo de interpolación lineal en dos ejes.¹⁵

Otra alternativa para la curvatura en las piezas es la Interpolación Circular, que tiene la capacidad de simplificar el proceso de programar arcos y círculos (Fig. 1.21).

¹⁵ KRAR, Steve / GILL, Arthur, Computer Numerical Control Programming Basics, Machinery's Handbook, pág. 21

El sistema CNC calcula un conjunto de posiciones intermedias a lo largo del segmento circular definido entre dos puntos dados. Durante el desplazamiento de una posición intermedia a otra, los movimientos en cada uno de los ejes afectados se corrigen continuamente de tal manera que la trayectoria no se desvía de la circunferencia prefijada más allá de la tolerancia permitida.

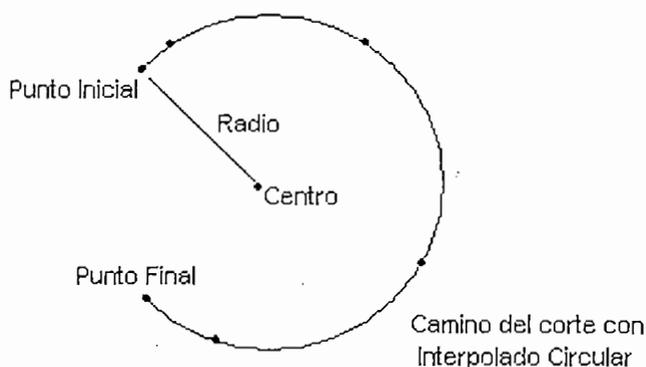


Fig. 1. 21 Para la interpolación circular bidimensional del MCU se suministran los valores del radio, punto de inicio, punto final y dirección de corte.¹⁶

1.3.5 MECANISMOS DE POSICIONAMIENTO.

Los mecanismos de posicionamiento tienen como objeto conducir los dispositivos móviles (carros, husillos, etc.) automáticamente a una posición determinada según una trayectoria especificada con las condiciones de precisión, velocidad y aceleración adecuadas.

Los componentes básicos de los mecanismos de posicionamiento son los accionadores y el propio sistema de control de posicionamiento. Por accionadores se entienden como aquellos dispositivos que permiten realizar algún movimiento (motores, válvulas, etc.), incluyendo aquellos dispositivos asociados de regulación y amplificación de la señal de mando.

¹⁶ KRAR, Steve / GILL, Arthur, Computer Numerical Control Programming Basics, Machinery's Handbook, pág. 22

El control de posicionamiento de una máquina-herramienta de CN puede realizarse generalmente mediante el uso de dos sistemas de servomecanismos de posicionado, el Sistema de bucle cerrado y el Sistema de bucle abierto.

1.3.5.1 Sistema de Bucle Cerrado.

En este tipo de servomecanismos, las órdenes que se dan a los motores llegan de la información proporcionada por la unidad de cálculo del CN y de los datos suministrados por el sistema de medida de la posición real (captador de posición) y de la velocidad real (captador de velocidad) que se encuentran sobre la máquina.

Su principio consiste en comparar en todo momento la posición del móvil con la orden dada. Usualmente se utilizan dos bucles de retorno de información, uno para el control de posición y otro para el control de la velocidad de desplazamiento del móvil, debido a que antes de llegar a la cota deseada se disminuye la velocidad para alcanzar el posicionado correcto.

Generalmente para el control de posicionamiento en bucle cerrado se utilizan motores de corriente continua de imán permanente y de baja inercia debido a su funcionamiento flexible, con aceleraciones rápidas y regulaciones de velocidad proporcionales a la tensión. Actualmente se busca la incorporación de motores de corriente alterna a los sistemas de bucle cerrado por sus mejores prestaciones y menor mantenimiento.

1.3.5.2 Sistema de Bucle Abierto.

En este tipo de sistemas se elimina el retorno de la información de posición y velocidad del móvil. Se utilizan forzosamente motores paso a paso para el movimiento de los ejes, debido a que un motor de este tipo tiene un rotor que efectúa una rotación con un ángulo determinado cada vez que recibe un impulso eléctrico.

El motor a paso realiza un control de desplazamiento y velocidad en forma sencilla; su alimentación se la realiza a través de impulsos eléctricos cuyo número tiene relación con la posición que se desea alcanzar, y su cadencia (número de impulsos por unidad de tiempo) establece la velocidad de giro.

Este tipo de sistemas se utilizan para aquellas máquinas en las que no es necesario controlar en todo momento la velocidad de avance y la posición de la herramienta como es el caso de punteadoras, taladradoras, plegadoras, etc. Los inconvenientes principales que presentan este tipo de motores son:

- La posible pérdida de pasos en el desplazamiento por un esfuerzo elevado en el eje del motor, lo que conduce a un error de posición.
- Limitaciones de potencia y par intrínsecas a las características del motor.
- Debido a su avance producen un peor acabado de las piezas a mecanizar.

1.4 MOTOR PASO A PASO (PAP).

Uno de los elementos de la máquina herramienta NEURACTOR CNC-III utilizado en el presente Proyecto de Titulación es el motor paso a paso; por tal razón, en éste primer capítulo se dará una breve explicación acerca de éste dispositivo, con el objetivo de que el lector entienda su función y sus ventajas en los procesos de automatización en los sistemas de corte.

Los motores en general están compuestos por espiras devanadas sobre un núcleo magnético conocido como inducido, y de un electroimán o inductor que genera el campo magnético. Al hacer girar el inducido en el seno del campo creado por el inductor se obtiene una corriente eléctrica, la misma que genera el giro de su eje obteniéndose con ello un movimiento rotatorio. La velocidad de giro de un motor es el número de revoluciones por minuto (r.p.m.) o por segundo (r.p.s.) a la que gira el eje del motor. Adicionalmente, los motores requieren de escobillas que procuran el contacto con los dos extremos de los bobinados que se

hallan conectados a las dos partes del colector para recibir la tensión externa que se aplica a los extremos de sus bobinados.

Los motores paso a paso tienen su principal aplicación en sistemas de posicionamiento (precisan de un posicionamiento seguro y exacto del eje), como sucede con las impresoras, los brazos electromecánicos, los escáneres, etc.

A éstos motores se los define como conversores electrodinámicos que transforman energía eléctrica (tren de pulsos) en mecánica, generando saltos o pasos del mismo ángulo de giro (desde $1,80^\circ$ hasta 15°). Cada paso representa un desplazamiento angular fijo del eje del motor, donde sus movimientos presentan insensibilidad a las variaciones de tensión y posicionamiento. Los motores paso a paso, se controlan por el cambio de dirección del flujo de corriente a través de las bobinas que lo forman, en la secuencia apropiada; mientras que el ángulo de giro depende del número de pares polares estatores (fases o bobinas) y del número de pares polares rotores.

Una de las ventajas más importantes de éste tipo de motores es que no precisa de sensores (dispositivo que detecta, o sensa manifestaciones de cualidades o fenómenos físicos, como la energía, velocidad, aceleración, tamaño, cantidad, etc.) ni encoder (Codificador o aplicación que convierte formatos entre sí) para determinar la posición del eje, puesto que queda perfectamente fijado por los pasos girados que se producen por los impulsos aplicados a sus bobinados.

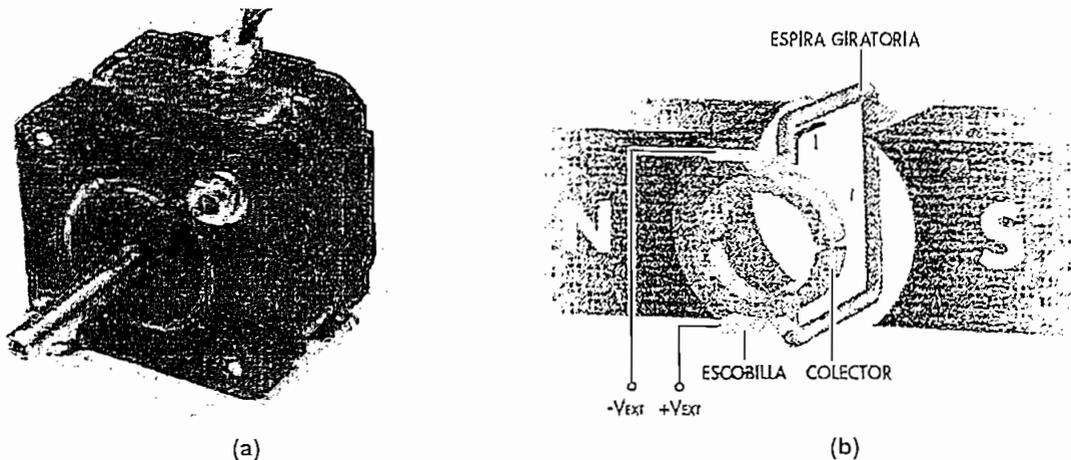


Fig. 1. 22 (a) Motor a paso y (b) La tensión externa aplicada a los extremos de las bobinas del inducido, conectadas a las dos partes del colector, se introduce por las escobillas.

1.4.1 PRINCIPIO DE FUNCIONAMIENTO.

El motor PAP está construido con un estator fijo, formado por varios arrollamientos independientes que se hallan devanados sobre un material ferromagnético (los materiales ferromagnéticos son el Hierro, Niquel, Cobalto, y algunas aleaciones) y en cuyo interior puede girar el rotor (parte giratoria de una máquina electromagnética, turbina o molino de viento).

El voltaje de alimentación se aplica a las bobinas del estator, alcanzando un desplazamiento angular del flujo magnético que obliga al rotor a moverse para ir enfrentando polos de distinto signo (existe mayor fuerza de atracción).

En la siguiente figura, al colocar el conmutador rotatorio en la posición 2, se aplica un voltaje de alimentación a la bobina B2, circulando a través de ésta una corriente I que genera un campo magnético obligando al imán giratorio del rotor a moverse hasta orientarse de forma que queden enfrentados los polos de distinto signo. Si el conmutador cambia de posición, el proceso antes mencionado se repetirá generando un constante movimiento del rotor giratorio conocido como salto o paso y que dependen del número de bobinas que se encuentren en el perímetro circular del estator ($360/\text{número de bobinas}$).

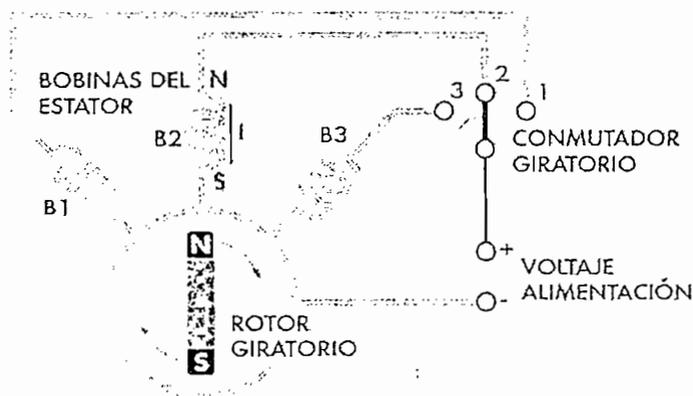


Fig. 1. 23 El conmutador giratorio selecciona la bobina que atravesará la corriente.

Si en lugar de un conmutador mecánico, se aplica a las bobinas del estator una secuencia de impulsos de excitación constante y uniforme a una frecuencia F , el número de r.p.s. del eje del rotor vendrá dado por el cociente entre la frecuencia F

y el número de bobinas en el estator. Por otra parte, el sentido de giro lo determina la dirección de la secuencia de excitación de las bobinas.

1.4.2 TIPOS DE MOTORES A PASO SEGÚN SU CONSTRUCCIÓN.

Desde el punto de vista de su construcción mecánica, los motores de paso se fabrican dentro de 3 categorías:

- Motores a paso de Reluctancia variable (VR)
- Motores a paso de Imán permanente (PM)
- Motores a paso Híbridos (HB)

1.4.2.1 Motores a paso de Reluctancia variable (VR).

Este tipo de motores tienen un multi-dentado de hierro suave y un estator. El rotor no está magnetizado permanentemente por lo que su giro es libre y no tiene torque de frenado. Aunque la relación torque a inercia es buena, el torque para cierto tamaño es limitada, razón por la cuál se los construyen de tamaños pequeños y raramente se los emplea en aplicaciones industriales.

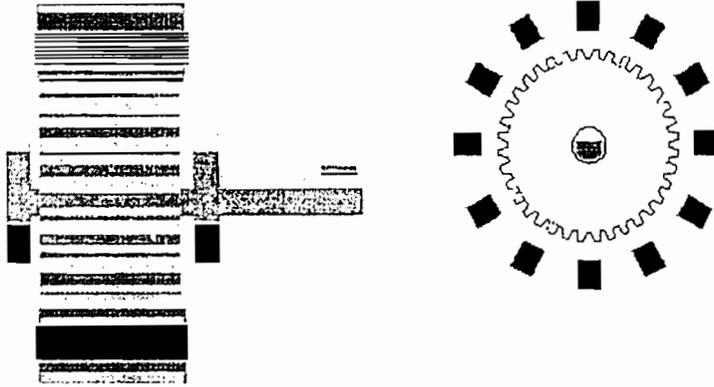
1.4.2.2 Motores a paso de Imán permanente (PM)

Conocidos como motores con polos dentados, y cuya característica es que poseen un rotor que es un imán permanente magnetizado radialmente y un estator como el de los motores de reluctancia variable. Son los que típicamente se usa en las impresoras para mover la rueda impresora.

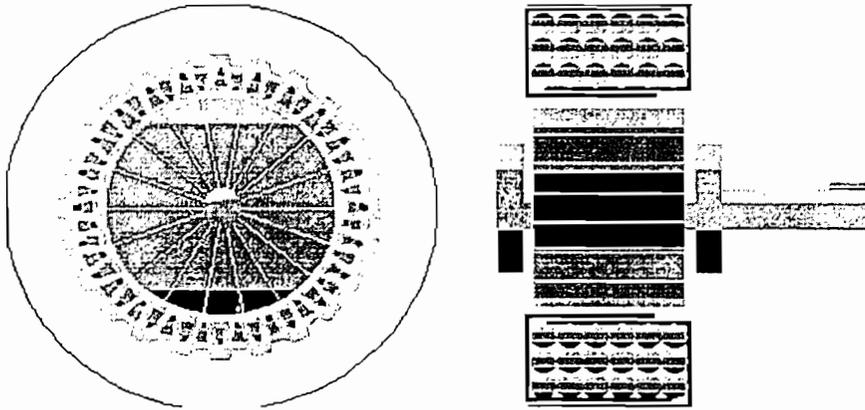
1.4.2.3 Motores a paso Híbridos (HB)

Los motores híbridos son los más utilizados debido a que son una combinación entre los RV y PM. Generalmente poseen un estator dentado con varias fases y un rotor compuesto de tres partes; esto es, dos piezas polares dentadas separadas por un imán permanente magnetizado radialmente. Cada pieza está alineada de tal forma que los dientes de una quedan a medio paso de los dientes de la otra, logrando así que el motor tenga una resolución alta.

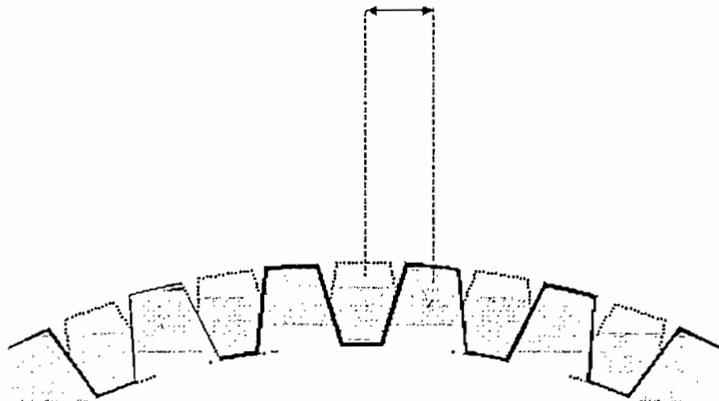
Debido a su gran torque estático y dinámico y que pueden girar a altas velocidades, los motores híbridos se emplean ampliamente en la industria.



(a)



(b)



(c)

Fig. 1. 24 (a) Estructuras mecánicas del Motor de Reluctancia Variable, (b) Motor de Imán Permanente y (c) off-set en los dientes del motor híbrido.

1.4.3 TIPOS DE BOBINADOS.

Según el tipo de bobinado, se pueden clasificar a los motores de la siguiente manera:

- Motores de Reluctancia Variable
- Motores Unipolares
- Motores Bipolares
- Motores Bifilares
- Motores Multifase

Debido a que éste tema es muy extenso y complejo, a continuación se darán las características básicas de los tipos de bobinados.

1.4.3.1 Motores de Reluctancia Variable.

Estos motores se construyen con varias bobinas y con un terminal común a todos los bobinados (el cable común va al positivo de la fuente y los bobinados son energizados secuencialmente).

La figura 1.25a muestra un motor de 30 grados, cuyo rotor está compuesto por dos pares polares o 4 dientes y el estator cuenta con 3 bobinas que forman 6 polos, con cada bobinado envuelto alrededor de dos polos opuestos. Usando un motor de más polos y más dientes en el rotor, es posible la construcción de motores con ángulo del paso más pequeño.

1.4.3.2 Motores Unipolares.

Se llaman motores de paso unipolares a los de imán permanente o híbridos alambrados, que tienen una toma central en cada uno de los bobinados. Las tomas centrales de los bobinados en la práctica se unen al voltaje positivo y se conectan a tierra los dos extremos de cada bobinado alternadamente para evitar la dirección del campo proporcionado a cada bobinado (suelen tener 5 o 6 cables).

Para resoluciones angulares más altas, el rotor debe tener proporcionalmente más polos. En la figura 1.25b, el motor de imán permanente cuenta con 30 grados por paso, pero también los hay de 15 y 7.5 grados por paso (actualmente hay de 1.8 grados por paso); mientras que los motores híbridos se los puede encontrar de 3.8 y 1.8 grados por paso generalmente, habiendo disponibles con resoluciones de hasta 0.72 grados por paso.

1.4.3.3 Motores Bipolares.

Los motores bipolares de imán permanente e híbridos son construidos con el mismo mecanismo que los motores unipolares, pero los dos bobinados se alambran sin tomas centrales; y aunque los motores bipolares son más simples, los circuitos de control necesitan invertir la polaridad de cada bobina.

Con respecto al NEURACTOR CNC-III, cabe mencionar que trabaja con motores bipolares para el movimiento de los ejes (X, Y, Z).

1.4.3.4 Motores Bifilares.

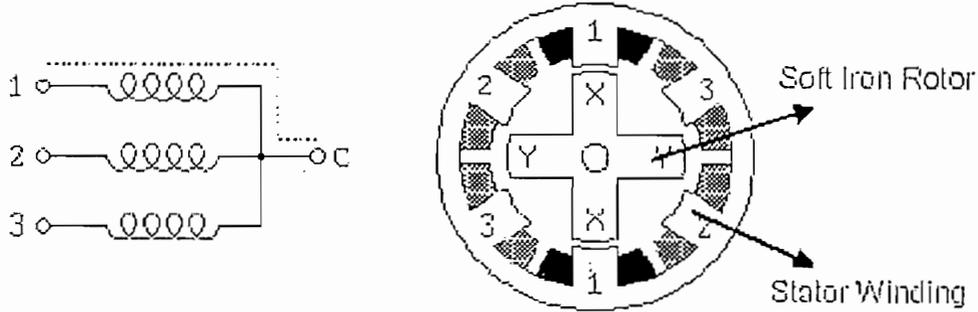
Algunos motores de imán permanente tienen 4 bobinados independientes, organizados en dos juegos de dos.

Dentro de cada juego, si los dos bobinados se alambran en serie se obtiene un motor bipolar de baja corriente y alto voltaje; y si los bobinados se alambran en paralelo se obtiene un motor bipolar de alta corriente y bajo voltaje. Si se alambran las bobinas en serie con una toma central el resultado es un motor unipolar de voltaje bajo.

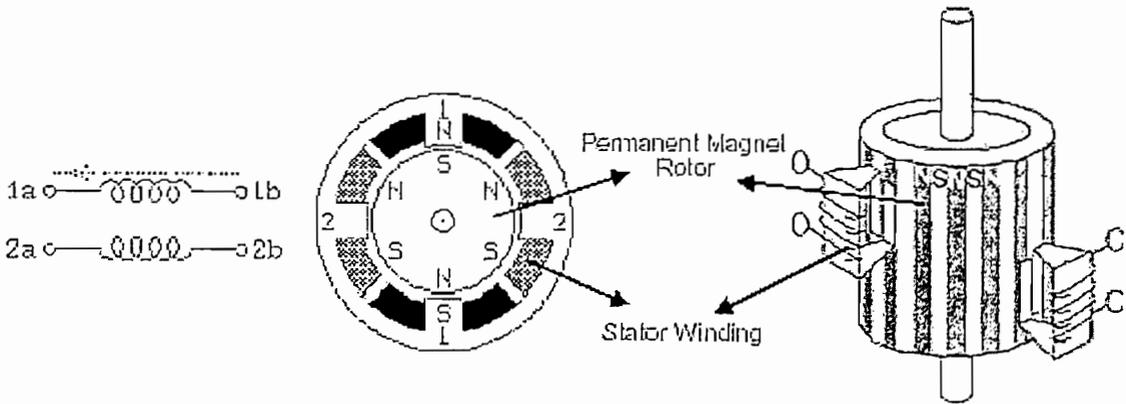
1.4.3.5 Motores Multifase.

Tiene la característica de estar alambrado con todos los bobinados del motor en una serie cíclica (motor de paso de imán permanente), con una toma entre cada par de bobinados en el ciclo. Los diseños más comunes usan cableados de 3 fases y 5 fases, proporcionando más torque para un tamaño de motor dado.

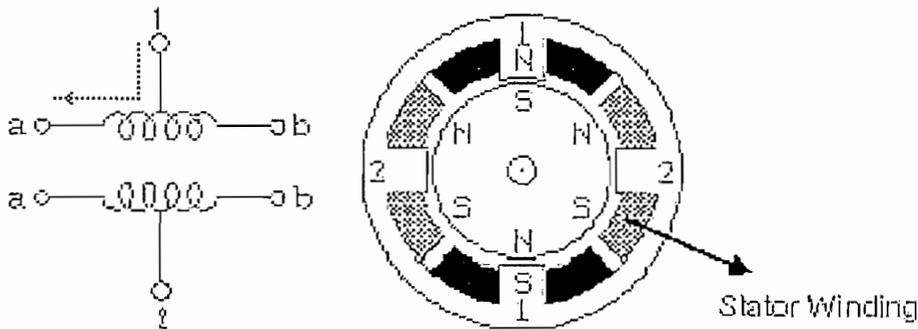
Cada terminal está conectado al bus positivo o negativo de la fuente de poder del motor, donde a cada paso sólo un terminal cambia de polaridad. Este cambio quita la energía del bobinado ligado a ese terminal y aplica energía a un bobinado que estaba previamente desactivado; ésta secuencia de control hará girar el motor dos revoluciones (resoluciones de 0.72 grados por paso equivalente a dar 500 pasos por revolución).



(a)



(b)



(c)

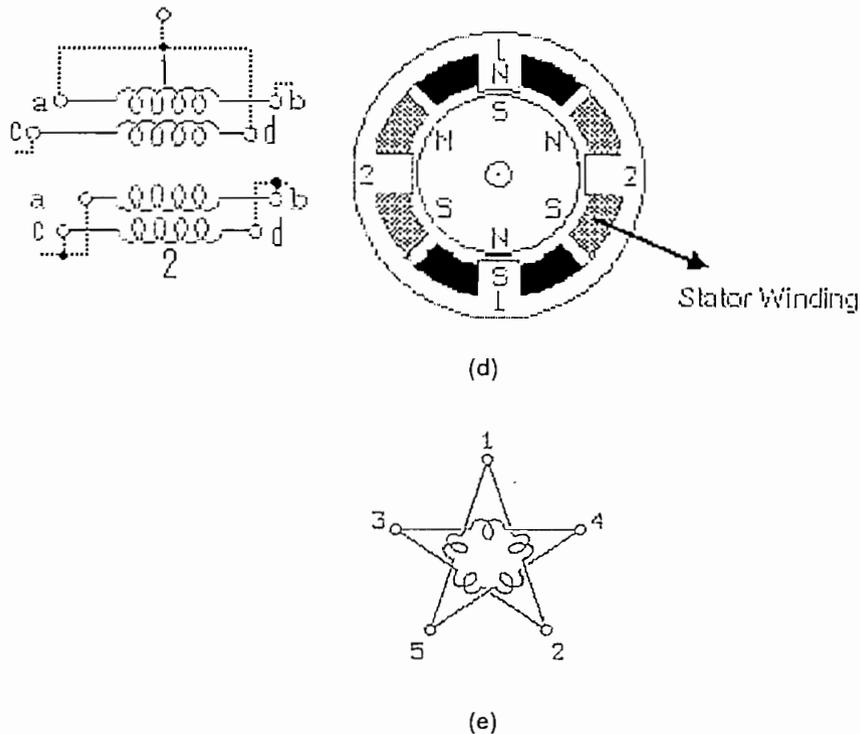


Fig. 1. 25 Bobinado de los motores de reluctancia variable, (b) Motor de imán permanente o híbrido bipolar, (c) Motor de imán permanente o híbrido unipolar, (d) Motor Bifilar y (e) Motor Multifase.

1.4.4 MODO DE TRABAJO.

Los motores a pasos requieren definir un modo de trabajo, es por eso que a continuación se indica su clasificación con una breve explicación que dará al lector una idea general acerca del tema.

- Paso Completo (full step)
- Medio Paso (half step)
- Micropasos

1.4.4.1 Paso Completo.

El modo de trabajo completo se logra energizando alternadamente los bobinados y al mismo tiempo invirtiendo la polaridad del voltaje aplicado. Por ejemplo, los motores de paso híbridos estándar tienen 200 dientes en el rotor; o sea, son capaces de dar 200 pasos para completar una vuelta completa. Esto significa un paso completo de 1.8° .

Generalmente el estímulo que se aplica a las bobinas del estator suele ser un pulso de corriente que puede aplicarse desde un circuito discreto, circuito integrado o PC.

Para el funcionamiento del motor de pasos completo, el controlador debe generar una onda (por olas) como se indica en la figura 1.26a, o aplicando la excitación por dos fases como se indica en la figura 1.26b.

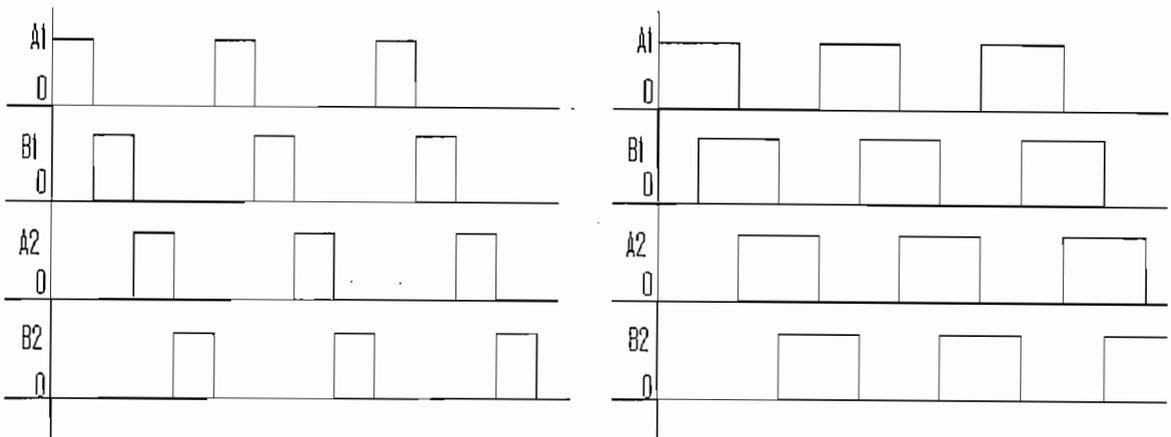


Fig. 1. 26 (a) Manejo del motor de pasos por olas, (b) Manejo del motor de pasos por dos fases.

1.4.4.2 Medio Paso.

Primero se excita a un bobinado y luego a los otros dos alternadamente, forzando al rotor a dar medios pasos. Tomando el mismo ejemplo anterior, el motor rotará 400 pasos por revolución o 0.9° por cada medio paso. Aunque se puede tener la misma resolución con un motor de 400 pasos por revolución, en aplicaciones industriales, se prefiere recurrir al modo de medio paso, debido a que el torque se reduce ligeramente así como el número de saltos.

1.4.4.3 Micropasos.

Considerada como una nueva tecnología para controlar motores de pasos; y consiste en controlar la corriente que alimenta a los bobinados de tal forma que se puedan obtener voltajes y campos magnéticos, de tal magnitud que pueden orientar al rotor en un cierto número de posiciones por paso.

Por ejemplo los motores de micropasos de AMS son capaces de rotar a $1/256$ de paso, es decir, 50000 pasos por revolución.

1.5 SISTEMAS DE CORTE.

La primera impresión que se recibe de un producto o pieza fabricada es su forma y tamaño. Estos dos aspectos tienen relación con la estética, por lo que la tarea del operador es la de crear un producto agradable. Pero, el operador también debe considerar los aspectos forma y dimensiones de las piezas debido a que en el proceso de ensamblaje se requiere que las partes encajen una con otra, y para ello se trabaja con técnicas y procedimientos de medición para obtener desviaciones permisibles (tolerancias dimensionales) específicas.

Existen varias empresas en el mercado que se especializan en el diseño, fabricación y producción de piezas; utilizando equipos que cuentan con elementos de alta precisión, rigidez, calidad y rapidez para cortar, primordialmente madera, metal y plástico con brocas router, fresadoras, sierras circulares, cuchillas, ruedas de diamante con banda polimérica, banda vitrificada, banda metálica, etc.

La secuencia de maquinar un componente es un proceso de ingeniería que se extiende a toda la secuencia de manufactura, planeando con esto la secuencia de los pasos de maquinado. Esta planeación considera la maquinabilidad del material de la pieza de trabajo, la forma, la dimensión, las tolerancias dimensionales y los aspectos económicos de la producción.

Dentro de los sistemas de corte, los factores control numérico, sistema de posicionamiento y motores a paso, son los componentes principales para definir la forma de las piezas, movimiento, control de máquinas herramientas, etc.

1.6 SISTEMAS DE TRANSMISIÓN.

Haciendo referencia a la parte mecánica, los recorridos de la herramienta se originan por la acción combinada de los desplazamientos en cada uno de sus ejes principales. Los sistemas de transmisión producen traslaciones rectilíneas en los ejes a partir del giro básico generado por el grupo del motor-reductor, con lo cual se consiguen movimientos suaves y estables; pero sobretodo la capacidad de reaccionar rápidamente en las aceleraciones y deceleraciones.

Adicionalmente, se recomienda no sobrecargar los motores para evitar posibles problemas con el equipo y con la pieza; siendo las posibles causas de la sobrecarga el uso de herramientas inadecuadas, restricciones anómalas en el movimiento y fuerzas de inercia excesivas durante el frenado o aceleración.

En las MHCN más simples con prestaciones basadas en la precisión del mecanizado se utilizan los motores paso a paso como actuadores primarios. Con motores de este tipo, el giro se subdivide en incrementos fijos que son controlados mediante un número de pulsos dado. Sin embargo cuando se desean trabajos pesados de mecanizado con pares resistentes elevados durante el frenado o aceleración, su fiabilidad y prestaciones disminuye.

Con respecto a la transmisión de datos de la computadora hacia la máquina herramienta, se presentan dos tipos de transmisiones serial y paralela, que a continuación se explicará en forma de resumen sus características principales.

1.6.1 TRANSMISIÓN SERIAL.

Es aquel modo de transmisión en que los bits que forman parte de un carácter son enviados uno a continuación de otro, a una determinada frecuencia de reloj y a través de una sola vía de transmisión. De manera que una de las ventajas fundamentales en éste modo es la de minimizar los costos en la transmisión.

Generalmente se utiliza para comunicaciones de larga distancia, con lo cual su velocidad de transmisión disminuye debido a que cada carácter que se envía

debe ser desmembrado bit a bit por el transmisor y luego reconstituido por el receptor; esto se convierte en una desventaja en el proceso de transmisión. La mayoría de los sistemas de comunicaciones, debido a que realizan comunicaciones a gran distancia, son del tipo serial.

El equipo NEURACTOR CNC-III aunque está diseñado para trabajar con transmisión paralela, puede a través de ciertos integrados modificar el tipo de transmisión de datos y trabajar con comunicación serial; en el capítulo dos se hará una descripción más profunda de los integrados antes mencionados.

Según la facilidad del operador se escogerá el tipo de transmisión, ya que como se presenta en éste Proyecto de Titulación, al trabajar con el Software Visual Basic 3.0 se recomienda la comunicación serial, mientras que con el Sistema Visual Basic 6.0 se facilita el uso de la transmisión paralela.

1.6.2 TRANSMISIÓN PARALELA.

En éste modo de transmisión todos los bits que conforman un carácter se transmiten simultáneamente, por tal razón se requieren tantas vías de comunicación como bits tiene el carácter.

La ventaja de éste modo de comunicación es que se trabaja con velocidades de transmisión altas; sin embargo, los costos de transmisión se incrementan debido a que su infraestructura de trabajo es más compleja.

Esta forma de comunicación es adecuada para distancias cortas, como por ejemplo entre el computador y la impresora; adicionalmente el sincronismo en éste sistema complica su implementación práctica.

Generalmente las comunicaciones en paralelo utilizan 8 vías para enviar caracteres de ocho bits de datos (byte) simultáneamente.

CAPÍTULO 2

**DESCRIPCIÓN DEL SISTEMA
MECÁNICO NEURACTOR CNC-III.**

CAPÍTULO 2.

DESCRIPCIÓN DEL SISTEMA MECÁNICO NEURACTOR CNC-III.

2.1 INTRODUCCIÓN.

Con el desarrollo de la manufactura aparecen procesos de fabricación de bienes y artículos realizados a mano cuyo grado de precisión era bajo y que requería una gran inversión de tiempo. La necesidad de crear piezas con mayor precisión, rapidez, seguridad y bajos costos de producción ha llevado a investigadores y operadores a diseñar sistemas y componentes que permiten automatizar éstos procesos de manufactura.

El NEURACTOR CNC-III es uno de éstos dispositivos electromecánicos que aparecen con el desarrollo de la automatización, diseñado para la producción de figuras exactas, perforación de agujeros con variación de profundidades, aplicación de spray o pegamento con movimientos en tres dimensiones, cortado de materiales con diferentes tipos de herramientas, corte de artículos de cuero o tejidos siguiendo un patrón, estampado de hojas de metal, automatización del movimiento de una fotografía en un stand, control de movimiento para fotografía y otros tipos de aplicaciones que se los realiza con alta exactitud y rapidez.

Basado en los objetivos planteados en éste plan de tesis, y aprovechando las ventajas que proporciona el equipo, se desarrollará un sistema que interactuará con el NEURACTOR CNC-III para cortar figuras según patrones diseñados y artículos hechos en espumaflex o fomex.

A continuación se muestran las principales funciones de control de movimiento para el NEURACTOR CNC-III:

- Velocidad de pasos programable desde 20 pasos/seg. a 27.000 pasos/seg.
- Parámetros individuales para velocidad inicial, velocidad final y aceleración.
- Control interno de la posición durante todo el movimiento.

- Movimientos absolutos dentro del rango de 16 Millones de pasos.
- Movimientos continuos que permiten aceleraciones desde la velocidad inicial a la velocidad final, seguida de movimientos indefinidos.

2.2 REQUERIMIENTOS DEL SISTEMA.

El sistema de corte requiere de una correcta comunicación entre el computador y el equipo NEURACTOR CNC-III; por tal razón es necesario cumplir con ciertos requerimientos para obtener movimientos precisos en el proceso de corte de las piezas. A continuación se dará una descripción breve de los requerimientos, y más adelante en éste mismo capítulo se dará una explicación más amplia de los puntos más importantes.

Uno de éstos requerimientos consiste en el desarrollo del sistema mecánico (correcto armado de la máquina herramienta) utilizado para el proceso de corte, con lo cuál se consiguen movimientos precisos generados por los motores a pasos, desplazando a la herramienta de corte en forma confiable y segura a través de la trayectoria designada por el operador. Con respecto al manejo de los motores a pasos US-Cyberlab, se utiliza el microcontrolador CY545 de CYBERNETIC MICRO SYSTEM, que permite manejar motores de pasos de diferentes fases y en diferentes condiciones, poseen una interfase serial y una paralela, y un juego de señales disponibles para manejo del usuario.

Con respecto a la comunicación entre el equipo y el computador, se utilizan microcontroladores de Comunicación de CYBERNETIC MICRO SYSTEM CY-233, que tienen la característica de enviar o recibir la información al computador en forma serial y la retransmite en forma paralela a los controladores CY545. Cabe recalcar que el computador mantiene una comunicación serial con el microcontrolador CY233, y éste retransmite la información en forma paralela al microcontrolador CY545, el mismo que a través de sus comandos controla los movimientos del NEURACTOR.

Con el cumplimiento de los requerimientos más importantes ya mencionados, se procede con el desarrollo de un programa amigable y de fácil uso para el usuario;

en éste caso se trabaja con el lenguaje VISUAL BASIC desarrollado por MICROSOFT, que permite trabajar enviando y recibiendo información.

2.3 DISEÑO DEL SISTEMA MECÁNICO.

La máquina herramienta de corte NEURACTOR CNC-III usa en su diseño cuatro MOTORES NEURACTORS idénticos (dos para el eje Y, uno para el eje X y el último para el eje Z) que consiguen exactitudes de 0.0005" por paso; con respecto a la comunicación, se puede decir que el software CNC-III usa un esquema de protocolo paralelo, que transmite la información mediante la "tarjeta de interfase".

Aunque el equipo está diseñado para mover la parte mecánica en los tres ejes (X, Y, Z), es importante indicar que se elaborará un programa en VISUAL BASIC que permitirá realizar cortes bidimensionales a diferentes tipos de materiales, como espumaflex y fomix. A continuación se muestra el esquema del NEURACTOR CNC-III (Fig. 2.1).

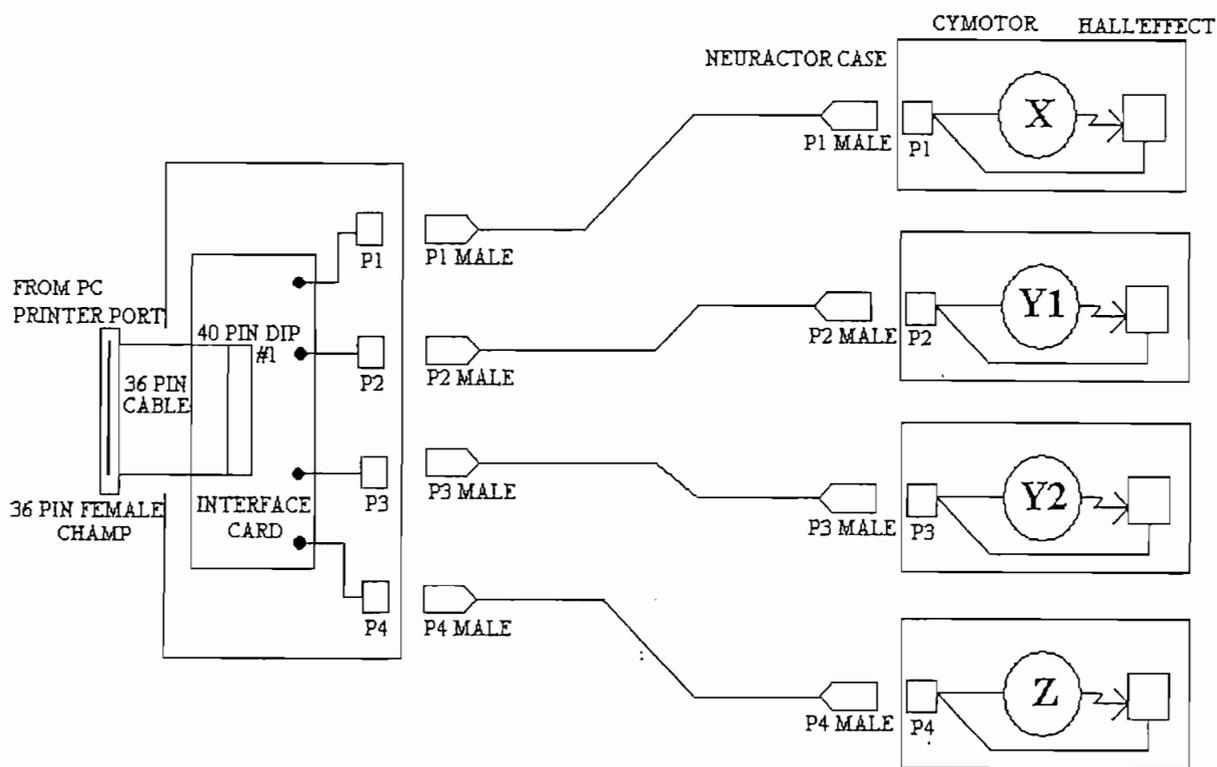


Fig. 2. 1 Esquema del NEURACTOR CNC-III.¹

¹ U.S. CYBERLAB, INC., Assembly and Operation Manual, pág. 2

En la figura 2.1 el ingreso de la información transmitida por el computador se transmite por el cable de 36 pines (la información entregada por el computador es transmitida serialmente hacia el controlador de redes CY-233 que se explicará más adelante, el mismo que retransmite la información al controlador de motor de pasos CY-545 en forma paralela y ésta a su vez se la envía a través del cable de 36 pines) a la tarjeta de interfaz que direcciona las órdenes a sus respectivos ejes (X y/o Y), para generar el movimiento de los motores a pasos (el eje X se encuentra provisto por un solo motor a pasos, mientras que el eje Y requiere de dos motores a pasos debido a que el eje Y se encuentra subdividido en los ejes Y1 y Y2); adicionalmente, las señales entregadas por la tarjeta de interfaz a los motores, también ingresan a las tarjetas de efecto Hall que se hallan junto a cada motor, y sensor continuamente si los ejes han llegado al origen (se detienen los ejes).

El objetivo de éste sistema mecánico es lograr que la herramienta de corte se desplace en forma rápida, segura y con un alto grado de precisión en cualquiera de los tres ejes, (principalmente en los ejes X y Y para éste sistema de corte). Para los ejes X y Y los diseños son similares, con respecto al eje Z la única diferencia es su longitud (menor longitud en el eje Z, aunque sus partes son idénticas).

2.3.1 CONSTRUCCIÓN DEL SISTEMA MECÁNICO NEURACTOR CNC-III.

Con respecto a la construcción del equipo NEURACTOR CNC-III, es necesario indicar, que el equipo fue entregado completamente armado, por lo que no es necesario profundizar en el tema.

Pero, debido a que todo equipo utilizado en el campo de la industria, como es el caso de los procesos de automatización (desarrollado en el actual Proyecto de Titulación), es necesario un mantenimiento periódico del equipo, para asegurar una buena calidad en la producción, y lograr que el tiempo de vida útil del equipo se prolongue; por tal razón, se ha creído conveniente dar una guía acerca de la construcción del NEURACTOR CNC-III, para que los usuarios puedan dar un mantenimiento continuo con el fin de tener en buen estado al equipo.

Ésta información, acerca de la construcción del Sistema Mecánico, se la podrá encontrar en los anexos propios del Capítulos 2, que se encuentran al final del presente Proyecto de Titulación.

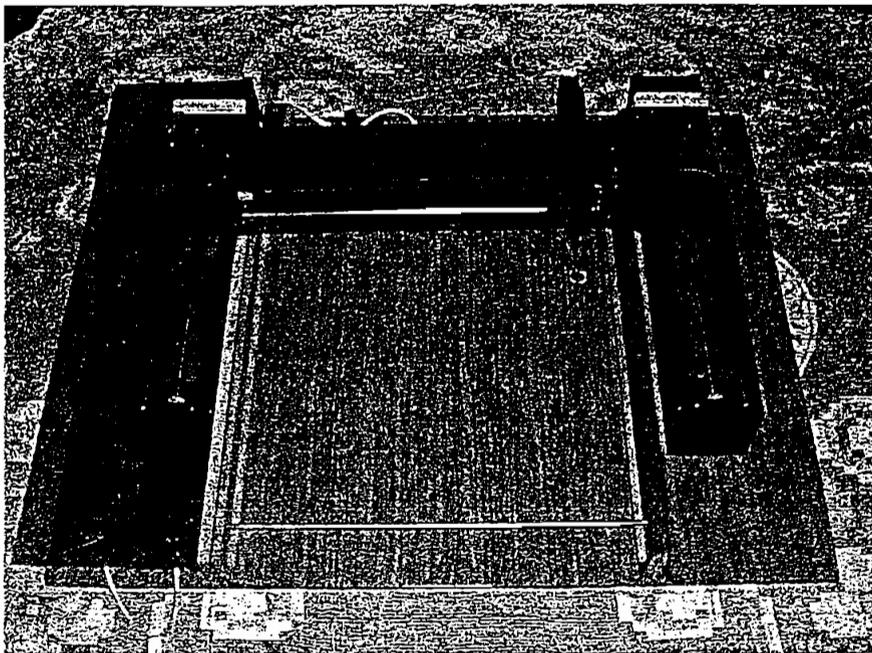


Fig. 2. 2 NEURACTOR CNC-III.

2.3.2 FUENTE DE PODER.

Para el uso del sistema mecánico NEURACTOR CNC-III, se requieren de dos voltajes, +5 Voltios para todos los circuitos integrados (incluido el encendido y apagado del cortador), incluido el display, y una fuente de +12 Voltios para los motores a paso. La fuente de +12 Voltios suministra energía a los motores de paso USCYBERLAB; los cuales necesitan de 1 Amperio y +12 Voltios por fase; y en total se requieren 4 Amperios a +12 Voltios (por contar con 4 motores).

Debido a que el actual Proyecto de Titulación cuenta con todos los elementos que conforman al NEURACTOR CNC-III, no fue necesario la construcción de la fuente de poder; a pesar de esto, se cree importante mencionar sus características de energía, ya que sin éste requisito no se podría hacer funcionar cada uno de los dispositivos con los que cuenta el sistema mecánico.

Se adquirió en la compañía norteamericana JACOME una fuente de poder Switching de 73 Watts de marca Power General, cuyas características son:

Entrada: 115/230 VAC a 47-450 Hz
Salida: +5 VDC a 5A / +12 VDC a 4A
Tamaño: 6.5" Largo x 4" Ancho x 1.75" Altura
Peso: 1.5 lbs.

En la siguiente figura (Fig. 2.3), se muestran los detalles, elementos y conexión de la fuente de poder para un mayor entendimiento.

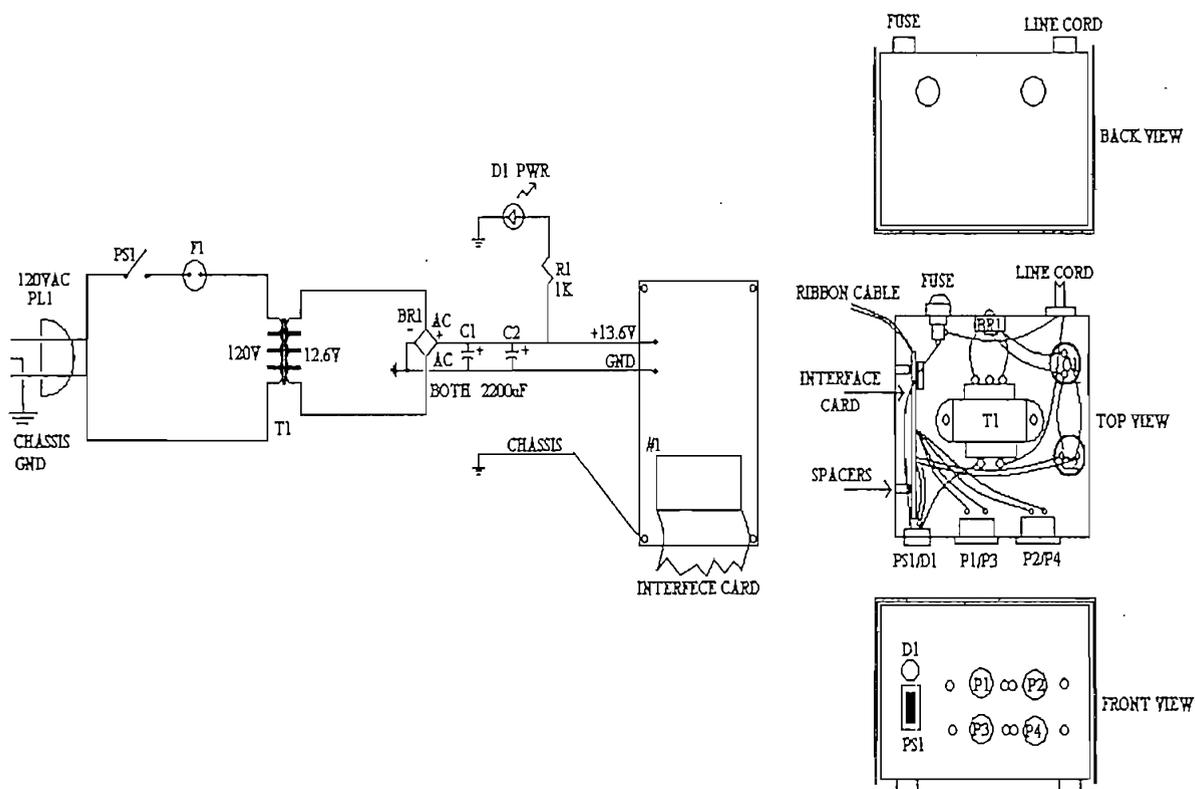


Fig. 2.3 Detalles de la Fuente de Poder.²

2.3.3 FUNCIÓN DE LA TARJETA DE EFECTO DE CAMPO (HALL-EFFECT).

Se han desarrollado cuatro placas que contienen el sistema de Efecto de Campo (Hall-Effect), que tiene la finalidad de sensar la posición HOME de cada uno de los motores, así como de brindar un nivel básico de seguridad.

Diseño de Seguridad y Protección.

² U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 6

Existe un nivel básico de seguridad en el sistema mecánico controlado por la tarjeta de efecto de campo y que consiste en el control de la posición HOME (inicial), disponible en cada uno de los motores de paso.

El control de ésta posición se lo realiza a través de una tarjeta que permite a los motores de pasos conocer cuando el bloque deslizante se encuentra en su posición inicial. Por tal razón, se colocan éstas tarjetas en los postes deslizantes destinados para el efecto Hall que se encuentran al frente de los motores y en el camino del bloque deslizante.

Ésta tarjeta está compuesta por un regulador de voltaje 7805 que convierte el voltaje de 12 Voltios entregado por la fuente a 5 Voltios, una resistencia de 4.7 Kohmios y un transistor de efecto de campo N119 (Fig. 2.4), que sensa el campo magnético del imán que se encuentra en el bloque deslizante.

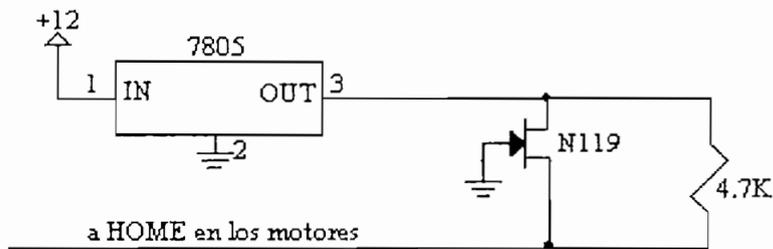


Fig. 2. 4 Tarjeta Hall Effect.³

Como se puede observar en la figura, la salida de 5 Voltios del regulador se ha conectado a la entrada Source del transistor de efecto de campo. Gate se conecta a la tierra del sistema y Drain ingresa a Home del motor de pasos. Adicionalmente, se coloca una resistencia de 4.7 Kohmios entre Source y Drain. Con respecto al bloque deslizante, se coloca un pequeño magneto, la posición del magneto debe ser la correcta, puesto que uno de los lados produce efecto en el transistor, y el otro no.

³ CORDOVA FRANCISCO, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos, pág. 103

Mientras se encuentre lejos el magneto (transistor abierto), la salida del transistor estará en 1L, cuando el magneto se encuentre cerca (bloque deslizante), el transistor se cierra y la salida se coloca en 0L, lo que indica que el bloque llegó a su posición inicial. Una vez que el motor ha detectado el cambio de voltaje en la señal HOME, detiene el motor (el motor podrá moverse en dirección contraria).

En el caso del eje Y, al disponer de dos motores, la salida HOME de las tarjetas ingresa a la entrada de una compuerta OR, para que sólo cuando las dos entradas sean 0L la salida sea 1L.

2.3.4 TARJETA DE INTERFACE.

La tarjeta de interfaz conecta los cuatro motores directamente al pórtico paralelo de la PC de la computadora. De igual forma, ésta tarjeta suministrará a cada motor CYMOTORS +12 voltios regulados. La tarjeta está compuesta por los siguientes elementos:

- 4 Resistencias de potencia de 3,9 Ohmios de 10Watts
- 1 Resistencia de 470 Ohmios
- diodos 1N4148
- Reguladores de voltaje 7812 de 1 Amperio
- Cable multipar de 12 pares
- Placa metálica

La tarjeta de interfaz, fue adquirida con el equipo NEURACTOR CNC-III, así que no fue necesaria su construcción; a pesar de esto, a continuación se muestra el esquema de la tarjeta (Fig. 2.5), con el objetivo que el lector tenga un conocimiento para el montaje de sus elementos para su construcción:

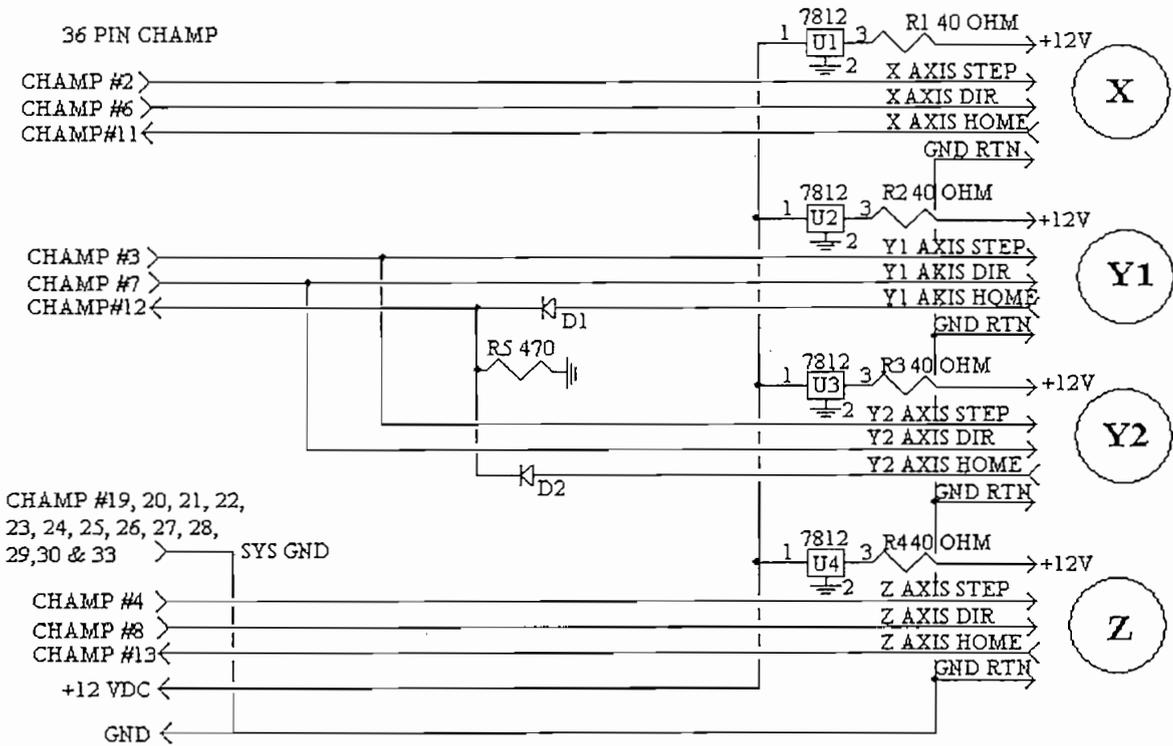


Fig. 2. 5 Esquema de la Tarjeta de Interfase.⁴

La tarjeta de interfase requiere un cable multipar que transporte la información para la interacción entre el computador y el NEURACTOR CNC-III (Fig. 2.6).

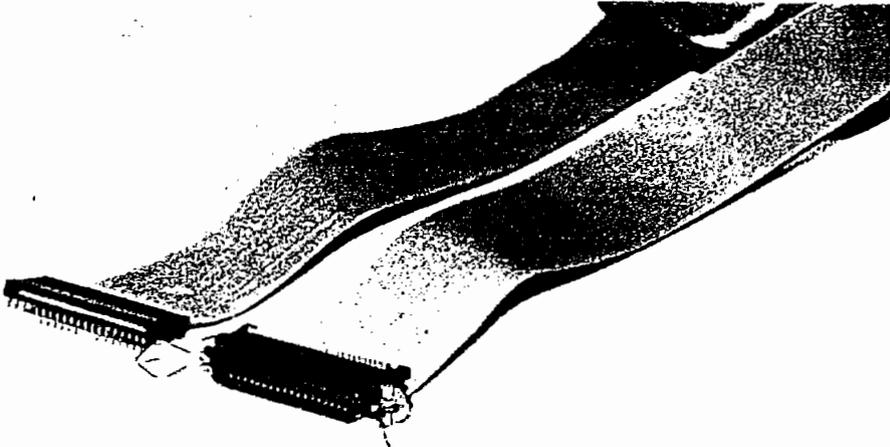


Fig. 2. 6 Cable de la tarjeta de Interfase para la transmisión paralela.⁵

De uno de los lados del cable multipar, se retiran cuatro hilos completos del cable, mientras que el otro extremo del cable de interconexión es unido a la tarjeta de interfase usando 40 pines IDC en la cabecera del conector. Al finalizar todo el

⁴ U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 10

⁵ U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 15

procedimiento, se procede a soldar su cabecera de 40 pines a la tarjeta de interfaz, como se indica en la figura 2.7.

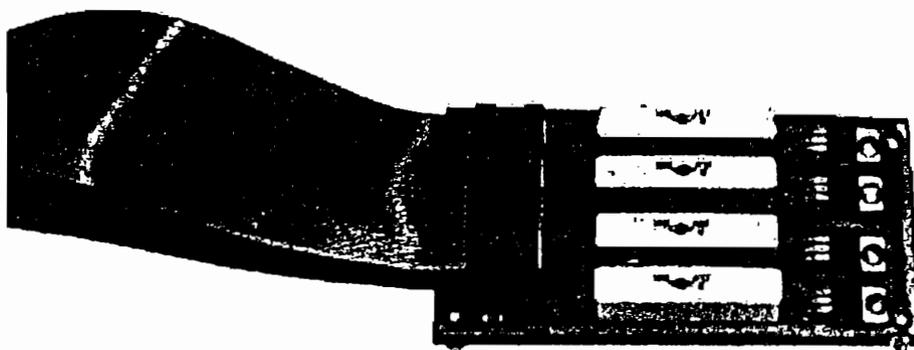


Fig. 2.7 Tarjeta de Interfase completa y conectada al cable para la transmisión paralela.⁶

Con el cable y la tarjeta de interfase completadas, se procede a la conexión de los mismos con los motores CY-MOTOR que utiliza el sistema mecánico NEURACTOR CNC-III.

La figura 2.8 muestra el esquema de la tarjeta de interfase con los motores CY-MOTORS:

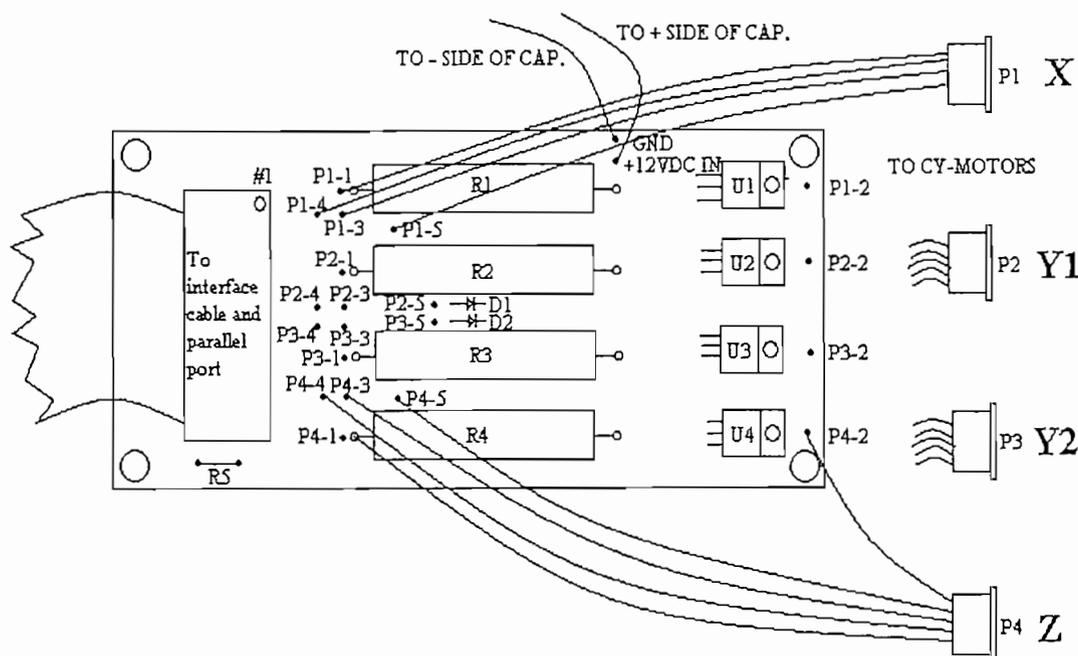


Fig.

2. 8 Esquema de la tarjeta de interfase con los motores CY-MOTORS.⁷

⁶ U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 15

⁷ U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 14

2.3.5 PANEL DE CONTROL.

Las dos primeras placas, contienen a los microcontroladores CY545 y CY233 usados para la comunicación; mientras que las otras dos placas corresponden a la fuente y a los circuitos de potencia para los motores de paso. Los circuitos de las tarjetas se encuentran intercomunicados a través de un cable paralelo.

Con ayuda de LED's que forman parte de las tarjetas que contienen a los microcontroladores se pueden observar los niveles de las principales señales que maneja el CY545; adicionalmente, se encuentran los switches que permiten mover manualmente los motores a pasos. Las tarjetas están compuestas por 3 switches. El sw1 se lo coloca en la posición que indica que se va a trabajar con el microcontrolador CY233, el sw2 de la primera tarjeta se la coloca con la dirección 0100000, mientras que el de la segunda tarjeta se la coloca en la posición 0010000, con ello se asegura el movimiento del eje X y Z con la tarjeta principal y el eje Y con la segunda tarjeta; por último el sw3 se coloca en la posición 0000001 en ambas tarjetas, con lo cuál se podrá visualizar los led's de las tarjetas que nos da información acerca de su comportamiento. Además, en éstas tarjetas se encuentran los conectores seriales para la comunicación con el computador.

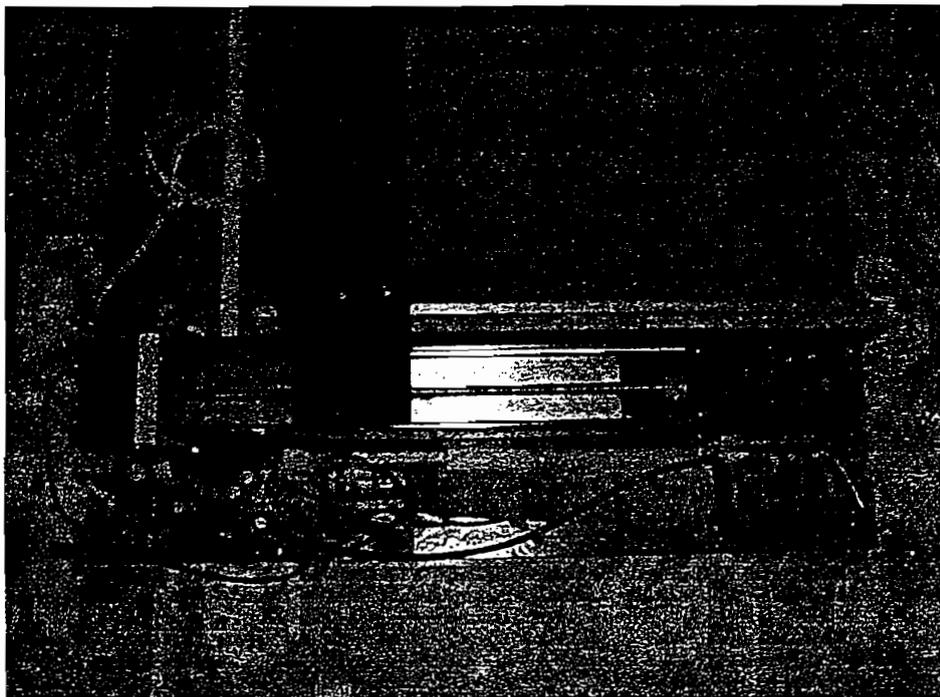


Fig. 2. 9 Panel de Control.

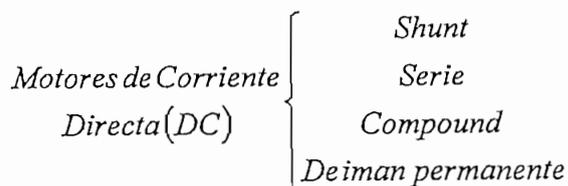
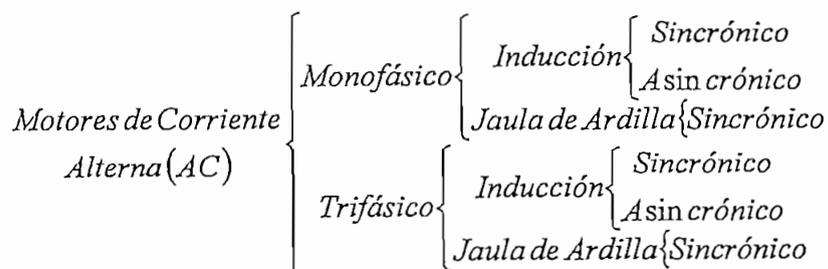
Las otras dos tarjetas, como se mencionó anteriormente corresponden a la fuente de poder y al sistema de potencia que controla los motores de paso con sus respectivos conectores tipo DIN⁸, que se encuentran en cada uno de los motores.

2.4 MOTORES ELÉCTRICOS.

Los motores eléctricos son máquinas que transforman a la energía eléctrica en mecánica (trabajando como motor), o bien de energía mecánica en eléctrica (trabajando como generador) por acción de un campo magnético.

En el campo de la industria, el uso de motores eléctricos ha ido en aumento, debido a su variedad, costos relativamente bajos, facilidad de transporte, simplicidad de la puesta en marcha, buena adaptación a los diversos tipos de carga, etc.

De acuerdo a la fuente de tensión que alimente al motor, podemos realizar la siguiente clasificación:



Motores Paso a Paso



⁸ Conector de clavijas de conexión múltiples que cumple la especificación de la Organización Nacional de Normalización Alemana (DIN, acrónimo de Deutsche Industrie Norm).

En el presente ítem, se dará un breve análisis acerca de los motores utilizados en el área del Control Numérico; por tal razón, aunque la clasificación de los motores es amplia, sólo se analizarán tres tipos de motores que son utilizados en aplicaciones referidas al actual Proyecto de Titulación: Motores Universales, Motores a Pasos y Motores DC en serie.

2.4.1 MOTORES DE CORRIENTE DIRECTA (DC).

Aún cuando toda la energía eléctrica producida comercialmente es generada y distribuida en forma de corriente alterna, más de la mitad de ésta se utiliza en forma de corriente directa. Por tal razón, los motores de corriente directa están mejor adaptados para muchos procesos industriales que demandan altos grados de flexibilidad en el control de velocidad y par.

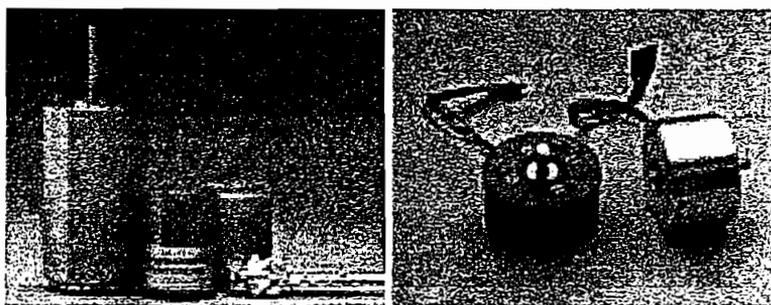


Fig. 2. 10 Motores de Corriente Directa.

Los motores de corriente continua, inicialmente fueron usados sólo donde no se disponía de corriente alterna; en la actualidad, con el desarrollo de la automatización de procesos, así como con la aparición de la robótica, el uso de motores DC ha ido en aumento, siendo útiles para regular continuamente la velocidad del eje y en los casos de necesitar un torque de arranque elevado; adicionalmente, aunque es común utilizar motores universales o a pasos para los procesos de Control Numérico, para maquinarias grandes, se trabaja con esta clase de motores, razón por la cual se desarrolla información referente a éste tipo de motores.

El motor DC, está compuesto por dos piezas, el rotor y el estator, que a continuación se observan en la figura 2.11:

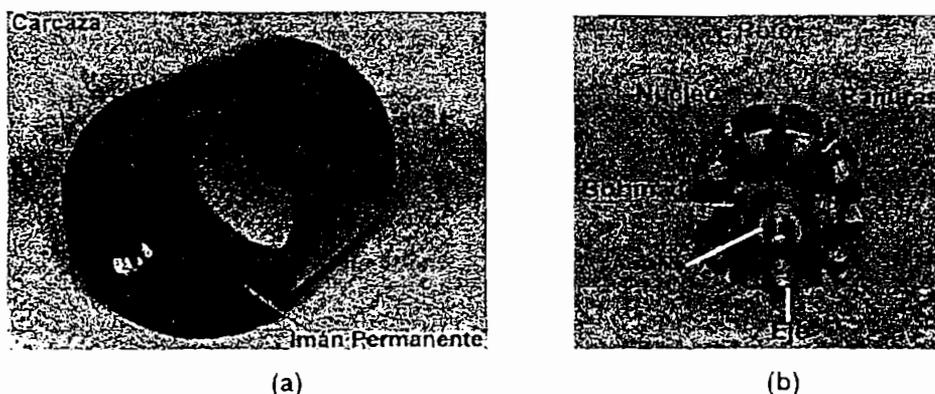


Fig. 2. 11 (a) Estator y (b) Rotor.

El rotor, constituye la parte móvil del motor, el cual proporciona el torque⁹ para mover a la carga. El rotor está conformado por el **eje**, el cual imparte la rotación al núcleo, devanado y colector; el **núcleo**, compuesto por láminas de acero que proporciona una trayectoria magnética entre los polos para que el flujo magnético circule; el **devanado**, compuesto por bobinas (alojadas en las ranuras y conectadas eléctricamente con el colector) aisladas entre sí; y del **colector** o conmutador, compuesto por láminas conductoras sobre uno de los extremos del eje del rotor, y su función es la de recoger la tensión producida por el devanado inducido y transmitirla al circuito por medio de las escobillas.

El estator constituye la parte fija de la máquina y suministra el flujo magnético utilizado por el bobinado del rotor para realizar un movimiento giratorio. El estator está conformado por el **armazón** o yugo que sirve de soporte y proporciona una trayectoria de retomo al flujo magnético del rotor y del imán permanente para completar el circuito magnético; el **imán permanente**, que se encuentra en el armazón y proporciona un campo magnético uniforme al devanado del rotor el cual interactúa con el campo generado por el bobinado para producir el movimiento del rotor; y las **escobillas**, que transmiten la tensión y corriente de la fuente hacia el colector y al bobinado del rotor.

⁹ Fuerza angular ejercida sobre un cuerpo o brazo, con el fin de producir movimiento rotacional o giratorio.

2.4.1.1 TIPOS DE MOTORES DE CORRIENTE CONTÍNUA

Los motores de corriente continua o DC, se clasifican en: Motores Shunt, Motores en Serie, Motores Compound y Motores de Imán Permanente. A continuación, se realizará una breve explicación acerca de estos motores debido a que éste tema es demasiado complejo como para abarcarlo en un solo capítulo.

Motores Shunt.- En los motores con arrollamiento shunt, el inductor está conectado en derivación con el inducido¹⁰, y tiene características de velocidad constante. Los motores con arrollamiento shunt estabilizados poseen un pequeño arrollamiento serie adicionado para estabilizar las características de velocidad y evitar una elevación de la velocidad cuando se aplica la carga.

Motores Serie.- En los motores con arrollamiento serie el inductor está conectado en serie con el inducido y tienen las características de velocidad variable de los motores serie universales, siendo su principal ventaja su facilidad de inversión.

Motores Compound.- Los motores con arrollamiento compound emplean a la vez un inductor serie y shunt, cuyas características de velocidad son intermedias (función del valor del devanado mixto). El arrollamiento serie está conectado en un sentido tal que refuerza el inductor shunt, provocando un incremento del flujo.

Motores de Imán Permanente.- Los motores de imán permanente tienen arrollamientos de inducido, pero no arrollamientos inductores (en el circuito magnético, está dispuesto un imán permanente para crear el flujo requerido).

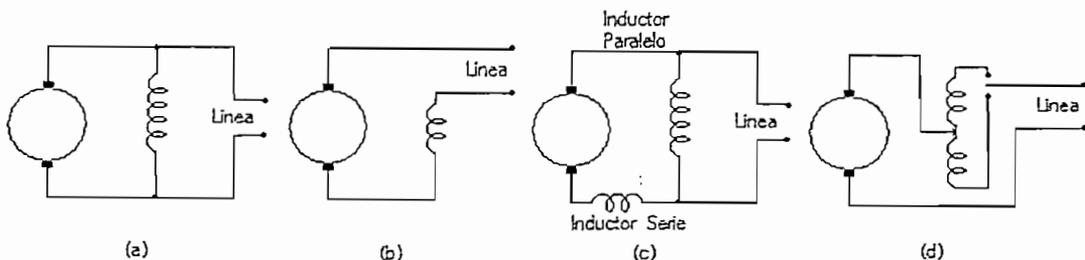


Fig. 2. 12 (a) Motor Shunt, (b) Motor Serie, (c) Motor Compound, (d) Motor serie con Inductor partido.

¹⁰ (f.c.e.m.) Elemento giratorio, constituido por chapas ranuradas entradas a presión sobre el eje, para proporcionar un paso al campo magnético.

2.4.1.2 MOTOR DC SERIE.

Aunque la clasificación de los motores DC es amplia, únicamente se hablará de los motores DC en serie, debido a que el área industrial dedicada a la automatización, utiliza motores DC en serie cuando trabajan con maquinarias grandes y especiales destinadas a un nivel de producción alto. Esto no quiere decir que se descarta el uso de otros motores para éste tipo de maquinarias, por el contrario, las industrias prefieren los motores universales o a pasos por su facilidad de operación. Aunque éstos motores no son la primera opción de uso, se ha creído conveniente, hablar acerca de ellos, ya que existen empresas orientadas al campo del Control Numérico, que prefieren y requieren su uso.

Conocido como un motor de velocidad variable (elevada velocidad en vacío), debido a que su variación de velocidad es más grande que la de un motor shunt o compound. Cuando el motor se encuentra parado, la fuerza contraelectromotriz (f_{cem})¹¹ es nula y la corriente elevada, de manera que el motor se acelera rápidamente; éstos motores están alimentados por un suministro de corriente continua bajo una tensión uniforme, como el de una batería.

A medida que el motor se acelera, la f_{cem} y la velocidad se incrementan, disminuyendo la corriente del inducido (disminuye la excitación). Por tal razón, a medida que el motor se acelera se debilita el campo haciendo que el inducido gire rápidamente hasta generar una f_{cem} que limita la corriente del inducido. A continuación se muestra la regulación de la velocidad de los motores DC¹².

CV ¹³	Velocidad r.p.m.	Arrollamiento Shunt	Arrollamiento Compound
1/20 a 1/8, incl.	1725	20%	30%
1/20 a 1/8, incl.	1140	25%	35%
1/6 a 1/3, incl.	1725	15%	25%
1/6 a 1/3, incl.	1140	20%	30%
½ a ¾, incl.	1725	12%	22%
1/2	1140	15%	25%

¹¹ En cualquier motor DC, la rotación del inducido, es causa de que se induzca una tensión en el circuito del inducido que se opone a la circulación de la corriente que da lugar a la rotación.

¹² VEINOT, CYRIL G., "Motores Eléctricos de Potencia Fraccionaria y Subfraccionaria", 1978, pág. 309

¹³ CV= Caballo de Vapor, HP= Horse Power. 1hp= 1.014 cv = 0.746 kW.

Las características de funcionamiento pueden expresarse bajo la forma de sencillas fórmulas, que a continuación se indican:

$E = f.c.e.m. \text{ generada en el inducido}$

$I_i = \text{corriente del inducido}$

$N = r.p.m. \text{ del inducido}$

$R_i = \text{resistencia total del circuito del inducido, incluidos todos los arrollamientos que forman parte del circuito, la resistencia de las escobillas de contacto}$

$T = \text{par desarrollado por el inducido}$

$V = \text{tensión aplicada al circuito del inducido}$

$\phi = \text{flujo por polo}$

$K_1, K_2, K_3 = \text{constantes de proporcionalidad}$

$$T = K_1 \phi I_i \quad (2-1)$$

$$E = K_2 N \phi \quad (2-2)$$

$$E = V - R_i I_i \quad (2-3)$$

$$I_i = \frac{V - E}{R_i} \quad (2-4)$$

$$N = \frac{K_3 E}{\phi} \quad (2-5)$$

Las tensiones de régimen normales son de 115 y 230 voltios., y la variación de velocidad es definida como la diferencia entre la velocidad en vacío y la velocidad a plena carga, expresada en porcentaje de la velocidad a plena carga.

2.4.1.3 MOTORES PARA USOS ESPECIALES.

Aunque es imposible abarcar todas las clases de servicios especiales de los motores de corriente continua, a continuación, se darán algunos de sus usos.

Motores para Servicios Militares.- Los motores DC de dimensiones reducidas, son usadas en aviones civiles y militares, debido a su peso reducido, dimensiones pequeñas (diámetro aproximado de 3.5 cm) y porque están contruidos para satisfacer las severas exigencias del medio ambiente (para circuitos de 28 voltios o menos). Se encuentran provistos de un freno capaz de parar el motor de 3 a 5 revoluciones desde una velocidad de 10000 rpm.

El rango de dimensiones y características que presentan los motores para éste uso son de:

- De 6.35 a 7.6 cm de longitud
- Potencia nominal para servicio continuo de 0.001 a 0.02 CV a 10000 rpm
- Par con rotor calado de 450% a 350%
- Rendimiento a plena carga del 49% al 50%
- Velocidad en vacío de 25000 a 22000 rpm

Motores Invertibles.- Existen aplicaciones reversibles en que las características de la excitación serie no son convenientes; tales aplicaciones son las de aparatos de elevación, de las máquinas-herramienta; para éstos servicios, los motores Compound (igualmente utilizados los motores DC en serie) tiene las mejores características de trabajo.

Ciertas consideraciones a tomar con los motores, es que puede requerirse un aislamiento especial para los arrollamientos inductores, para su correcto funcionamiento; además, los motores con velocidades de 1725 a 3450 rpm no son convenientes para los servicios mencionados anteriormente, debido a la fuerte absorción de corriente por el motor en el momento de cerrar el conmutador.

Motores para carretillas eléctricas destinadas al manejo de materiales.- Los motores de corriente continua eléctricamente reversible son habitualmente bobinados para 24 a 35 voltios (aún cuando pueden serlo para una tensión más alta), y están equipados con un freno magnético. Éstos motores están construidos para aplicaciones que utilizan como fuente de energía baterías, por tal razón, son utilizadas para pequeñas carretillas y carritos para bolsas de golf.

Motores para servicio marítimo.- Su diámetro es de aproximadamente de 12.7 cm, requieren potencias de 0.1 a 0,75 CV, tensiones de 6 a 110 voltios, velocidades de 500 a 4000 rpm. Pueden ser de excitación serie o compound, siendo usados en la marina en aplicaciones tales como bombas, sopladores y lavabos.

2.4.2 MOTORES UNIVERSALES.

Tiene la forma de un motor de corriente continua en conexión serie. La principal diferencia es que es diseñado para funcionar con corriente alterna. Se utiliza en los taladros, aspiradoras, licuadoras, lustradoras, etc. su eficiencia es baja (de orden del 51%), pero como se utilizan en maquinas de pequeña potencia esta ineficiencia no se considera importante (motor económico y controlable).

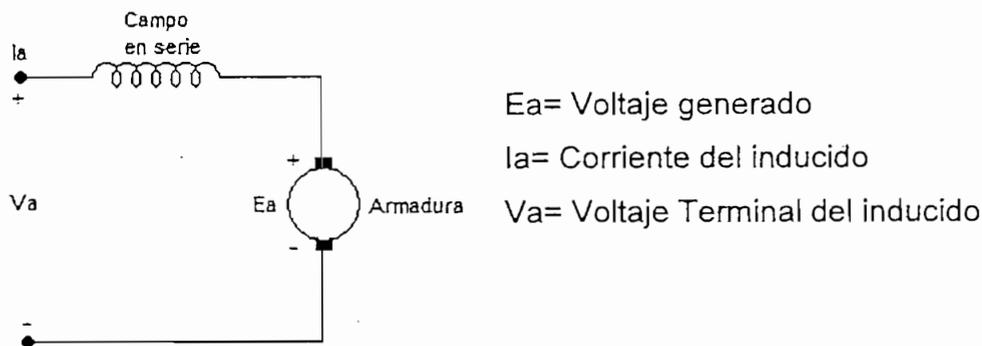


Fig. 2. 13 Máquina Universal conectada en serie.

El motor universal está caracterizado por su aptitud para funcionar con los mismos resultados, tanto con corriente continua, como con alterna de frecuencia de hasta 60 hertzios. Desarrollan mayor potencia por kilo que otros motores de corriente alterna (debido a su elevada velocidad).

Las potencias nominales varían de 0.01 a 1 CV, para motores de servicio continuo, con velocidades a plena carga entre 4000 a 16000 rpm. Dentro de las velocidades más altas es posible obtener las mejores características universales., así como más potencia por kg.

Los motores universales están construidos para una aplicación específica y muchas veces son vendidos como elementos de un conjunto antes que como motores aislados. Se utilizan motores universales pequeños, en los que el peso liviano es importante, como en aspiradoras, aparatos de cocina, perforadoras portátiles, sierras, contorneadores (aplicaciones de Control Numérico), máquinas limpiadoras de alcantarillado, mezcladoras de alimentos, batidoras, etc., que por lo general funcionan a altas velocidades (1500 a 15000 rpm).

2.4.2.1 CARACTERÍSTICAS DEL PAR DE LOS MOTORES UNIVERSALES.

Se emplean dos tipos de motores universales, no compensados, que están contruidos con polos concentrados o salientes; y compensados, que tienen casi la misma velocidad de funcionamiento tanto para corriente alterna como para corriente continua. El motor no compensado es menos costoso y de más sencilla construcción, razón por la cuál es de uso generalizado.

Las curvas de par-velocidad de un motor universal de polos concentrados no compensados (Fig. 2.14a) y compensados (Fig. 2.14b), funcionando sobre corriente continua y alterna, se muestran en la siguiente figura.

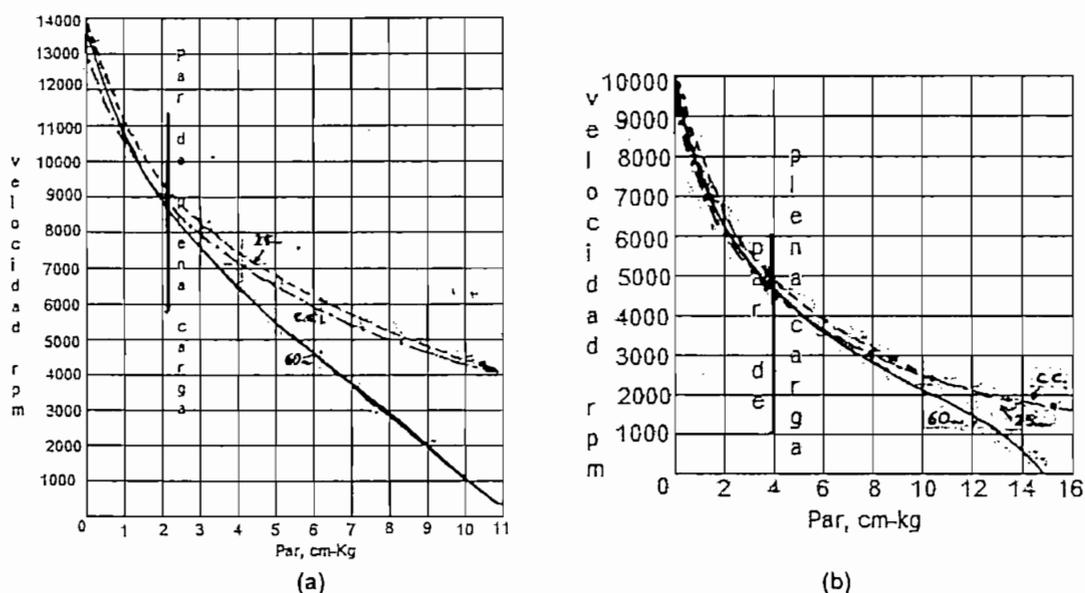


Fig. 2. 14 Curva par-velocidad de un motor universal (a) no compensado de polos salientes de 1/4CV, 8000 rpm; y (b) compensado de 1/4CV, 4000 rpm.¹⁴

En ambos gráficos, se observa que la velocidad disminuye rápidamente para un aumento de la carga y crece para una disminución de ésta.

2.4.2.2 PRINCIPIO DE FUNCIONAMIENTO DE LOS MOTORES UNIVERSALES.

Funcionamiento con corriente continua.- El funcionamiento del motor serie con corriente continua es igual que los motores DC que se explicaron anteriormente.

¹⁴ VEINOT, CYRIL G., "Motores Eléctricos de Potencia Fraccionaria y Subfraccionaria", 1978, pág. 274 y 275

Éstos motores están alimentados por un suministro de corriente continua bajo una tensión uniforme, tal como el de una batería; y tiene una velocidad muy elevada en vacío, cuya variación de velocidad es mucho más grande que para un motor shunt o compound (parado la fuerza contraelectromotriz es nula y la corriente es elevada provocando que el motor se acelera rápidamente).

Funcionamiento con corriente alterna.- Si se alimenta un motor serie con corriente alterna, arranca y gira. La corriente en el circuito del inducido se invierte 120 veces por segundo (60 hertz), pero el campo inductor y el flujo del estator se invierten igualmente 120 veces por segundo, haciendo éstas inversiones en fase con las de la corriente del inducido. Con corriente alterna, el par varía instantáneamente 120 veces por segundo, mientras que el par desarrollado siempre es del mismo sentido.

2.4.3 MOTORES A PASOS.

Los motores a pasos son utilizados en la industria electrónica, especialmente en aplicaciones de automatización, discos, impresoras, fotocopiadoras, y robots, así como en algunos sistemas utilizados en la industria médica.

En el mercado existe una amplia variedad de opciones para controlar el funcionamiento de los motores a pasos, desde microcontroladores y procesadores digitales de señales, hasta circuitos integrados diseñados específicamente para realizar estas funciones. Los motores por pasos son dispositivos electromagnéticos, rotativos, incrementales que convierten pulsos digitales en rotación mecánica (que pueden girar y pararse con una precisión del orden de centésimas de milímetro con alta precisión y gran fiabilidad).

La cantidad de rotación es directamente proporcional al número de pulsos y la velocidad de rotación es relativa a la frecuencia de dichos pulsos. Los motores por pasos son simples de operar en una configuración de lazo cerrado o abierto y debido a su tamaño, proporcionan un excelente torque a baja velocidad.

2.4.3.1 PRINCIPIO DE FUNCIONAMIENTO.

En el primer Capítulo del actual Proyecto de Titulación, se encuentra una explicación más amplia acerca de éste tipo de motores a pasos, debido a que el equipo NEURACTOR CNC-III utiliza estos motores para su funcionamiento.

Por tal razón, a continuación se dará una breve explicación acerca del funcionamiento de motores con el propósito de que el lector entienda el uso de éste tipo de motores en los procesos de Control Numérico.

La característica principal de estos motores es el hecho de poder moverlos un paso a la vez por cada pulso que se le aplique. Este paso puede variar desde 90° hasta pequeños movimientos de tan solo 1.8° , es decir, que se necesitarán 4 pasos en el primer caso (90°) y 200 para el segundo caso (1.8°), para completar un giro completo de 360° .

Básicamente estos motores están constituidos normalmente por un rotor sobre el que van aplicados distintos imanes permanentes y por un cierto número de bobinas excitadoras bobinadas en su estator. Las bobinas son parte del estator y el rotor es un imán permanente. Toda la conmutación (o excitación de las bobinas) deber ser externamente manejada por un controlador.

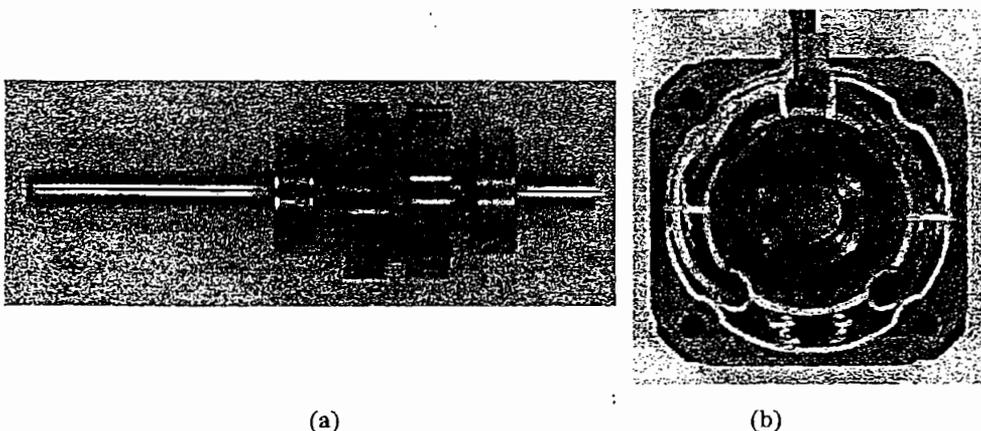


Fig. 2. 15 (a) Imagen del Rotor, (b) Imagen de un estator de 4 bobinas.

El voltaje de alimentación se aplica a las bobinas del estator, alcanzando un desplazamiento angular del flujo magnético que obliga al rotor a moverse para ir enfrentando polos de distinto signo donde existe una mayor fuerza de atracción.

Los motores paso a paso son construidos dentro de una gama de tamaños cuyos pesos varían entre 85 gr. a 510 gr. Los pares de funcionamiento varían desde 80 gr-cm (1.1 onz-pulg) hasta 350 gr-cm (5 onz-pulg). Los pares de retención varían desde 150 hasta 500 gr-cm (2 a 7 onz-pulg).

El motor a pasos puede parar, arrancar, invertir su sentido de giro o marchar a velocidades variables, sin pérdida de impulsos. Con una cantidad baja de impulsos por segundos, el par disponible será aproximadamente el nominal del motor, pero el par disponible decrece con el incremento de los impulsos, tal como se indica en la figura 2.16.

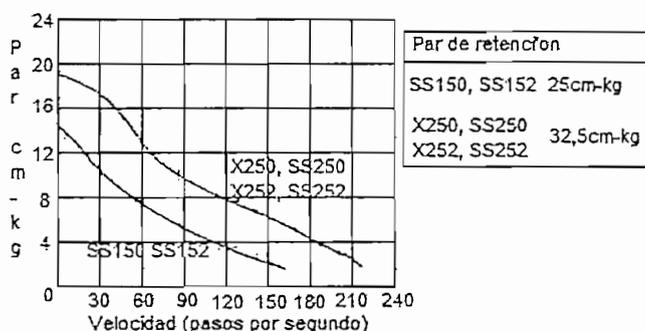


Fig. 2. 16 Efecto de la rapidez de la secuencia de los impulsos sobre el par útil de los motores ordinarios a pasos.¹⁵

2.4.3.2 TIPOS DE MOTORES PASO A PASO.

En el Capítulo I, se desarrolló con más precisión los tipos de motores a pasos según su construcción y según el tipo de bobinado, así como el modo de trabajo de los mismos. Así que a continuación, se hablará en forma general y resumida acerca de los dos tipos de motores principales que se encuentran, los motores unipolares y bipolares.

Los motores bipolares son llamados así porque para hacer girar el eje, deben invertir la polaridad de la alimentación de sus bobinas con una progresión de un paso a la vez. Estos motores se reconocen fácilmente por los 4 hilos que salen de su cuerpo debido al par de bobinas que no tienen toma central. Los motores

¹⁵ VEINOT, CYRIL G., "Motores Eléctricos de Potencia Fraccionaria y Subfraccionaria", 1978, pág. 365

bipolares son más corrientes ya que para una misma potencia, tienen dimensiones menores que los unipolares.

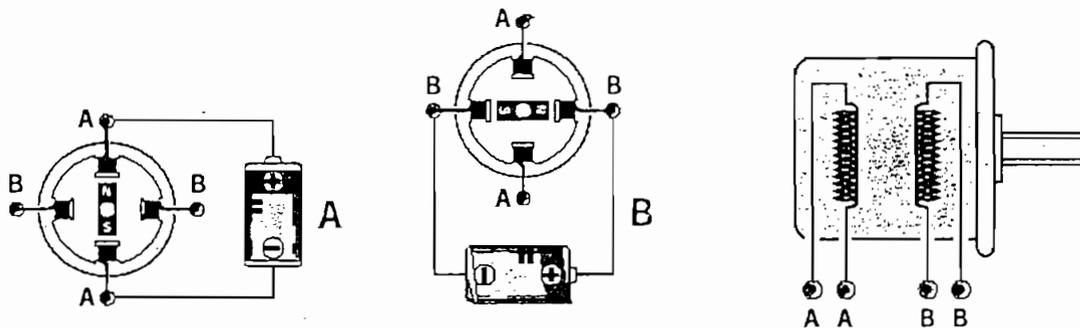


Fig. 2. 17 Motores a pasos bipolares.

Los motores unipolares son llamados así porque al tener un doble arrollamiento en sus bobinas, no necesitan invertir la polaridad de la alimentación; y se los reconocen por los 5 o 6 hilos que salen de su cuerpo, ya que el par de bobinas tienen una toma central cada una.

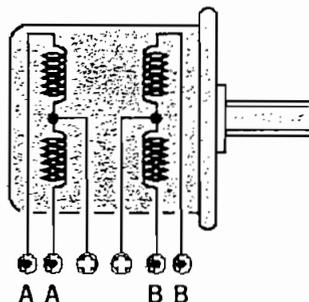


Fig. 2. 18 Motor a pasos unipolar.

Los motores paso a paso pueden ser alimentados con tensiones de 9, 12, 15, 18, 24 o 28 voltios, a condición de que el circuito de control disponga de un sistema que limite la corriente que circula por sus bobinas. Sin este control, corrientes muy elevadas atravesarán los arrollamientos de las bobinas, aumentando al mismo tiempo que el valor de la tensión de alimentación. En este caso, no solo el circuito de control se dañará rápidamente, sino que los arrollamientos de las bobinas del motor se calentarán en exceso y terminarán dañándose igualmente.

2.5 DESCRIPCIÓN DE LOS MOTORES DE PASOS UTILIZADOS EN EL SISTEMA MECÁNICO NEURACTOR CNC-III.

A partir del análisis del trabajo que deben realizar los motores de paso y principalmente de la fuerza que debe realizar el motor para controlar a la herramienta de corte, el NEURACTOR CNC-III utiliza los motores de paso fabricados por la Empresa US-Cyberlab (Figura 2.14), los mismos que tienen las siguientes características:¹⁶

CANTIDAD:	4
MARCA:	US-CYBERLAB
VOLTAJE NOMINAL:	12 Voltios
CORRIENTE POR PASO/BOBINA:	1 Amperio
PRECISIÓN:	1.8 Grados/Pasos (Pasos Completos)
TORQUE:	530 onzas/pulgada

Para que los motores funcionen adecuadamente, se necesita regular el voltaje provisto por la fuente de 12 Voltios, para cada uno de los motores, utilizando un circuito regulador 7812 de 1 Amperio. Además, por recomendación del fabricante de los motores, se ha incluido una resistencia de 10Watts, 4 Ohmios (o alguna resistencia estándar más cercana a éste valor como 3.9 o 4.5 ohmios), para evitar que se quemen los motores cuando el motor deje de funcionar. La rotación del eje de éstos motores se divide en fracciones, el mismo que es de 1.8 grados/paso, o 200 pasos en 360 grados.

El eje principal de los sistemas de los ejes X, Y y Z armados (como se especificó anteriormente), permiten una resolución de 0.1 pulgadas por revolución. Y debido a que el eje del motor está acoplado directamente al eje mecánico de movimiento, la resolución total del sistema es de:

$$0.1 \frac{\text{pulgadas}}{\text{Revolución}} * \frac{1 \text{Revolución}}{200 \text{pasos}} = 0.0005 \frac{\text{pulgadas}}{\text{pasos}}$$

¹⁶ CORDOVA FRANCISCO, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos, pág. 4

Hay que mencionar que excesivas resoluciones limitarán la velocidad máxima del movimiento del sistema, y al reducir la resolución se incrementará ésta velocidad. Si la velocidad de trabajo de los motores (parámetro R=10) fuese de 250 pasos/segundo, la velocidad lineal de movimiento sería:

$$250 \frac{\text{Pasos}}{\text{Segundos}} * 60 \frac{\text{Segundos}}{\text{Minutos}} * \frac{1 \text{Revolución}}{200 \text{pasos}} = 75 \frac{\text{Revoluciones}}{\text{Minuto}}$$

$$75 \frac{\text{Revoluciones}}{\text{Minuto}} * 0.1 \frac{\text{Pulgadas}}{\text{Revoluciones}} = 7.5 \frac{\text{Pulgadas}}{\text{Minuto}}$$

Por lo que se puede concluir que éstos motores poseen suficiente fuerza para mover todo el sistema mecánico y para permitirle a la herramienta de corte una buena precisión y resolución al momento de cortar cierto material con la forma específica del diseño de una pieza. Para cada uno de los motores de pasos salen cinco señales necesarias para que los motores US-Cyberlab funcionen, siendo estos:

- +12 Voltios
- Pulse
- Dir
- Home
- Gnd

Las señales referidas a los +12 Voltios y GND se las toma de la tarjeta de la Fuente de Poder; la señal Home en cambio se la toma de la tarjeta Hall Effect, la cuál indica si los ejes llegaron al origen del sistema deteniendo los movimientos producidos por el motor. La señal referida a la dirección DIR, puede un 0L o 1L, que indican el sentido de giro del motor, y ésta señal llega del Controlador de Motor a pasos CY-545. Mientras que la señal Pulse, se refiere a los pulsos generados por el cristal (12MHz) que general el movimiento de los motores. Las señales Pulse y Dir son conectadas a través de traductores a drivers de corriente externa que son los que proporcional la corriente para que el motor se mueva.

Un dato importante que se quiere mencionar es que aunque los motores pueden alcanzar velocidades altas, al hacer las pruebas con el equipo, se encontró que con los siguientes parámetros los ejes se movieron adecuadamente:

Eje X	Eje Y
F=14 (350 pasos/seg)	F=10 (250 pasos/seg)
S=6 (150 pasos/seg ²)	S=4 (1000 pasos/seg ²)
R=18 (5000 pasos/seg)	R=12 (300 pasos/seg)

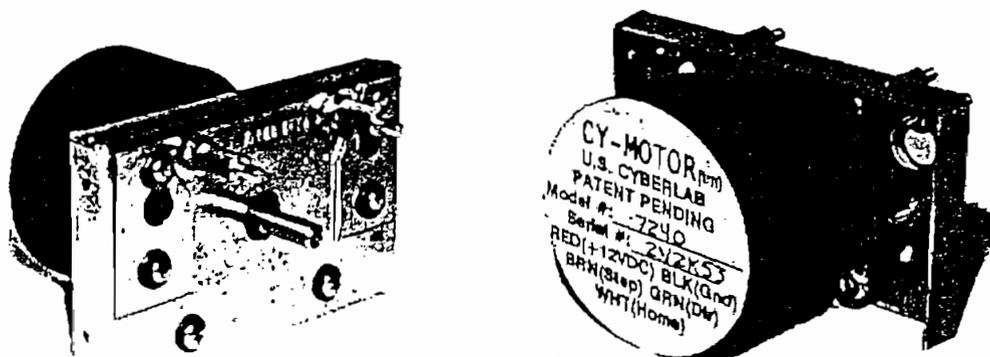


Fig. 2. 19 Motor a pasos U.S. CYBERLAB.¹⁷

2.6 DESCRIPCIÓN DE LOS MICROCONTROLADORES CY-545

CYBERNETIC MICRO SYSTEMS, tuvo la misión de inventar un chip controlador de motores de pasos. Hace ya dos décadas, sacaron al mercado el CY-500, y sus futuras variaciones fueron el CY512 en 1981, el CY525 en 1983 y en 1988, introdujeron un controlador de cuarta generación, el CY-545, que tiene la capacidad de trabajar con un CPU Host¹⁸ o uno independiente.

Las principales características de este controlador son las siguientes:

- 27.000 Pasos/Seg.
- 16 millones de movimientos de pasos.
- Interfase serial y paralelo.
- Interfase para Display de Led's.

¹⁷ U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 4

¹⁸ Hace referencia a cualquier máquina conectada a una red de ordenadores, un nodo con nombre de dominio. Servidor de Internet.

- Interfase para Display LCD¹⁹.
- Circuito integrado de 5 Voltios, 40 pines.
- Parámetros y comandos basados en caracteres ASCII.
- Operación independiente o con Host.
- Interfases internos para Display's LCD o LED.

Es necesario indicar, que en este capítulo, se dará una descripción básica acerca de las características del microcontrolador CY-545, ya que en los anexos propios del capítulo se complementará la información. Adicionalmente, la tesis "Taladro con Control Numérico para la Fabricación de Circuitos Impresos" realizada por el Ingeniero Francisco Córdova, se la tomó como bibliografía base para el desarrollo del presente capítulo.

Mientras que sus principales funciones de control de movimiento son las siguientes:

- Velocidad de pasos programable desde 20 pasos/seg. A 27.000 pasos/seg.
- Parámetros individuales para velocidad inicial, velocidad final y aceleración. Estos parámetros permiten movimientos totalmente programables.
- Movimientos absolutos dentro del rango de 16 Millones de pasos.
- Movimientos continuos si el número especificado de pasos, los que permiten aceleraciones desde la velocidad inicial a la velocidad final, seguida de movimiento indefinido.
- Control interno de la posición durante todo el movimiento.

Mientras que las principales señales y funciones utilizadas para controlar los movimientos son las siguientes:

- Señales de Dirección y Pulso.
- Señales de Control de Dirección internas y externas.
- Señales de Abort y Rampa forzada de bajada para emergencia o control externo de fin de movimiento.

¹⁹ Liquid Crystal Display.

- Señales para controlar los límites CW (horario) y CCW (antihorario) que inhiben el movimiento cuando se alcanza un límite.
- Modo JOB (movimientos manuales) operado externamente, con señales de control de movimiento y señales de inicio y parada.
- Sensor automático de búsqueda de posición inicial.
- Interfase paralelo con 2 líneas de HandShake.
- Interfase serial con selección de velocidad de transmisión fija o adaptativa.
- Soporte de Memoria Externa de hasta 64K, implementada como RAM, ROM, EPROM o EEPROM²⁰.
- Soporte para seteo externo de parámetros, a través de switches tipo Thumb-wheel.
- Display para mensajes de salida, que puede recibir datos desde el interfaz serial o paralelo.
- Líneas multipropósito que pueden ser controladas por los usuarios para funciones de fijación y limpieza de bit, funciones de salto y prueba de bits, funciones de espera por valores de bits determinados y funciones de búsqueda automática de Posición Inicial (Home).

2.6.1 COMANDOS PARA INTERFASES.

El microcontrolador CY545 soporta dos tipos de comandos básicos para interfases, interfase serial e interfase paralelo.

2.6.1.1 Interfase Serial.

El microcontrolador, dispone de un interfaz serial de comunicaciones, el cual puede ser conectado a un computador o un terminal. Los niveles de voltaje que utiliza el microcontrolador CY545 son del tipo TTL²¹ (0L y 1L), por tal razón, se requieren drivers y receivers externos para RS-232, que son los que convierten los niveles de voltaje RS-233 a TTL. El chip MAXIM MAX-233 es el dispositivo que realiza esta función, el cual utiliza una fuente de 5 voltios.

²⁰ RAM = Random-Access Memory (memoria de acceso aleatorio); ROM = Read-Only Memory (memoria de sólo lectura); EPROM = Erasable Programmable Read-Only Memory (ROM borrrable programable); EEPROM = electrically-erasable programmable read-only memory (electrónicamente borrrable programable ROM).

²¹ Tiempo de Vida (Time To Live), cuando se habla de Protocolo IP; o a Tecnología TTL (Transistor-Transistor Logic), la primera tecnología de construcción de circuitos electrónicos digitales que existió.

El interfaz serial puede ser operado mediante la selección fija de la velocidad de transmisión, así el CY545 leerá un 0 lógico al encendido y el valor de la línea IO_REQUEST (pin 13) determinará la velocidad de transmisión (57600, 9600, 2400, 300 baudios); o se la puede operar con una velocidad adaptativa (300, 19200, 38400, 76800 baudios) seleccionada que envía dos caracteres de retorno. El interfaz serial del CY545 trabaja con caracteres de formato fijo, los cuales son de 8 bits de datos, sin paridad y con un bit de parada; los 8 bits de datos son usados para interpretar el valor de los caracteres de comando.

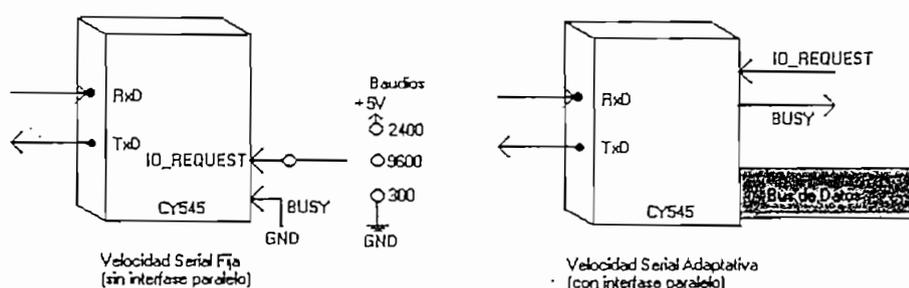


Fig. 2. 20 Velocidad serial fija y adaptativa.²²

2.6.1.2 Interfase Paralelo.

El interfase paralelo usa las señales IO_REQUEST y BUSY; la señal BUSY cuando se encuentra en alto se la utiliza para enviar caracteres a través de las líneas del bus de datos, y si la señal se encuentra en nivel bajo, se dice que el CY545 ha leído al carácter. Con respecto a la señal IO_REQUEST, se coloca en bajo cuando la señal BUSY se encuentra en alto e indica que el carácter está disponible para enviarse, y se coloca en un nivel alto (cuando BUSY está en bajo), cuando se ha retirado el carácter del bus de datos.

Este proceso antes descrito se repite hasta que todos los caracteres hayan sido enviados al CY545. En el modo directo de comandos, el microcontrolador ejecuta cada comando luego de recibido, es decir, que el CY545 irá a ocupado por un periodo largo luego de recibido el último carácter de un comando.

²² CORDOVA FRANCISCO, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos, pág.25

2.6.1.3 Interfase del CY545 al CY233.

El controlador de redes CY233 es ideal para sistemas distribuidos donde se requiere un control central, no son necesarias velocidades altas. Para un interfase con el CY233, la señal del controlador de redes se conecta a la señal IO_REQUEST del CY545, la señal ACK del CY233 es conectada a la señal BUSY del CY545, y la señal ACK del CY233 debe conectarse al USRB6 (pin 27) del CY545, el cual comparte la función FPL en esta línea.

El controlador CY233 tiene la capacidad de conectar hasta 255 dispositivos a una única línea de comunicación serial RS-232, tales como un p rtico COM1 o COM2 de un PC-IBM, a trav s de una  nica direcci n asignada a cada dispositivo.

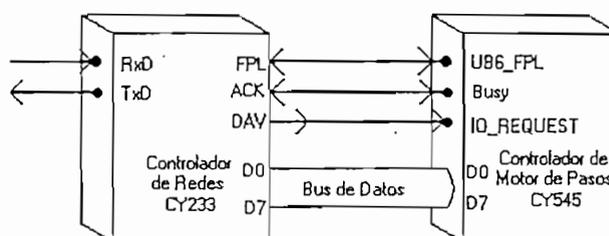


Fig. 2. 21 Interfase del CY545 al CY233.²³

2.6.2 SUMARIO Y FORMATO DE LOS COMANDOS.

En el actual Proyecto de Titulaci n, se dar  una breve explicaci n de los temas ya que el objetivo es el de dar al lector una informaci n b sica y clara acerca de los elementos que se utilizan para la realizaci n del proyecto.

El microcontrolador CY545 trabaja con los formatos ASCII o Binario. Cada uno de estos formatos est n conformados por una letra may scula y un conjunto de n meros conocidos como argumento (los comandos terminan con un car cter de retorno). Entra la letra y el argumento se coloca un espacio, y cuando se tienen m ltiples argumentos, la separaci n es por medio de una coma o un espacio.

²³ CORDOVA FRANCISCO, "Taladro con Control Num rico para la Fabricaci n de Circuitos Impresos, p g.24

2.6.2.1 Formato De Comandos ASCII y Binario.

Todos los formatos de comandos ASCII, empiezan con una letra ASCII que selecciona el comando, seguido de un espacio y luego el valor del parámetro.

Todos los comandos ASCII terminan con un carácter de retorno ASCII, que se muestra como Chr\$(13); para los valores de los parámetros en el modo ASCII, se pueden usar dos formatos, un formato decimal que consiste de dígitos del 0 al 9 o un formato hexadecimal que consiste de dígitos del 0 al 9, letras de A a la F, y el sufijo H. Los parámetros hexadecimales deben comenzar con un dígito.

A continuación se presentan algunos ejemplos de comandos ASCII y Binario respectivamente:

R 135 chr\$(13)	Comando Rate (velocidad inicial del motor), parámetro decimal simple en ASCII.
R 87H chr\$(13)	Comando Rate, parámetro en forma hexadecimal en ASCII.
D 12345 chr\$(13)	Parámetro Delay (retardo) de 16 bits en decimal, en ASCII.
N 3000 chr\$(13)	Argumento Número de pasos de tres bytes, valor 332211H, en Binario.

2.6.2.2 Comandos de Control del Motor.

Estos comandos se usan para fijar las condiciones para el movimiento del motor de pasos, para fijar la velocidad de los pasos, para determinar la dirección de los movimientos del CY545, para controlar el modo de operación de pasos para el CY545, para manejo de bits, para permitir el control de la memoria externa, para controlar los saltos y el flujo de programa cuando el CY545 ejecuta programas desde la memoria externa y otros comandos generales para inicialización, retardos y solicitud de datos. Adicionalmente; la señal JOB se usa para mover el motor manualmente en cualquier momento.

A continuación, se dan ejemplos de varios de los comandos antes mencionados:

R Rate	Fija la velocidad máxima de los pasos.
F Rate	Fija la velocidad inicial de los pasos.

S Slope	Fija la aceleración.
N Num24	Fija el número de pasos.
+	Selecciona dirección horaria.
-	Selecciona dirección anti horaria.
A	Pos24 Fija la posición actual.
C	Selecciona el modo continuo de pasos.
G	Pasos en modo relativo
P pos24	Movimiento a la posición determinada.
B Bit	Fija o limpia un bit específico.
B 2<cr>	Fija el User Bit (pin 24 del microcontrolador CY545) B
02H<cr>	2 a alto.
/B 2<cr>	Limpia el User Bit 2 a bajo.
B 12H<cr>	
T bit,addr	Salto a la dirección especificada hasta que el bit alcance el valor especificado.
W bit	Espera hasta que el bit alcanza el valor especificado.
00H a 07H	Verifica si son 1 los user bits 0 al 7
18H a 1FH	Verifica si son 0 los bits del bus de datos 0 al 7.
Y Addr16	Fija el puntero de dirección externa.
E	Inicia la entrada de comandos seguidos en la memoria.
Q	Termina la entrada de comandos en la memoria.
X	Ejecuta comandos desde memoria externa.
0	Para la ejecución de los comandos desde la memoria externa.
J Addr	Salto a una dirección en la página actual.
O Mode	Fija el modo de operación del CY545.
I	Inicializa el CY545.

2.7 CONTROLADOR DE REDES LOCALES CY-233.

El microcontrolador CY545, se comunica con el computador a través del pórtico serial mediante un Controlador Inteligente de Redes Locales CY233, fabricado por Cybernetic Micro System.

La principal función del CY233 es la de transferir información entre dos sistemas separados y asincrónicos en tiempos multiplexados o con característica serial, manejando el flujo de datos que se transfiere entre varios sistemas. El CY233 provee todas las funciones necesarias para soportar aplicaciones InterSistema (sistema a sistema) tales como Redes de Área Local (LAN), incluyendo el soporte para el sistema "Token Ring"²⁴ El CY233 puede direccionar hasta a 255 dispositivos locales no inteligentes, y ha sido optimizado para aplicaciones en las cuales algunos o todos de los CY233s, en una red de anillo, son dispositivos inteligentes tales como micro computadores.

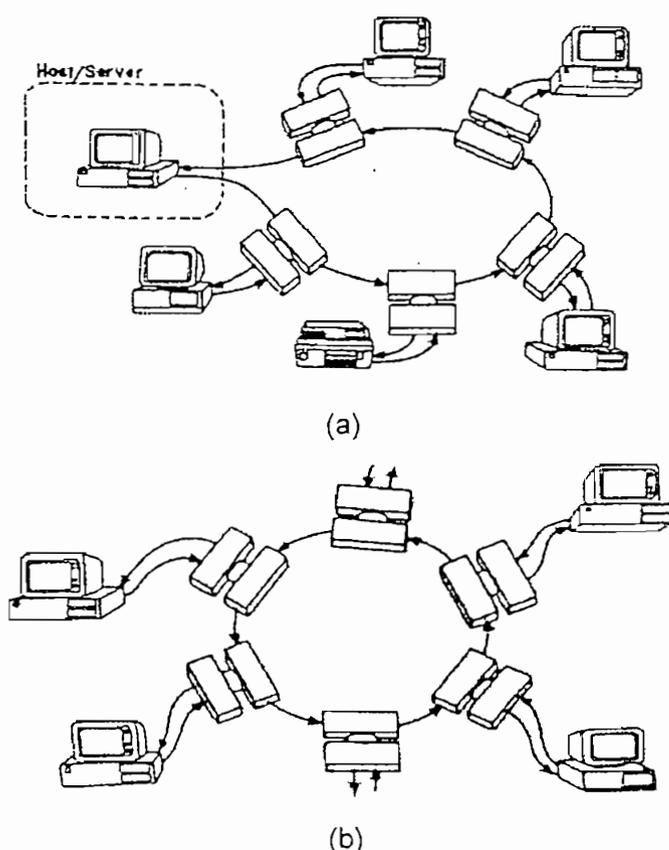


Fig. 2. 22 (a) Red de anillo HOST- el Host está en el anillo; (b) Redes de anillo Punto a Punto- Todos los dispositivos están en el anillo.²⁵

Una gran ventaja del controlador inteligente de redes locales CY233 es que simplifica la comunicación de información digital entre sistemas y entre secciones

²⁴ Tipo de tecnología de red que controla el acceso de medios pasando un paquete distintivo, llamado token (ficha), de máquina en máquina. Una máquina puede transmitir un paquete sólo cuando tiene la ficha.

²⁵ CORDOVA FRANCISCO, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos, pág.106-107

de un sistema. Provee un interfaz fácil de usar entre dispositivos seriales RS-232 y dispositivos paralelos compatibles TTL.

2.7.1 MODOS DE OPERACIÓN.

El CY233 tiene dos modos de operación, el Modo Esclavo y el Modo Master.

En el Modo Esclavo, el CY233 espera por un mensaje serial de la red antes de iniciar la transferencia paralela de datos. La red y la computadora de control determinan el flujo de comunicaciones y las acciones del esclavo CY233.

En el Modo Master, el CY233 solicita las lecturas desde el dispositivo paralelo cuando no hay actividad en la red. Las lecturas se producen desde direcciones secuenciales válidas, sabiendo que el CY233 chequea la validez de la dirección antes de realizar la operación de lectura.

2.7.2 COMANDOS DEL CY-233.

A continuación, se dará una breve reseña acerca de los comandos más importantes usados por el controlador CY-233 para la comunicación con el computador. Cabe mencionar que en el programa diseñado en Visual Basic no se utilizó esta codificación, pero la información que se recibe del controlador, concuerda con la codificación que a continuación se mencionan.

Los encabezamientos de mensajes de los comandos usados por el CY-233 usan letras mayúsculas, sin restricciones en la porción de datos que forman parte del mensaje, y adicionalmente, cualquier código (excepto el carácter terminador) puede ser usado dentro de la sección de datos.

Los comandos se clasifican en tres grupos, los *Comandos Seriales Network* (Fig. 2.23a), que son enviados y recibidos como mensajes en el lado serial del CY233 (parte de una red serial); el segundo grupo, son los *Comandos Paralelos* (Fig. 2.23a), que son enviados al CY233 a través de un dispositivo local paralelo conectado al mismo; y el último, conformado por los *Comandos LAN* (Fig. 2.23b), que son comandos seriales enviados al CY233 LAN desde el sistema serial local

al nodo LAN de una red, estos comandos seriales LAN, se usan para controlar las comunicaciones entre el sistema local y la red.

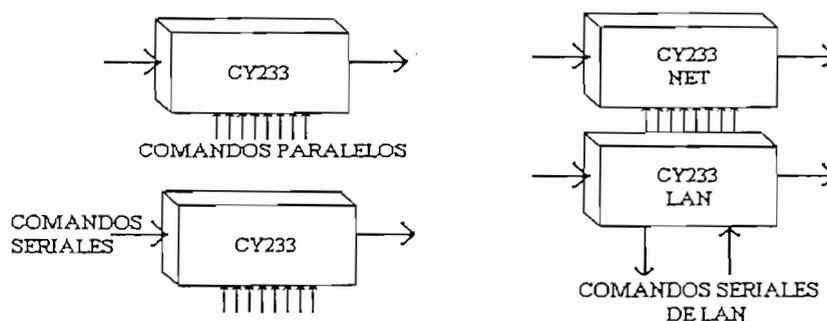


Fig. 2. 23 (a) Comandos Paralelos y Seriales; (b) Comandos LAN.²⁶

A continuación, se enuncian algunos ejemplos de comandos usados por el controlador inteligente de redes locales CY-233:

Raa chr\$(13)	Mensaje enviado por el Host serial.
Wad...d chr\$(13)	Comando write, transfiere cualquier número de bytes al dispositivo paralelo y termina al llegar el terminador.
Xaad...d chr\$(13)	Comando transfer, similar al comando write, no escribirá el carácter de retorno.
laa chr\$(13)	Initialize, que permite a la red reiniciar el CY233.
Qaa chr\$(13)	Query, mecanismo para reporte de errores de paridad.
Taat...t chr\$(13)	Timeout, permite a la red redefinir el periodo de tiempo fuera para el handshake ²⁷ paralelo.
Paat...t chr\$(13)	Periodic delay, para un retraso periodico, cuando el CY233 está en modo Master.
l chr\$(13)	Comando de inicialización paralelo.
S chr\$(13)	Cuando lo envía el pórtico paralelo.
Waad...d chr\$(13)	Como se lo envía al sistema serial local.
Raad...d chr\$(13)	Como se lo envía al sistema serial local.

²⁶ CORDOVA FRANCISCO, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos, pág.132-133

²⁷ Se refiere al protocolo de comienzo de comunicación entre dos máquinas o sistemas, es decir, parte inicial del protocolo en el que dos máquinas se ponen de acuerdo sobre el formato, velocidad y secuencia que seguirán en el resto de la comunicación.

2.7.3 CONEXIÓN AL COMPUTADOR.

Los niveles de voltaje del CY233 son TTL, siendo necesario adaptarlos a niveles RS-232, para ello, se utiliza el integrado MAX-233, que posee un circuito interno de generación de voltaje y dos juegos de receivers y drivers. El MAX-233 requiere de una fuente de voltaje de 5 voltios, la misma que es utilizada en todo el sistema. Las señales TxD y RxD del CY233 están conectados a un driver y un receiver del MAX-233, permitiendo la comunicación con el Host o el otro CY233 del sistema.

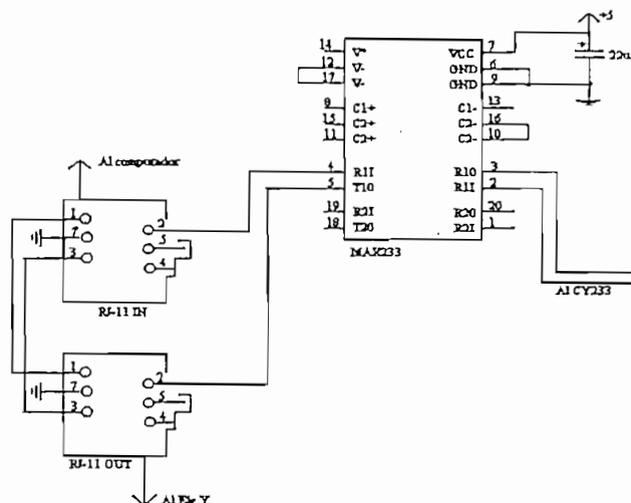


Fig. 2. 24 Conexión del MAX233 al CY233.²⁸

2.8 DESCRIPCIÓN DEL PROGRAMA CIMCO EDIT V.5.

CIMCO Edit V5 es la última versión de una larga línea de soluciones CNC de edición de CIMCO Integration. CIMCO Edit V5 incluye un número de mejoras sobre las versiones anteriores.

El programa CIMCO carga archivos extensos rápidamente, con un uso mínimo de la memoria. El protocolo standard avanzado le permite comunicarse con múltiples CNC simultáneamente. CIMCO Edit V5 tiene la capacidad de crear, cambiar y controlar programas CNC para su máquina CNC.

²⁸ CORDOVA FRANCISCO, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos, pág.156

El RJ11 (Registered Jack-11) es una interfaz física usada para conectar redes de teléfono. El RJ11 se refiere expresamente al conector de medidas reducidas el cual está grimpado al cable telefónico y tiene cuatro contactos (pines) para cuatro cables de teléfono aunque se suelen usar únicamente dos. Estos conectores también se usan para conectar algunos tipos de redes de área local (LANs), aunque los más comunes son los conectores RJ-45.

El usuario puede situar las ventanas y la barra de herramientas como se desee dentro de la ventana principal. Las funciones de CIMCO Edit V5 pueden ser activadas a través del menú principal, o a través de las barras de herramientas.

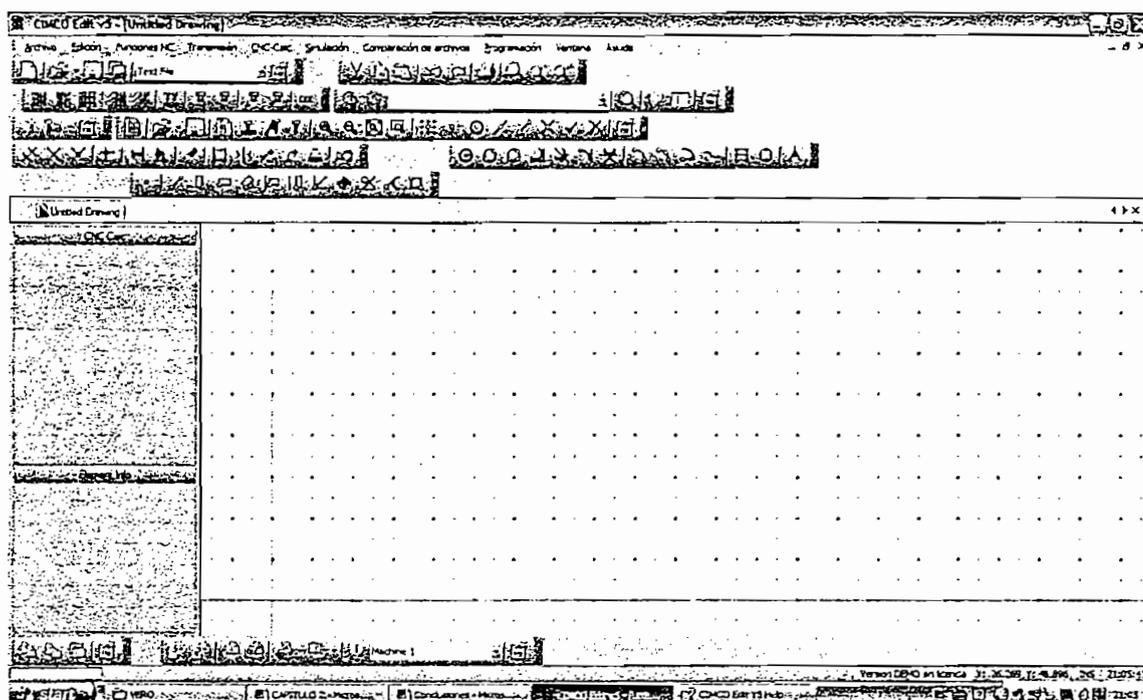


Fig. 2. 25 Pantalla principal con las funciones del programa CIMCO Edit v.5.

2.8.1 FUNCIONES.

El usuario puede deshacer y rehacer (casi) todas las operaciones realizadas desde el archivo abierto, entre las opciones está el de encontrar la posición de un cambio de herramienta, un cambio de la velocidad del eje de rotación o mostrar el rango de los ejes X, Y y Z.

A través de las opciones que se encuentran en la parte superior de la ventana o pantalla principal, es posible personalizar los colores utilizados para mostrar coordenadas, reenumerar números de bloque y agregar o quitar espacio para incrementar la legibilidad, o disminuir tamaño del programa/tiempo de transmisión.

La parte DNC de CIMCO Edit V5 le permite enviar y recibir programas CNC a su máquina en el taller. El archivo puede ser enviado desde una ventana de CIMCO

Edit V5 o directamente desde el disco duro. La transmisión del archivo es realizada mientras se trabaja con otros programas CNC.

2.8.2 MENÚ ARCHIVO.

En este capítulo se describen archivos en apertura, grabado e impresión. Las funciones están colocadas en el mismo orden que en el menú Archivo, la función puede ser activada a través de la barra de herramientas de Archivo o por una secuencia de las teclas se mostrarán junto al nombre del comando.

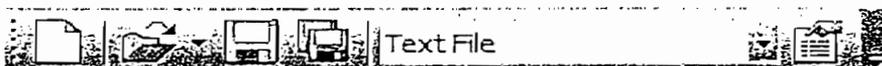


Fig. 2. 26 Barra de herramientas del Menú Archivo.

Cabe mencionar que para que el programa diseñado en Visual Basic para realizar un sistema de corte, requiere escogerse la opción "**Text File**", si el usuario escoge otra opción, el programa no lo reconocerá como uno generado por CIMCO.

- **Nuevo / Ctrl-N.-** Una ventana vacía será abierta. Instale el tipo de máquina para el programa seleccionando de la lista de la derecha.
- **Abrir / Ctrl-O.-** la ventana abrir archivo, permite abrir un programa existente en la memoria.
- **Cerrar / Ctrl-F4.-** Cierra la ventana en uso. Se le preguntará si desea grabar cambios si los archivos han sido modificados.
- **Grabar / Ctrl-S.-** Graba el programa CNC en uso.
- **Grabar como.-** Graba el programa CNC en uso bajo un nuevo nombre.
- **Imprimir.-** Imprime el programa CNC en uso. Este ícono está en la barra de herramientas de Edición. Si desea imprimir una parte de programa, marcar el área de interés y elija Imprimir.
- **Salir / Alt-F4.-** Salida de Edición de CNC. Si hay archivos abiertos, que no han sido grabados, se le preguntará si desea grabarlos.

2.8.3 MENÚ EDICIÓN.

En este capítulo se describen las funciones ordinarias de la edición de texto. La función puede ser activada a través de la barra de herramientas de Edición o con un atajo de teclas.



Fig. 2. 27 Barra de herramientas de Edición.

Una parte del texto es marcado arrastrando el cursor desde el punto de comienzo hasta el punto final mientras se presiona el botón izquierdo del ratón. El color del fondo en el texto marcado es cambiado para indicar el área seleccionada.

- **Deshacer / Ctrl-Z.-** deshacer la última operación.
- **Rehacer / Ctrl-Y.-** Rehacer la última operación. Usted puede rehacer todas las operaciones deshechas realizadas desde la última apertura del archivo.
- **Cortar / Ctrl-X.-** Elimina el texto seleccionado y lo graba en el sujetapapeles.
- **Copiar / Ctrl-C.-** Copia el texto seleccionado en el sujetapapeles.
- **Pegar / Ctrl-V.-** Pega el contenido en la posición actual del cursor.
- **Eliminar / Del.-** Elimina la parte seleccionada del programa CNC.
- **Ir a línea/número de bloque / Ctrl-J.-** Pasar a bloque o número de línea.
- **Encontrar / Ctrl-F.-** Especificar una secuencia de texto y buscar la misma en todo el programa CNC.
- **Encontrar próximo / F3.-** Encuentra la próxima ocurrencia de la secuencia de texto buscada en el programa CNC .
- **Reemplazar.-** Reemplaza una secuencia especificada en el programa CNC con otra secuencia.
- **Insertar archivo.-** Inserta un archivo en la posición actual del cursor.
- **Añadir archivo.-** Añade un archivo al final del presente programa CNC.

2.8.4 MENÚ DE FUNCIONES NC.

Describe las funciones que se refieren a la apariencia del programa CNC. Los comandos y los números de bloque pueden ser cambiados; si la función puede ser activada a través de la barra de herramientas de NC, se mostrará un icono próximo al nombre del comando.

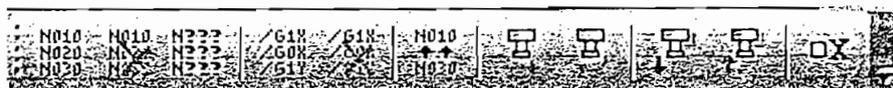


Fig. 2. 28 Barra de herramientas NC.

- **Insertar número de bloque / Ctrl-T.-** Inserta números de bloque en el programa CNC.
- **Quitar números de bloque.-** Elimina los números de bloque del programa.
- **Insertar espacios en blanco.-** Inserta espacios entre los comandos en el programa CNC. Los espacios existentes son dejados como antes.
- **Mayúsculas / Minúsculas.-** Cambia el texto a mayúscula o minúsculas.

Aunque se han descrito ciertos íconos del Menú, indicaremos que no se utilizó para el diseño de figuras.

2.8.5 MENÚ DE TRANSMISIÓN.

Se tienen las funciones relativas al envío de archivos desde y hacia máquinas CNC. Las funciones se pueden realizar a través de la barra de herramientas de Transmisión.



Fig. 2. 29 Barra de herramientas DNC de enviar/recibir.

Nota: El editor debe tener licencia para operaciones DNC para usar esta barra de herramientas.

Esta parte de la descripción de cada ícono que conforma el Menú de Transmisión no se la explicará debido a que no se la utilizó para el diseño de los gráficos a recortar.

2.8.6 MENÚ PROGRAMACIÓN.

Hay un conjunto de parámetros para cada modelo de máquina.

- **Colores.-** Parámetros de color, para destacar comandos específicos, comentarios y selecciones.
- **Números de bloque.-** Parámetros de la numeración de bloques en el programa CNC.
- **Modelos de máquinas.-** Parámetros de la sintaxis del programa para los diferentes modelos de máquina.

- **Mostrar Barra de Estado.**- Selecciona cuándo la línea de estado debe ser visible y cuándo no.
- **Mostrar Barra de Herramientas.**- Selecciona cuál de las barras de herramientas deben mostrarse.

2.8.6.1 Parámetros Del Programa General.

Si se desea cambiar los parámetros del programa que conciernen al programa en general, ingrese al menú Programación y escoja Programación Global.

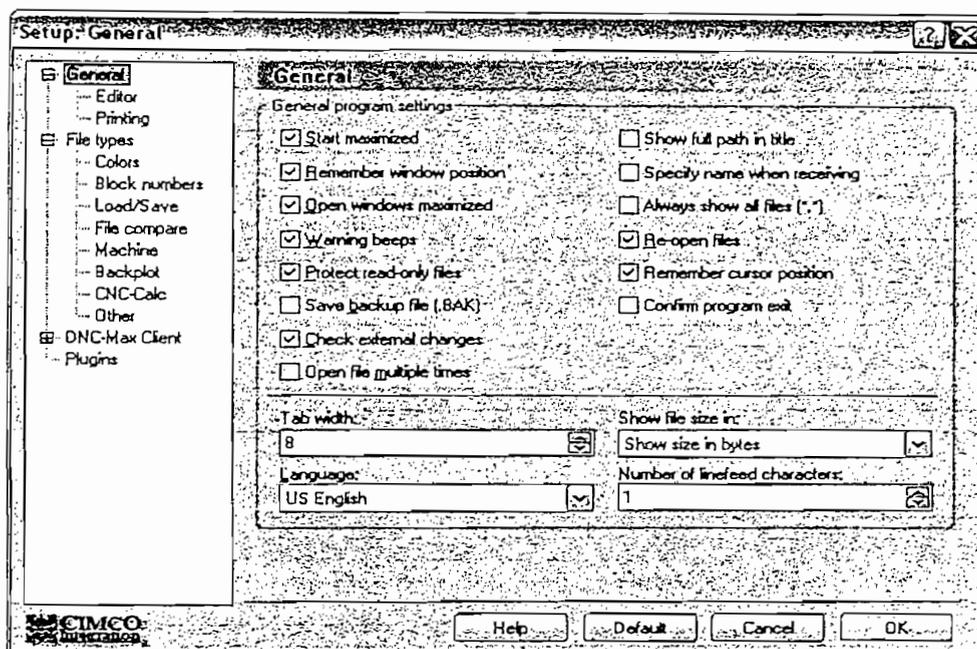


Fig. 2. 30 Programación General.

En ésta ventana, el usuario puede manipular los parámetros de la pantalla principal seleccionado cualquier opción que se presenta en la parte izquierda de la pantalla general.

2.8.7 MENÚ CNC-CALC.

El Menú CNC_Calc, es el más importante, debido a que da la posibilidad de diseñar cualquier gráfico que el usuario desee recortar más adelante.

Su uso es sumamente sencillo, por lo que a continuación, se mostrarán, las opciones que presenta éste Menú con una breve explicación de las viñetas más usadas para la elaboración de figuras.

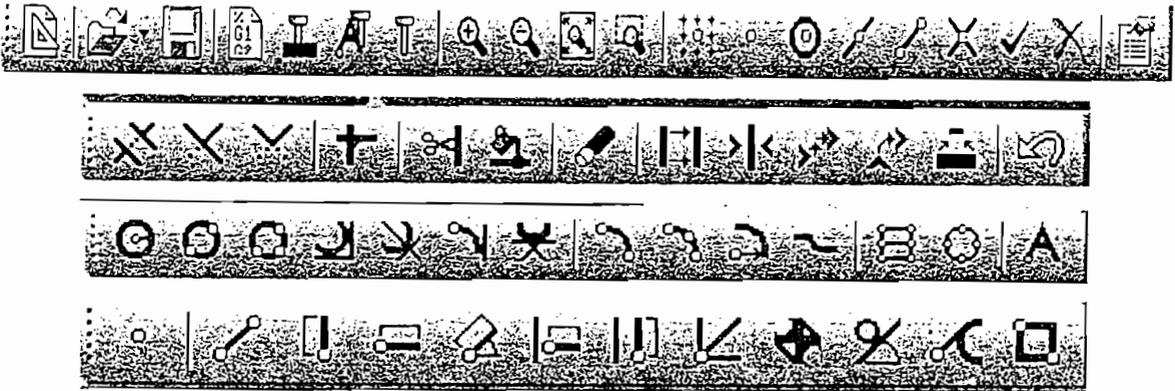


Fig. 2. 31 Herramientas del Menú CNC_Calc.

- **Open New Drawing Windows.-** A través de ésta opción se abre una nueva pantalla que el usuario puede utilizar para realizar un nuevo gráfico.
- **Zoom In / Out.-** Permite acercar o alejar cierto gráfico creado en CIMCO.
- **Trim two elements at intersection.-** Ésta opción permite cerrar o unir dos líneas consecutivas de un gráfico.
- **Break element in two pieces.-** Se utiliza ésta opción para dividir en varias parte a un determinado gráfico.
- **Delete element.-** Permite eliminar o borrar algún elemento incorrecto del gráfico.
- **Construct a circle.-** Para el diseño de un círculo, se tienen tres opciones. La primera opción requiere los datos del centro y radio; la segunda, requiere las posiciones de los puntos del círculo y la tercera opción requiere de tres puntos para formar el círculo.
- **Construct arc.-** Al igual que en el círculo se presentan tres opciones que poseen los mismos requerimientos que un círculo.
- **Construct line between 2 points.-** Este opción, permite la construcción de una línea utilizando posiciones iniciales y finales de la línea.
- **Construct a rectangle from two corners.-** Permite construir un rectángulo mediante las posiciones de dos esquinas de la misma.

2.8.8 ARCHIVOS CDD Y TXT.

Cada vez que el usuario construye y graba alguna figura, el archivo llevará la extensión CDD. Con ayuda de las herramientas antes descritas, el usuario podrá realizar cualquier gráfico, recordando que se debe construir el perfil de la figura, ya que si no se cumple con éste requisito, la herramienta atravesará el gráfico para construir las figuras que se encuentren en el interior de la misma. A continuación, se muestra un ejemplo, donde el nombre del archivo es Mujer.cdd

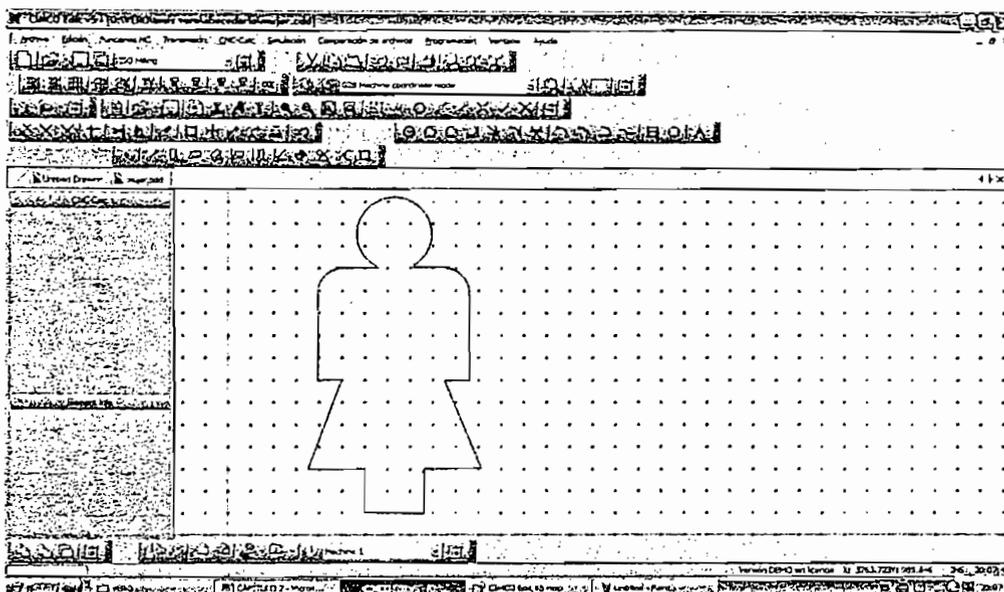


Fig. 2. 32 Archivo Mujer.cdd.

Como se puede observar, el programa CIMCO v.5 es parecido a otros programas como el AutoCad. La razón por la que no se escogió trabajar con el programa AutoCad, es por que Cimco está orientado al Control Numérico y permite convertir a un archivo TXT (da los datos para realizar el proceso de corte), los datos que se guardan se dan en forma desordenada, mientras que con el programa CIMCO, el archivo con extensión TXT (Mujer. txt) es más sencilla de procesar.

El formato de la información del ejemplo Mujer.txt realizado con el archivo CIMCO v.5 es el siguiente:

CNC-Calc 2000

Text File

ARC (7290.7473323448739, 13556.7725629192350, 1642.0978783944815, 5.213653246473237, 4.233278685142946)

LIN (5980.3720368957329, 1013.2318021314204, 8637.5711226480234,
 971.713066353830070)
 ARC (9468.9105372996091, 10897.7006049138650, 1134.4854628769538,
 6.209403788299736, 1.441202401341430)
 LIN (8637.5711226480234, 971.71306635383007, 8596.0523869331446,
 3006.131119455800200)
 LIN (8596.0523869331446, 3006.1311194558002, 11128.6952655407970,
 3047.649855233391000)
 LIN (11128.6952655407970, 3047.6498552333910, 9509.4645726604958,
 6991.929754104558000)
 LIN (9509.4645726604958, 6991.9297541045580, 10595.3309673010300,
 6991.929754104558000)
 LIN (10595.3309673010300, 6991.9297541045580, 10624.1633979525450,
 10797.810456566280000)
 LIN (9592.5020440902554, 12057.2155189706880, 8014.7900869248306,
 12057.215518970686000)
 LIN (3947.3818331838211, 7010.0938067994039, 3947.3818331838211,
 10915.255154213544000)
 LIN (3947.3818331838211, 10915.2551542135440, 3947.3818331838211,
 10939.899201134363000)
 ARC (5036.6630061046226, 10935.2620232531370, 1090.7373235623447,
 1.611335369866453, 3.109202966578845)
 LIN (6605.2307739354483, 12028.4358675447180, 5019.3705355906495,
 12037.446436740938000)
 LIN (3947.3818331838211, 7010.0938067994039, 4990.6143991424069,
 7010.093806799403900)
 LIN (4990.6143991424069, 7010.0938067994039, 3515.4702534018338,
 3006.131119455800200)
 LIN (5980.3720368957329, 3006.1311194558002, 3515.4702534018338,
 3006.131119455800200)
 LIN (5980.3720368957329, 3006.1311194558002, 5980.3720368957329,
 1013.231802131420400)
 END

CAPÍTULO 3

DESARROLLO DEL SOFTWARE.

3.2 DIAGRAMA DE FLUJO GENERAL.

El diseño del programa para la ejecución del proceso de corte, permite controlar los parámetros de trabajo del equipo, y da la posibilidad de ejecutar diferentes funciones según el modo de trabajo que se escoja.

El programa desarrollado para implementar un sistema de corte se dividió en cinco partes las cuales se indicarán a continuación:

- Menú Principal
- Manejo Manual del Cortador
- Manejo Automático del Cortador
- Manejo de Archivos
- Fin de Programa

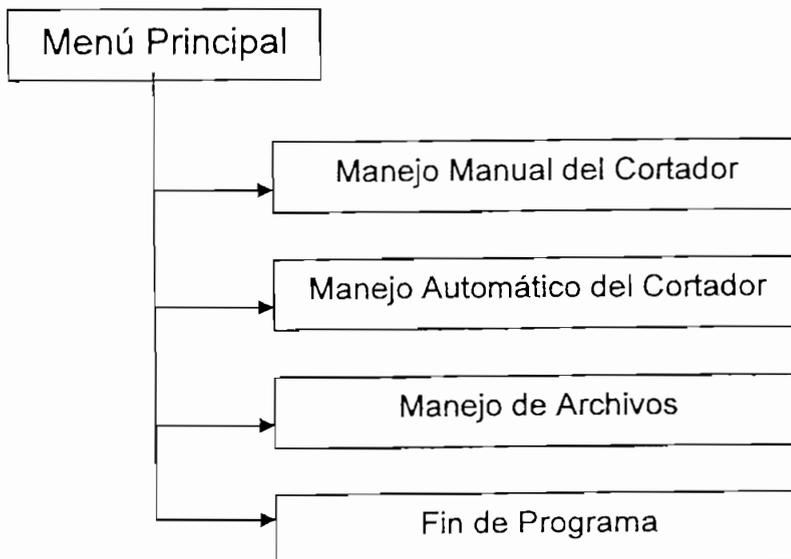


Fig. 3. 1 Diagrama de Opciones para el manejo del Equipo.

Al ejecutar el programa, se observa como primera pantalla a la presentación o Menú Principal del mismo, el cual contiene las opciones del programa.

El Modo Manual de trabajo, permite el movimiento independiente de cada uno de los ejes, en ésta opción el usuario fija los parámetros de funcionamiento de los microcontroladores para verificar el buen funcionamiento del equipo.

El Modo Automático, ejecuta un proceso continuo de corte sobre materiales como espumaflex o fomix, al seleccionar un archivo con extensión CBC (ejecutado por la opción de manejo y conversión de archivos) que contiene los datos de velocidad y posición que los microcontroladores reciben para ejecutar los movimientos.

El Manejo y Conversión de Archivos, permite escoger el archivo que se genera en el programa CIMCO V.5 (debe ser grabado con extensión TXT) que contiene la información de los puntos que conforman determinado gráfico; para convertirlo en un archivo de extensión CBC que transforma los datos en puntos y velocidades con los cuales el NEURACTOR CNC-III se mueve. Esta opción muestra en la pantalla el gráfico del archivo escogido.

3.2.1 DIAGRAMA DE FLUJO DEL PROCESO DE CORTE EN MODO AUTOMÁTICO.

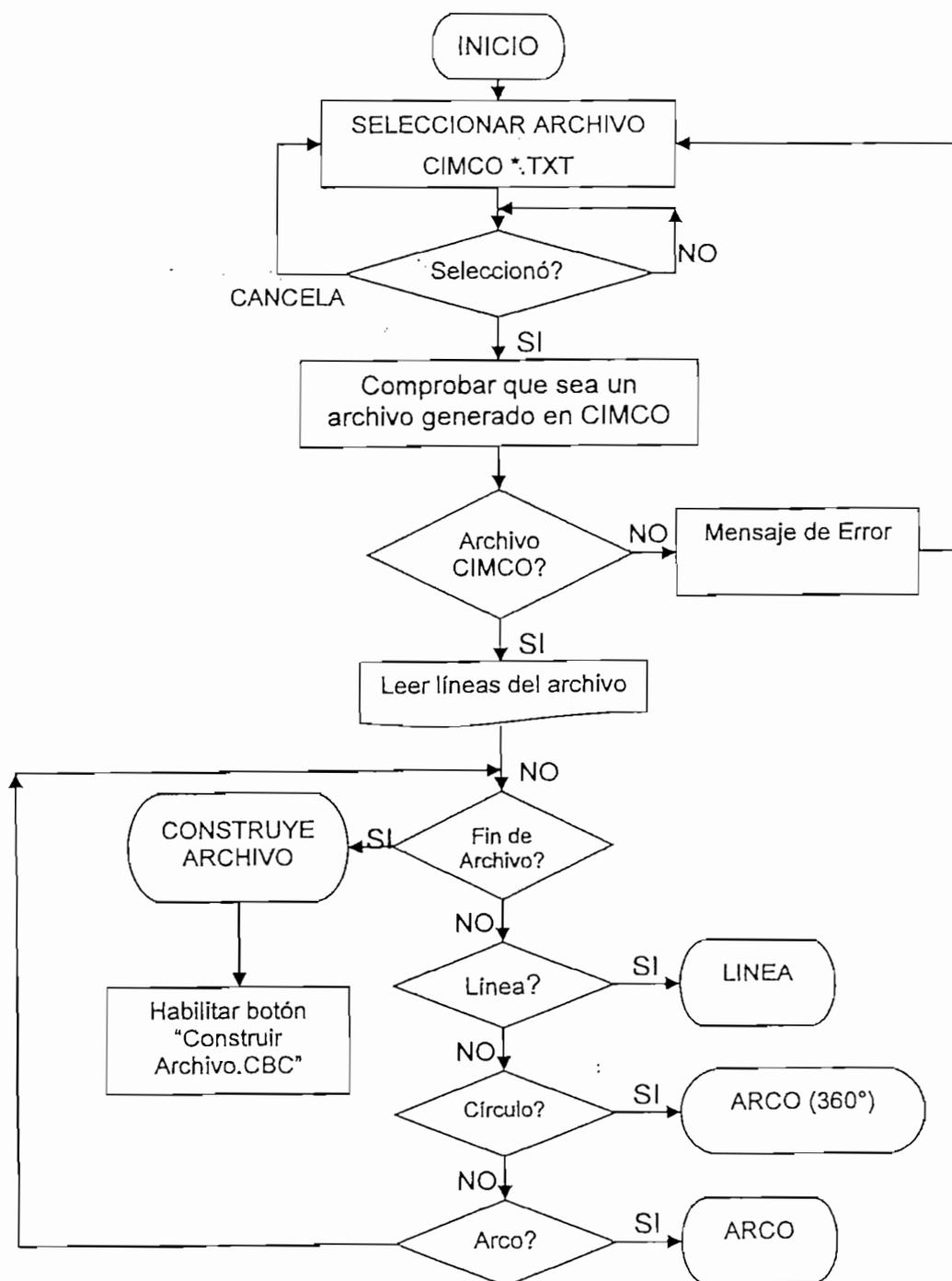
El modo de trabajo automático, que se lo encuentra en el Menú Principal, constituye la parte más importante del programa, debido a que es ésta opción en donde se ejecuta el proceso de corte y se escogen los parámetros óptimos de trabajo en función de los puntos que conforman la figura. Para el diseño de ésta opción se trabajó con los comandos para el microcontrolador CY-545 que en capítulos anteriores se mencionaron.

El objetivo de las subrutinas utilizadas en éste proceso de corte, es la de tomar los datos que proporciona el programa CIMCO V.5, y transformarlos en valores de velocidad y posición, según el tipo de trazo que conforma la figura.

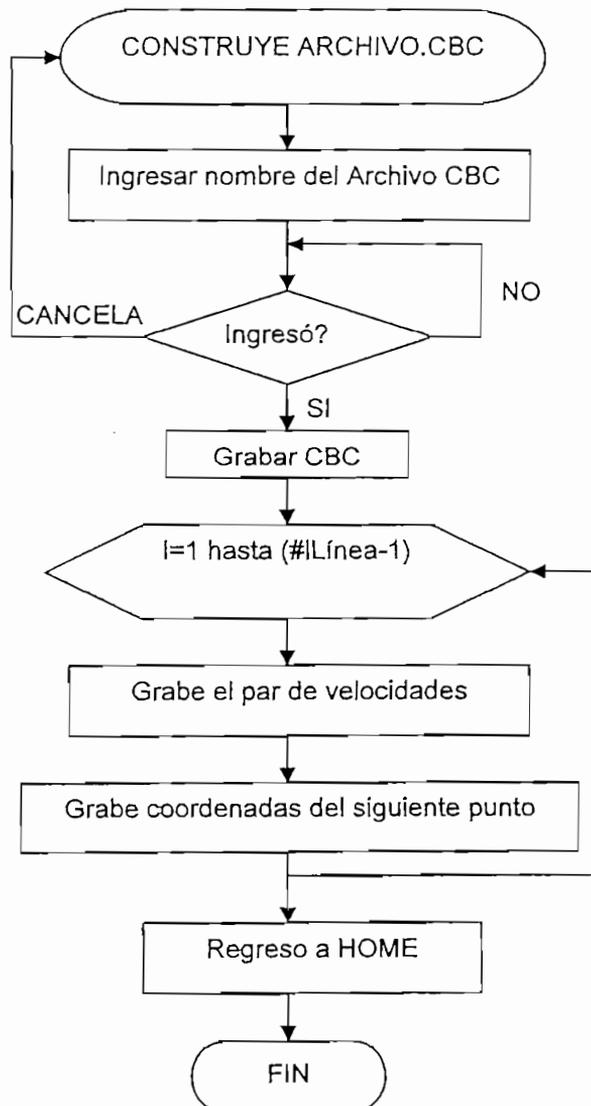
Con estos datos que se los convierte en la opción de Manejo y Conversión de Archivos que se localiza en el Menú Principal, y mediante las subrutinas del

modo automático, se cortará automáticamente una serie de líneas hasta completar el gráfico.

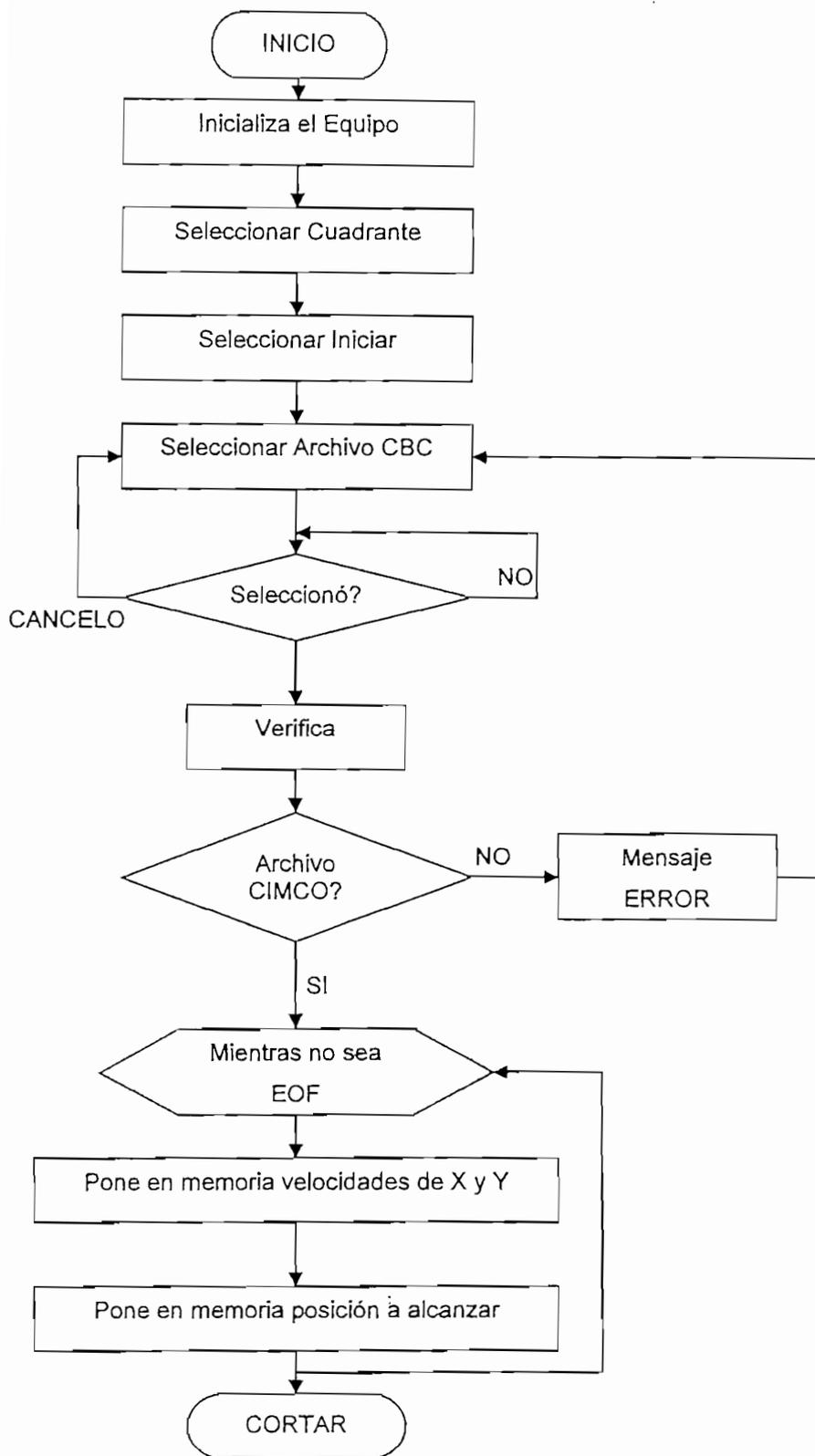
El diagrama de flujo del Proceso de corte Automático y el de Manejo y Conversión de Archivos se puede observa en la siguiente figura.

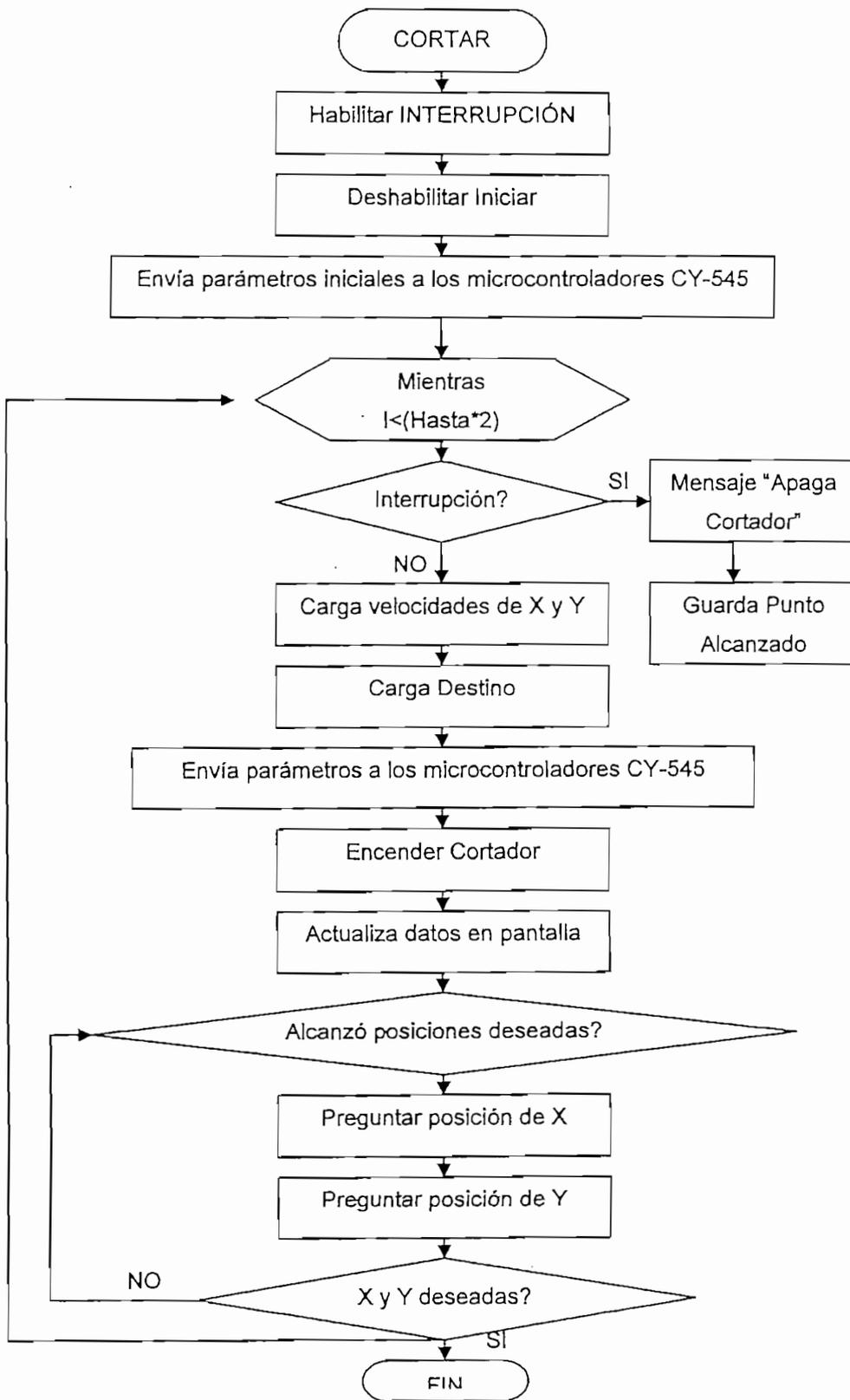


(a)



(b)





(d)

Fig. 3.2 (a) Diagrama de Flujo: (a) Del Proceso Archivo, (b) Del proceso de Construcción del Archivo CBC, (c) Del Proceso de Corte Automático, (d) Del proceso de Cortar.

3.2.1.1 Dibujo. frm.

EL proceso de corte inicia con el **Manejo de Archivos** que se selecciona en el Menú Principal del programa. Al cargarse la opción de manejo y conversión de archivos, se ejecuta la opción "Seleccionar Archivo CIMCO" cuya subrutina lleva el nombre de **CIMCO_MEMORIA**, y parte de su codificación principal es:

```
If Tipo <> "CNC-Calc 2000" Then
    MsgBox "Este archivo no ha sido generado por CIMCO", vbExclamation,
    "Archivo Erróneo"
    Close
    Exit Sub
End If
Input #1, Tipo
If Tipo <> "Text File" Then
    MsgBox "Este archivo no está en formato de Texto", vbExclamation, "Archivo
Erróneo"
```

Que consiste en comprobar si el archivo seleccionado fue generado por el programa CIMCO, en el cual analiza las primeras dos líneas del archivo en caso de so ajustarse al estilo del archivo sale un mensaje de ERROR y si el texto coincide se procede a ejecutar el siguiente proceso. A continuación se dará el formato de un archivo generado en CIMCO y de extensión TXT:

CNC-Calc 2000

Text File

```
CIR (4821.101985558604600,5127.066207097596800,3000.000000000000000)
ARC (6746.571126145939200, 14508.652981004127000,2037.7503440180099,
4.676958355664168, 1.602026043313177)
LIN (6682.943104428157300, 16545.409701724057000, 6398.55870499417870,
16505.911862661538000)
END
```

Al comprobar que el archivo seleccionado es generado por CIMCO, se leen el resto de líneas del archivo para clasificarlas según el tipo y procesar sus datos.

Do While Not EOF(1) 'Mientras no sea Fin De Línea

Input #1, Linea1

If Linea1 = "END" Then Exit Do 'Última línea del Archivo CIMCO

If Left(Linea1, 3) = "LIN" Then ' Si es una LINEA

Linea1 = Right(Linea1, Len(Linea1) - 5) ' Se retira el texto LIN (

Linea1 = Left(Linea1, InStr(Linea1, ".")-1) 'Toma la parte entera del Primer número y lo guarda en la variable X1 que representa la posición inicial en X

Linea1 = Left(Linea1, InStr(Linea1, ".")-1) 'Toma la parte entera del Segundo número y lo guarda en la variable Y1 que representa la posición inicial en Y

Linea1 = Left(Linea1, InStr(Linea1, ".") - 1) 'Toma la parte entera del Tercer Número y lo guarda en la variable X2 que representa la posición final en X.

Linea1 = Left(Linea1, InStr(Linea1, ".") - 1) 'Toma la parte entera del último número y lo guarda en la variable Y2 que representa la posición final en Y

Call Linea(X1, Y1, X2, Y2) 'Llama a la subrutina Línea que se explicará más adelante.

End If

If Left(Linea1, 3) = "CIR" Then ' Si es un CIRCULO

Linea1 = Right(Linea1, Len(Linea1) - 5) ' Se retira el texto CIR (

Linea1 = Left(Linea1, InStr(Linea1, ".") - 1) 'Parte entera del Primer número y lo guarda en la variable X1 que representa el centro en X del círculo

Linea1 = Left(Linea1, InStr(Linea1, ".") - 1) 'Parte entera del Segundo número y lo guarda en la variable Y1 que representa el centro en Y del círculo

```
Linea1 = Left(Linea1, InStr(Linea1, ".") - 1) 'Parte entera del Tercer número
y lo guarda en la variable X2 que representa el valor del radio del círculo
Call Arco(X1, Y1, X2, 0, 360) 'Llama a la subrutina ARCO cuyo ángulo de
giro es de 360° y su respectiva explicación se da más adelante.
```

```
End If
```

```
If Left(Linea1, 3) = "ARC" Then      'Si es un ARCO
    Linea1 = Right(Linea1, Len(Linea1) - 5) 'Se retira el texto ARC (
```

El procedimiento es igual que con el círculo, la única variante es:

```
Input #1, Linea1
Ang_I = CLng((Linea1 * 180) / 3.1416) 'Toma el siguiente número en la
línea que indica el ángulo inicial en rad y se guarda en la variable Ang_I
Input #1, Linea1
Linea1 = Left(Linea1, InStr(Linea1, ",") - 1) 'Toma el último número en la
línea que indica el ángulo final en rad y se guarda en la variable Ang_F
Ang_F = CLng((Linea1 * 180) / 3.1416) 'ANGULO FINAL
Call Arco(X1, Y1, X2, Ang_I, Ang_F)
End If
```

Call Construye_Grafico 'Llama a la subrutina **Construye_Gráfico** para mostrar en la pantalla la figura del archivo que se escogió.

Ésta subrutina, ordena las líneas (busca el inicial de la siguiente recta que es la final de la ya procesada) que se generan en el proceso antes mencionado; seleccionando primero a aquella línea que se encuentra más cercana a HOME (con el comando **Ordenar_Líneas**), éste proceso de ordenar se repite hasta que la última línea que se ordena coincide con la primera. Adicionalmente, al concluir el proceso de ordenar las líneas, se calculan las velocidades óptimas para el movimiento de los ejes.

Por último, se llama a la subrutina *Dibujar* que permite visualizar la figura del objeto a recortar.

Con respecto a las subrutinas *LINEA* y *ARCO*, se calcularán las velocidades óptimas para el proceso de corte mediante el algoritmo:

```

If Difx > Dify Then 'Difx=X2-X1 y Dify=Y2-Y1
  Razon = ((Dify * 15) / Difx) 'Si Difx es menor a Dify se invierten la posiciones
  en todos los cálculos, es decir, en éste caso Razon=((Difx* 15) / Dify)
  RazonTemp = Abs(Round(Razon) - Razon) 'Valor absolute de la resta entre
  redondeo de la razón y el valor real de la razón
  Call Linea_horizontal(X1, Y1, X2, Y1) 'Se traza una línea horizontal si la Razón
  es menor a 1
End If
  FactorTemp = 15
  Factor = 14
  For J = 1 To 14
    Razon = ((Dify * Factor) / Difx)
    If ((Abs(Round(Razon) - Razon)) <= RazonTemp) Or ((Abs(Round(Razon) -
  Razon)) - RazonTemp <= 0.02) Then 'Para encontrar las velocidades óptimas de
  X y Y según la menor diferencia que se calcula
      RazonTemp = Abs(Round(Razon) - Razon)
      FactorTemp = Factor
      Velocidad_Menor = Round(Razon)
    End If
  
```

Si la diferencia en X es menor a la diferencia en Y, el algoritmo se mantiene, con la única diferencia que se traza una vertical, si el valor absoluto de la resta entre el valor aproximado de la razón y el valor real de la razón son menores a 1.

3.2.1.2 Sub Archivo_CBC_Click().

Esta subrutina, tiene como objetivo principal grabar en archivo CBC (Corte Bidimensional por Computadora) los datos generados por el archivo CIMCO.

Al escoger la opción "**Construir Archivo.CBC**", el usuario ingresa el nombre del archivo CBC con el que quiere guardar la figura que se desee recortar; al ejecutar éste proceso, se procede a grabar los datos en el archivo mediante la siguiente codificación:

```
Print #1, CStr(15) + vbTab + CStr(15) 'En la primera línea del archive CBC se
graban las velocidades de 15 pasos/seg en X y 15 pasos/seg en Y, con las cuales
los ejes se desplazaran a la posición más cercana a HOME (posición 0 tanto en X
como en Y),que representa el punto de partida de la primera línea.
```

```
For I = 1 To Indice_Lineas - 1 'Índice_Lineas representa al número de líneas que
conforman al gráfico, además, en cada línea impar se guardarán las velocidades
con las cuales girarán los motores para que los ejes alcancen la posición
requerida, mientras que en las líneas par se guardan las posiciones de destino.
```

```
Print #1, CStr(Matriz_Lineas(I, 1)) + vbTab + CStr(Matriz_Lineas(I, 2)) 'En la
segunda línea se coloca la posición que debe alcanzar la línea trazada, y
representa el inicio de la siguiente línea a trazar.
```

```
Print #1, CStr(Matriz_Lineas(I, 5)) + vbTab + CStr(Matriz_Lineas(I, 6)) 'A
continuación se colocan las nuevas velocidades (calculadas por el programa) con
las cuales se deben alcanzar las siguientes posiciones.
```

```
Next I
```

```
Print #1, CStr(Matriz_Lineas(I - 1, 3)) + vbTab + CStr(Matriz_Lineas(I - 1, 4))
```

```
Print #1, CStr(15) + vbTab + CStr(15) 'En la penúltima línea del archivo CBC
se guardan las velocidades de 15 pasos/seg para cada motor , para que mediante
éstas velocidades el sistema regrese a la posición de Home.
```

Print #1, CStr(0) + vbTab + CStr(50) 'Regresan los ejes a la posición de HOME.

3.2.1.3 Auto.frm.

Para acceder a la opción "**Manejo Automático del Cortador**" (Auto.frm), se debe iniciar el proceso seleccionando el botón para Inicializar al Equipo el cual llama a la subrutina "**Inicio1**" localizada en el **MODULE1.BAS**.

Al llamar a la subrutina Inicio1, los motores llevan a los ejes a la posición HOME; aunque al llegar a esta posición deben alejarse 50 pasos cada eje para que los microcontroladores reciban sin dificultad las órdenes que generaran el movimiento de los ejes.

```
Dato = "W" + "01" + "I" + Chr$(13) ' Inicializa al Microcontrolador
Call Enviar(Forma1)
```

Con el comando antes mencionado, se Inicializa al microcontrolador CY-545 que hace mover al eje X, para inicializar al segundo microcontrolador (maneja al eje Y) se trabaja con la dirección "02". Después de la Inicializar o resetear a los microcontroladores, se envía la orden para que los ejes se dirijan a la posición HOME considerado como origen del sistema. Para ello, se utiliza el siguiente comando:

```
Dato = "W" + "01" + "F 100" + Chr$(13) 'Velocidad para ir a HOME
Call Enviar(Forma1)
```

```
Dato = "W" + "01" + "H 12H" + Chr$(13) ' Señal especificada por el BIT 2
Call Enviar(Forma1)
```

```
Dato = "W" + "02" + "H 04H" + Chr$(13) ' Señal especificada por el BIT 4
Call Enviar(Forma1)
```

Mediante estos comandos, el CY-545 seleccionado mueve a los motores con velocidad constante. La velocidad para el comando HOME se determina dividiendo la velocidad escogida para 20.

Los microcontroladores CY-545 verifican constantemente el valor de los User Bits para determinar cuándo llegan a la posición HOME. Al llegar al origen, los motores se mueven en sentido contrario 50 pasos, y conseguir que no existan conflictos en el Microcontrolador cuando se envían los comandos para generar un movimiento.

Do While True

Dato = "W" + "01" + "O 085H" + Chr\$(13) ' Respuesta se envia al Display

Call Enviar(Forma1)

Dato = "W" + "01" + "? B" + Chr\$(13) 'Se pregunta al primer microcontrolador el valor de los User Bits

Call Enviar(Forma1)

Dato = "W" + "01" + "O 081H" + Chr\$(13) ' Respuesta al Computador

Call Enviar(Forma1)

Ch1 = ""

If Len(Ch1) > 0 Then

Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "=")) 'En Ch1 se coloca la respuesta de los User Bits

Ch1 = Left(Ch1, Len(Ch1) - 1)

End If

If Ch1 = "65499" Or Ch1 = "65503" Then 'Al llegar a HOME X, Ch1 debe igualarse al valor 65499 o 65503 para retroceder 50 pasos.

Este proceso también se lo ejecuta para el microcontrolador de dirección "02", con la única diferencia que Ch1 toma los valores "65499" o "65503".

Al llegar a la posición de HOME, se actualizan los datos en el recuadro "Parámetros", que se encuentra en la pantalla de Manejo Automático del Cortador. Los datos que se muestran en la pantalla son:

- **Velocidad Inicial:** 12
- **Velocidad Final:** 16
- **Pendiente:** 4
- **Dirección:** 01 o 02

Por último, se procede a avanzar 50 pasos del HOME para que no existan conflictos con los comandos que se envían a los microcontroladores CY-545.

Dato = "W" + "01" + "N 50" + Chr\$(13) ' AVANZA 50 PASOS DEL HOME

Call Enviar(Forma1)

Dato = "W" + "01" + "F 12" + Chr\$(13) 'Se carga la velocidad inicial a alcanzar de 12 pasos/seg para el primer microcontrolador.

Call Enviar(Forma1)

Dato = "W" + "01" + "S 4" + Chr\$(13) 'Se carga una aceleración que determina cuán rápido el CY-545 acelera y desacelera cuando se mueve.

Call Enviar(Forma1)

Dato = "W" + "01" + "R 16" + Chr\$(13) 'Se carga la velocidad final a alcanzar de 16 pasos/seg para el primer microcontrolador.

Call Enviar(Forma1)

Dato = "W" + "01" + "-" + Chr\$(13) 'Carga el sentido negativo para que el movimiento sea en sentido contrario a la posición de HOME; cuando se utiliza la dirección "02", que hace referencia a los motores que mueven al eje Y el sentido es "+" para producir el mismo efecto.

Call Enviar(Forma1)

Dato = "W" + "01" + "G" + Chr\$(13) 'Al ingresar los parámetros anteriores se utiliza el comando G para hacer que el motor se mueva.

Call Enviar(Forma1)

Dato = "W" + "02" + "A 00" + Chr\$(13) 'Fija el registro de posición actual como 0.

Call Enviar(Forma1)

Al finalizar el proceso que ejecuta el botón para inicializar el equipo, se habilitan el botón para Seleccionar archivos y para seleccionar el cuadrante de trabajo.

Con respecto al recuadro **Seleccione el Cuadrante**, la opción **Primero** indica que el punto de origen es en la posición 0 (**Desplazamiento_X=0** y **Desplazamiento_Y=0**), mientras que la opción **Segundo** le indica al primer microcontrolador que debe alejarse del origen 17000 pasos (**Desplazamiento_X=17000** y **Desplazamiento_Y=0**).

La parte más importante de la ventana para el **Manejo Automático del Cortador**, es la de "**Seleccionar Iniciar**" (subrutina **Iniciar**), que carga el archivo CBC (seleccionado por el usuario y generado con anterioridad) y discrimina velocidades y posiciones mediante los comandos:

```

Impar = True
Do While Not EOF(1) 'Mientras no es fin del archivo.
    NumPuntos = NumPuntos + 1
    If Impar Then
        ' PAR DE VELOCIDADES
        Input #1, PosX(NumPuntos), PosY(NumPuntos) 'Toma los primeros puntos
como velocidades.
    Else
        ' PAR DE COORDENADAS
        Input #1, PosX(NumPuntos), PosY(NumPuntos) 'Toma los siguientes
puntos como posición a alcanzar.
        PosX(NumPuntos) = PosX(NumPuntos) + Desplazamiento_X
        'Desplazamiento_X, depende del cuadrante que se escoge.
    End If
    If Impar Then
        Impar = False 'Para discriminar entre velocidades y posiciones.
    Else
        Impar = True
    End If

```

Loop

Close #1

NumPuntos = NumPuntos / 2 'Se divide para dos a NumPuntos porque de todos los datos la mitad son velocidades y la otra mitad son posiciones.

Call Cortar(1, NumPuntos)

Mientras se procede a cortar el contorno de la figura seleccionada, se habilita el botón **Interrumpir** y cuando se selecciona esta opción se habilita el botón **Reiniciar**.

Con la opción **Interrumpir**, el proceso de corte se detiene cuando el cortador llega a la posición de destino de la última línea procesada. El dato de la última posición que alcanza el cortador es guardado en memoria, por lo que al acceder a la opción "**Reiniciar**", el proceso de corte se reanuda en aquel punto en el cual se detuvo.

3.2.1.4 Sub Cortar.

La subrutina **Cortar** inicia con los comandos que se muestran a continuación, para fijar el modo de trabajo de ambos CY-545 y toma los datos de memoria para empezar el proceso de corte:

Dato = "W" + "02" + "O 081H" + Chr\$(13)

Call Enviar(Me)

Dato = "W" + "01" + "O 081H" + Chr\$(13)

Call Enviar(Me) ' FIJA MODO DE TRABAJO EN AMBOS CY545

Velocidad_X = PosX(I)

Velocidad_Y = PosY(I)

I = I + 1

X_Destino = PosX(I)

Y_Destino = PosY(I) 'TOMA DATOS DE MEMORIA

Con los datos de velocidades y posiciones finales que se encuentran en memoria, se procede a cargarlos en sus respectivos microcontroladores, con los cuales empieza el corte.

```
Dato = "W02R " + CStr(Velocidad_Y) + Chr$(13) 'Se cargan las velocidades
```

```
Call Enviar(Me)
```

```
Dato = "W02F " + CStr(Velocidad_Y) + Chr$(13)
```

```
Call Enviar(Me)
```

```
FirstRateY.Caption = String(5 - Len(CStr(Velocidad_Y)), "0") +  
CStr(Velocidad_Y) 'Los valores de velocidad tanto para X como para Y se los  
envía a los recuadros de los parámetros para que el usuario observe los cambios  
de velocidades que se dan para cada línea a trazarse (Velocidad Inicial =  
Velocidad Final).
```

Enciende_Cortador 'Se manda la orden para que se encienda el cortador para alcanzar con facilidad las posiciones de destino.

```
Dato = "W" + "02" + "P " + CStr(Y_Destino) + Chr$(13)
```

Call Enviar(Me) 'Se cargan las posiciones que debe alcanzar el cortador mediante el comando P (movimiento a una posición P), es decir, inicia el proceso de corte de la figura seleccionada.

Después de enviar éstos comandos, se procede a verificar si se alcanzó la posición de destino mediante el comando "Dato = "W" + "01" + "? P" + Chr\$(13)", que pregunta acerca de la posición alcanzada por el cortador y lo compara con dato de memoria. Si los datos no coinciden, se sigue enviando el comando de pregunta a través de un lazo hasta que coincida y poder tomar la siguiente posición.

Cuando se alcanzó en ambos ejes de recortar la figura, se apaga el cortador, momento en el cual se acaba el proceso de corte. Es necesario especificar, que a través del programa, al terminar el trazo, el equipo regresa a la posición de origen (HOME).

Si el usuario desea realizar otro corte, debe inicializar nuevamente el equipo, debido a que el resto de opciones quedan deshabilitadas, siendo su única opción la inicialización.

3.3 PROGRAMA PRINCIPAL.

Al ejecutar el programa, se observa como primera pantalla a la presentación o Menú Principal del mismo, el cual contiene las opciones o modos de trabajo con los que se puede trabajar con el equipo de corte.

El programa desarrollado en éste trabajo tiene las opciones de trabajo en Modo Manual y Automático, adicionalmente, tiene las opciones para el Manejo de Archivos y la de finalización del programa.

Los Modos de trabajo Manual y Automático, así como la opción de Manejo de Archivos, constituyen la parte principal de éste programa, debido a que permiten al usuario interactuar con el equipo de corte, y escoger así, el gráfico que se desee recortar.

3.3.1 MENÚ PRINCIPAL.

Una vez que se ha cargado Windows y se ha ejecutado el proyecto *Cortador_Bi_Comp.exe*, el primer menú que se observa es el Menú Principal, que contiene cada una de las partes que conforman el programa. El formulario INICIAL.FRM, es el que contiene al Menú Principal, cuya pantalla de presentación es de la siguiente forma.

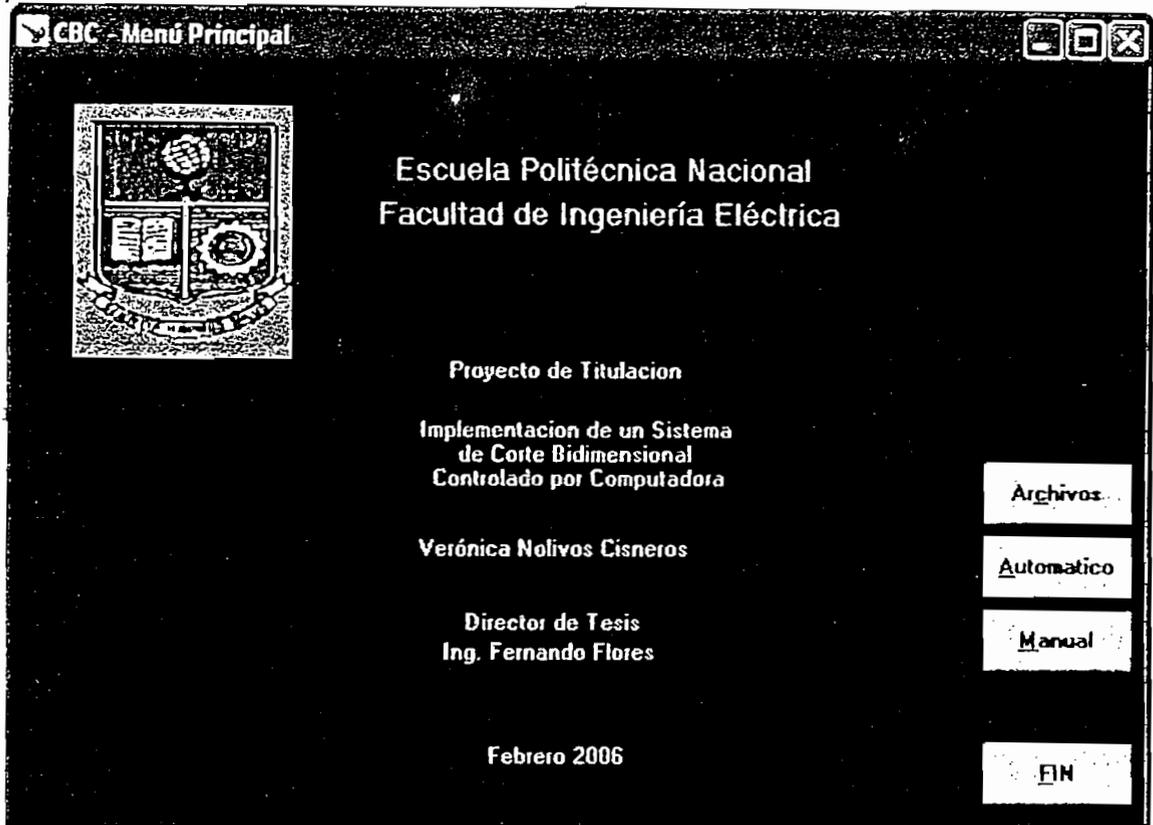


Fig. 3. 3 Menú Principal.

Como se puede observar, a través de éste menú el usuario puede acceder a las opciones de trabajo en Modo Manual, Modo Automático, así como para el Manejo de Archivos y por último, la opción Fin que termina la ejecución del programa.

3.3.2 MANEJO MANUAL DEL CORTADOR.

Este Modo de trabajo, es el que permite al usuario manipular manualmente al equipo de corte NEURACTOR CNC-III. Esto quiere decir, que se puede manejar en forma independiente cada uno de los motores que controlan a los ejes X y Y, con los cuales se trabaja en el proceso de corte.

Cabe mencionar que la comunicación entre el equipo de corte y el computador, así como el control de los motores, requieren de los microcontroladores CY-233 y CY-545 respectivamente; los cuales se encuentran en dos tarjetas, con el fin de manipular separadamente a los ejes X y Y. Por tal razón, en el Modo Manual

deberá escoger con cuál de los microcontroladores se desea trabajar, y cualquier acción que se ejecuta tendrá efecto sobre el microcontrolador escogido.

El formulario que almacena el programa del Modo de trabajo Manual se llama **MANUALF.FRM**, y su presentación es la siguiente:

Fig. 3. 4 Manejo Manual del Cortador.

Como se puede observar, ésta ventana tiene varias partes y opciones, las mismas que serán analizadas para su correcta manipulación.

Un dato importante a mencionar, es que al desplegarse la pantalla del Manejo Manual del Cortador, los únicos botones que pueden manipularse son los de Inicialización del Equipo y de regreso al Menú Inicial. Por lo tanto, para el manejo manual de los motores o ejes, se debe presionar el botón para la Inicialización del Equipo; con lo cual, se observará que el cortador se ubicará en el origen del sistema (los ejes se desplazan a sus respectivas posiciones de Home). Terminado éste proceso, se habilitarán el resto de botones u opciones que presenta la ventana, con lo cual se puede iniciar el manejo manual del equipo.

3.3.2.1 Controles del Display.

Esta opción se localiza en la parte superior izquierda de la pantalla, la cual permite el uso de los display's que se encuentran en las tarjetas que contienen a los microcontroladores antes mencionados.

Cada una de las tarjetas contiene un display, que permite la visualización de un determinado texto compuesto por un número máximo de 8 caracteres; o despliega información referente a los parámetros (posición, velocidad inicial, velocidad final, aceleración, número de pasos, puntero de memoria) con los cuales se han cargado los motores.

Los parámetros visualizados serán los correspondientes al CY-545 escogido, estos parámetros pueden ser:

- Posición
- Max Rate (Velocidad Final)
- Slope (Aceleración)
- First Rate (Velocidad Inicial)
- # Pasos (Número de Pasos)
- Mem Adress (Puntero de Memoria)

Seleccionado el parámetro que se desee observar en el respectivo display (X o Y), es necesario presionar el botón Parámetros que se encuentra a la izquierda de las opciones que da la instrucción necesaria para la visualización.

Debido a que se tiene una tarjeta que controla al eje X y otra que controla al eje Y, el usuario debe escoger el **DISPLAY X** o el **DISPLAY Y**, sobre el cual se observará la información enviada.

Para borrar el display correspondiente al CY-545 seleccionado, se utiliza el botón **Borrar Display** el cual ejecuta la subrutina **Borrar_Display**.

3.3.2.2 Controles.

Se ha diseñado un control que permite al usuario enviar al CY-545 seleccionado comandos que permiten el movimiento de los ejes X y Y en forma manual.

El segundo recuadro que se observa en la pantalla del Manejo Manual del Cortador, hace referencia al uso del motor correspondiente al eje X (su forma lleva el nombre de **Controles**); mientras que el tercer recuadro hace referencia a los controles correspondientes al movimiento del eje Y (**ControlesY**).

Los comandos para el control de movimiento de los ejes X y Y son los mismos, por tal razón, se debe especificar la dirección a la cual se va a enviar la dirección. Al utilizar la dirección de destino "01", la información llega al primer CY-545 correspondiente al eje X, mientras que con la dirección "02", la información es enviada al segundo CY-545 correspondiente al eje Y.

Tanto para el eje X como para el eje Y, se encuentran las opciones correspondientes al sentido de giro de los motores; al escoger el sentido "+", el motor se aleja de la posición Home, mientras que con la opción "-", los motores se acercan a la posición Home.

Existe otro comando que permite la inicialización del CY-545 seleccionado, el cual permite resetear por software al microcontrolador. El botón de inicialización utiliza el comando *I* y la subrutina asociada con este botón es la de **InicializarX** o **Inicializar Y** (según el microcontrolador escogido). Al utilizar éste comando, se fijarán los parámetros por omisión:

- **Posición:** 00000000
- **Velocidad Inicial:** 14
- **Aceleración:** 220
- **Velocidad Final:** 100
- **# Pasos:** 2000
- **Puntero de Memoria:**1

Al ser un Modo de trabajo Manual, el usuario puede modificar los parámetros de control de los motores, actualizando dichos valores con sus botones adjuntos o mediante la opción de Actualización.

A la derecha del botón de Inicialización, se encuentra el botón de Actualización, correspondiente a la subrutina **ActualizacionX** o **ActualizacionY**, y cuya función es la de preguntar al microcontrolador correspondiente, los valores de los últimos parámetros ingresados. Estos valores se mostrarán en los recuadros correspondientes a cada parámetro que se encuentran a la derecha del recuadro.

Con el botón Home, el motor moverá al eje correspondiente hasta el origen. Mediante la subrutina **HomeX** o **HomeY**, que utiliza los comandos "**H 12H**" (para el eje X) y "**H 04H**" (para el eje Y), el microcontrolador sabrá que debe regresar al origen del sistema.

Con el uso del botón **Parámetros Óptimos**, el usuario tendrá valores de parámetros con los cuales los motores se moverán favorablemente, cabe mencionar que éstos valores pueden ser modificados según la necesidad del usuario. Los valores por omisión tanto para el eje X como para el eje Y se indican a continuación:

- **Velocidad Inicial:** 12
- **Velocidad Final:** 16
- **Aceleración:** 4
- **Número de Pasos:**5000
- **Dirección:** +

Existe el botón **Go** (**GoX** o **GoY**) que utilizan el comando **G**, el cual mueve a los motores a la posición (número de pasos) especificada por el usuario manteniendo los parámetros de velocidad inicial, velocidad final, aceleración y dirección especificados con anterioridad.

Por último encontramos a los botones **Borrar Display** y **Encendido y Apagado**; el primer botón tiene la función de borrar el display que ha sido utilizado, así como

el recuadro en el cual se coloca un texto; con respecto al botón Encendido y Apagado se lo utiliza para activar o desactivar al cortador, para lo cual el usuario puede identificar si el mismo está encendido cuando se enciende una luz que se encuentra a la derecha del botón o al observar el led que se haya sobre el eje X.

Cabe mencionar, que el parámetro Posición se actualiza con cada movimiento que realiza el equipo, mostrando el valor de los números de pasos y su equivalencia en centímetros, el cual aumentará o disminuirá dependiendo de la posición a la cual el usuario envíe a los ejes.

Al ejecutar el programa, se observa como primera pantalla a la presentación o Menú Principal del mismo, el cual contiene las opciones o modos de trabajo con los que se puede trabajar con el equipo de corte.

3.3.3 MANEJO AUTOMÁTICO DEL CORTADOR.

Constituye la parte más importante del programa ya que permite recortar en forma continua el contorno de un gráfico específico. En la ventana del modo de trabajo Automático, el usuario no puede controlar ningún parámetro, debido a que el programa calcula y carga automáticamente los parámetros óptimos, con los que se puede trazar los cortes de la figura a recortar.

El formulario en el cual se almacena éste programa es llamado AUTO.FRM y su diseño se muestra a continuación:

CBC - Manejo Automático del Cortador

Porcentaje de Corte

0 50 100

Puntos

Total: 0.00 # Cortado: 0

Nombre del Archivo

Parametros

	Eje X	Eje Y
Vel. Inicial:	0	0
Vel. Final:	0	0
Pendiente:	0	0
Address:	01	02

Seleccionar Iniciar

Seleccione el Cuadrante

Primero Segundo

Presione aquí para inicializar el Equipo

Menu Inicial

CBC - Manejo Automático del Cortador

Porcentaje de Corte

0 50 100

Puntos

Total: 12 # Cortado: 2

Nombre del Archivo

D:\VERO\tesis\vero\abecedario\VERO.CBC

Parametros

	Eje X	Eje Y
Vel. Inicial:	00015	00000
Vel. Final:	00015	00000
Pendiente:	00004	00004
Address:	01	02

Seleccione el Cuadrante

Primero Segundo

Presione aquí para inicializar el Equipo

Fig. 3. 5 Manejo Automático del Cortador antes de Inicializar el equipo, y al seleccionar un archivo para iniciar el corte.

Al igual que en el modo de trabajo Manual, en ésta ventana es necesario cargar o inicializar los datos del equipo para acceder al botón que permite al usuario escoger el archivo de extensión CBC (convertido con anterioridad) que contiene la información para realizar el proceso de corte del contorno de la figura seleccionada.

Los parámetros con los que se carga el equipo son los siguientes:

- **Velocidad Inicial:** 12
- **Velocidad Final:** 16
- **Pendiente:** 4
- **Dirección:** 01 o 02 (Con la dirección 01 los comandos llegan al primer microcontrolador CY-545 que maneja al motor que controla al eje X, mientras que con la dirección 02 los comandos llegan al segundo microcontrolador que mueve al eje Y).

Al Inicializar el equipo, el usuario puede seleccionar el archivo con el cual desea trabajar y el cuadrante sobre el cual se iniciará el corte.

Con respecto al recuadro **Seleccione el Cuadrante**, la opción **Primero** indica que el punto de origen es en la posición 0 (**Desplazamiento_X=0** y **Desplazamiento_Y=0**), mientras que la opción **Segundo** le indica al primer microcontrolador que debe alejarse del origen 17000 pasos (**Desplazamiento_X=17000** y **Desplazamiento_Y=0**).

Al escoger el botón "**Seleccionar Iniciar**" (subrutina de nombre **Iniciar**), se carga el archivo de extensión CBC a memoria, discrimina velocidades y coordenadas, obtiene la información del archivo (velocidad en X, velocidad en Y, posición final en X, posición final en Y) generado por la opción Archivo (Manejo de Archivos) que se ubica en el Menú Principal del programa. Dentro de la subrutina **Iniciar** se llama a otra subrutina **Cortar** que fija el modo de trabajo de cada CY-545 (velocidades y posiciones) encendiendo al cortador hasta que los ejes lleguen a la posición de destino, momento en el cual apaga al cortador hasta leer el siguiente punto de destino.

Al seleccionar el archivo deseado, en el recuadro "**Nombre del Archivo**", se muestra la ubicación del archivo seleccionado (ubicación del archivo CBC generado en la opción Manejo de Archivos).

Cuando el usuario selecciona el archivo CBC que contiene la información para realizar un corte, se habilita el botón **Interrupción** (Fig. 3.5), que permite parar el proceso de corte. Cabe mencionar que aunque el usuario escoja la opción de

Interrumpir, los motores siguen en movimiento hasta alcanzar la posición de destino para culminar la línea procesada, haciendo que el proceso de corte se interrumpe cuando los ejes llegan al destino, con lo cual se guarda en memoria la última posición.

Como se puede observar, el usuario puede escoger interrumpir el proceso de corte cuando se presentan errores que pueden ser corregidos a tiempo; pero existen interrupciones involuntarias, que pueden presentarse en el desarrollo de una ejecución, un ejemplo claro es el corte de energía eléctrica. Éste tipo de interrupciones pueden corregirse con el uso de un UPS que da el tiempo necesario al usuario, para terminar el proceso de corte sin dificultades.

Si el usuario escogió la opción de Interrumpir, también puede acceder a la opción de **REINICIAR** el proceso de corte, el cual inicia en la última posición almacenada en memoria (posición en la que se detuvo el cortador).

3.3.3.1 Parámetros.

Como se puede apreciar, el usuario no cuenta con opciones para manipular o cambiar los parámetros con los cuales se recortará el contorno de una determinada figura, debido a que en el modo de trabajo automático, los parámetros se calcularán directamente por software.

Es importante explicar que para realizar un determinado gráfico, el programa genera una sucesión de líneas que conforman la figura; por lo tanto, se calculan las velocidades que deben adquirir los motores para que se trace con un mínimo de error las líneas.

Estas velocidades que adquieren los motores según el trazo se despliegan en el recuadro "**Parámetros**", en los cuales se informa el cambio de velocidades que sufren los motores con cada cambio de posición (con la que se realizará la siguiente línea).

3.3.3.2 Porcentaje de Corte.

A más de la información de cambios de velocidades, el usuario puede observar el avance del proceso de corte mediante el recuadro "**Porcentaje de Corte**", que indica el porcentaje de avance del proceso, que va en aumento cada vez que los microcontroladores procesan las posiciones a alcanzar para generar una línea. Adicionalmente, se indica el número de puntos que los microcontroladores CY-545 deben procesar (número de posiciones de destino=número de puntos totales) y el número de posiciones alcanzadas (**# Cortado**).

3.3.4 MANEJO DE ARCHIVOS.

Para acceder al modo de trabajo Automático, con anterioridad el usuario debe realizar una conversión del archivo de trabajo, y obtener la información adecuada para que los microcontroladores desplacen a sus respectivos ejes a las posiciones de destino.

Para la conversión de los archivos, en el Menú Principal se encuentra el botón "**Manejo de Archivos**", de nombre **DIBUJO.FRM**. Al escoger esta opción, se despliega una ventana cuyo modelo se muestra a continuación:

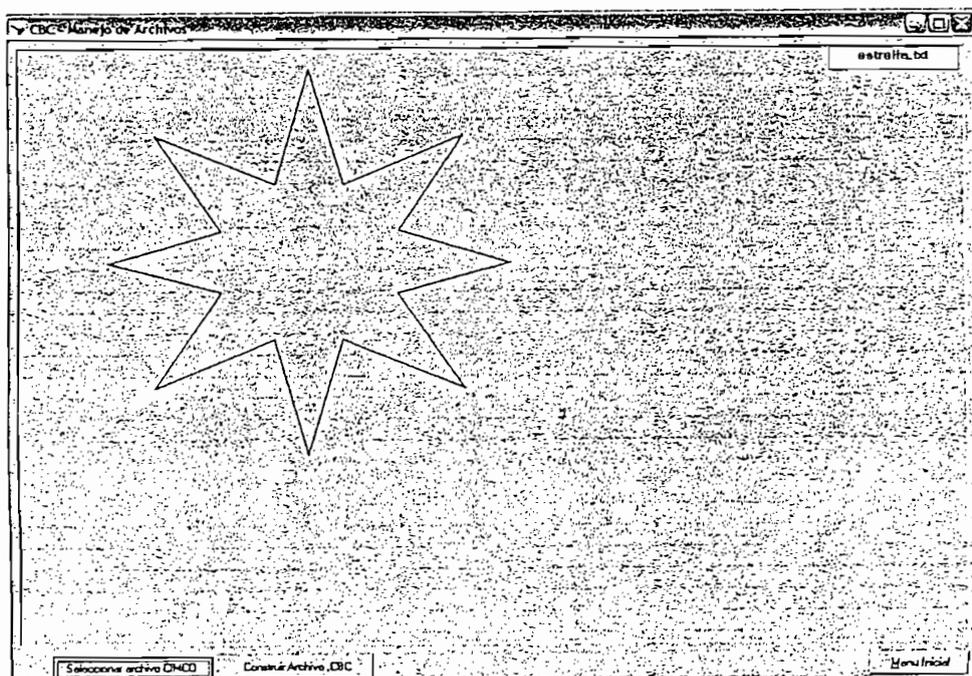


Fig. 3. 6 Manejo de Archivos.

Al aparecer la pantalla "**Manejo de Archivos**", el usuario puede escoger la opción "**Menú Inicial**", el cual regresa a la Pantalla Principal para escoger un determinado Modo de trabajo o para finalizar la sesión.

Otra opción a escoger y que se encuentra en la pantalla antes mencionada es la de "**Selección archivo CIMCO**", (subrutina de nombre **Cimco_Memori**), que permite al usuario escoger el archivo TXT que se generó en el programa CIMCO, y verifica si ésta información lleva el formato de inicio de línea que se muestra a continuación, y así poder ejecutar el siguiente proceso:

CNC-Calc 2000

Text File

En caso de que el archivo lleve el formato del programa CIMCO, se verifican las siguientes líneas, para determinar el tipo de trazo que conforma el gráfico. Los trazos que el usuario genera al realizar un determinado gráfico son: Líneas (LIN), Arcos (ARC) o Círculos (CIR).

A partir de estos datos, el programa transforma éstos puntos en líneas las cuales son graficadas mediante la subrutina **Construye_Grafico**, con lo cual el usuario observa el contorno del gráfico a recortar en la pantalla. Adicionalmente, en la parte superior derecha, se mostrará el nombre del archivo escogido por el usuario (con extensión TXT) y que es mostrado en la pantalla.

En éste momento, se habilita el botón "**Construir Archivo.CBC**" (subrutina de nombre **Archivo_CBC**), el cual tiene como objetivo principal que el usuario ingrese el nombre del archivo CBC con el cual desea grabar los datos del archivo seleccionado en el paso anterior (archivo con extensión TXT y generado por el programa CIMCO y convertido a un archivo de extensión CBC), el cual transforma los puntos que conforman la figura, en líneas: las mismas que requieren datos de velocidad y posición final (se graban en el archivo CBC), los cuales generarán el movimiento cuando se escoja la el modo de trabajo Automático.

CAPÍTULO 4

RESULTADOS EXPERIMENTALES.

CAPÍTULO 4.

RESULTADOS EXPERIMENTALES.

4.1 INTRODUCCIÓN.

En éste capítulo se explicarán las pruebas realizadas con el equipo NEURACTOR CNC-III, así como el análisis de los resultados obtenidos en cada una de las pruebas realizadas.

El objetivo principal del actual capítulo es el de verificar si el equipo NEURACTOR CNC-III recorta las líneas generadas por el programa en los sitios exactos, tomando en consideración, que siempre existen errores debido al movimientos desigual de los ejes y al movimiento de la herramienta de corte (cortador de espuma flex). Cabe mencionar que para comprobar el escalamiento de las figuras, la relación entre el número de pasos y centímetros es de $1/0.00133$ (1 paso/0.524 milipulgadas) .

En el desarrollo y ejecución del cuarto capítulo referente a los resultados obtenidos por los diferentes cortes que se han realizado, se llegó a la conclusión que los gráficos que se desean recortar deben tener un grosor mínimo de 2 centímetros equivalente a unos 2000 pasos. Dicha observación debe considerarse para reducir los posibles errores en el proceso de corte que se dan por la disipación de calor que genera la herramienta de corte.

Con respecto al tamaño máximo que pueden alcanzar las figuras que se desean recortar, debe considerarse la longitud de la herramienta de corte; ya que su reducido tamaño, limita el desplazamiento a 27 cm (20300 pasos) en el eje Y, mientras que en el eje X, se puede alcanzar un desplazamiento equivalente a 36 cm (27000 pasos).

Otro dato importante a considerar para ejecutar el proceso de corte, es que las figuras que se diseñan en el programa CIMCO no deben ser huecas, es decir, los gráficos que se diseñan deben ser periféricos. Con ésta consideración, aseguramos que no existirán cortes que atraviesen el gráfico.

Por último mencionaremos que los materiales escogidos para realizar el proceso de corte, son el espumaflex (principalmente) y el fómix (material difícil para trabajar). Con respecto al segundo material, se debe considerar, que se lo debe templar para realizar el corte, en caso contrario, el resultado será una figura con varias desviaciones.

4.2 RECTANGULO. CDD

Inicialmente, se diseño un arreglo de líneas horizontales y verticales formando un rectángulo de dimensiones 4.17 cm (equivalente a 3136 pasos) por 1.6 cm (equivalente a 1206 pasos), ubicados en el primer cuadrante de la pantalla del programa CIMCO.

El equipo deberá recortar las líneas que conforman al rectángulo en el sitio exacto, a través de los datos que se encuentran cargados en el archivo **Rectangulo.CBC**, los mismos que llevan valores de velocidad que deben adquirir los motores para que el cortador se desplace hasta una posición de destino (punto en el cual se concluye la línea).

El tiempo aproximado que lleva realizar el corte de un rectángulo de 4.17 x 1.6 centímetros (1.6 x 0.63 pulgadas) es de 30 segundos, pero se recomienda aumentar las dimensiones de la figura, para obtener resultados óptimos.

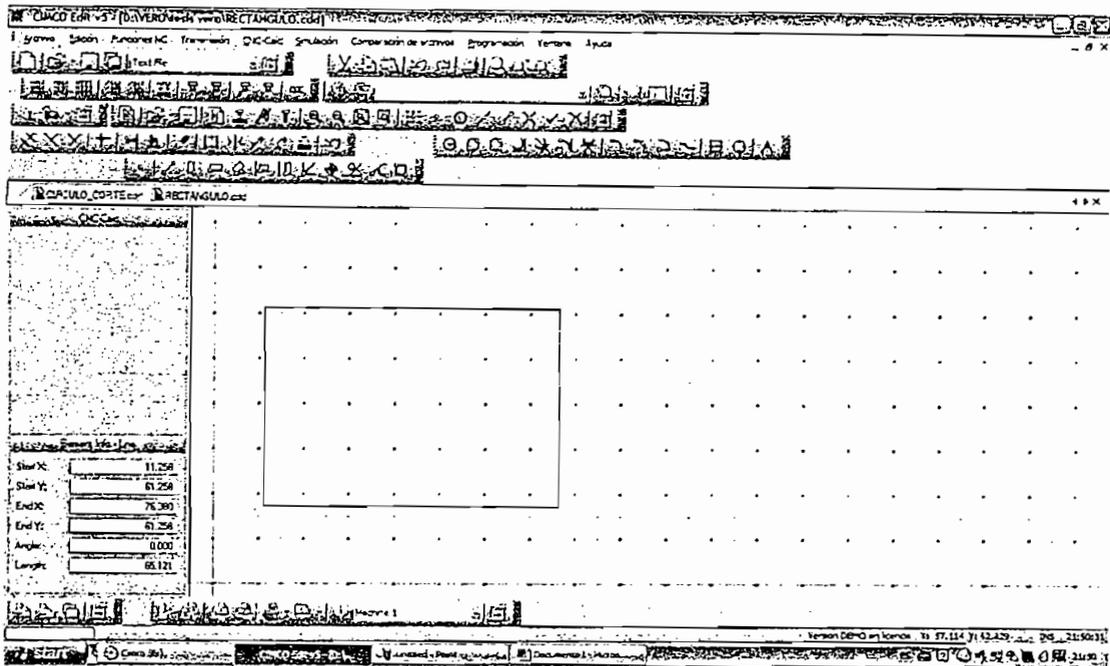


Fig. 4. 1 Rectangulo.cdd.

4.2.1 RECTANGULO.TXT

Finalizado el gráfico del rectángulo, es necesario guardar el archivo con extensión TXT, el mismo que lleva en su interior las posiciones en X y Y iniciales y las posiciones finales de X y Y respectivamente. A continuación se indican los datos generados por el programa CIMCO con el nombre de Archivos.TXT.

CNC-Calc 2000

Text File

LIN (5142.7342417925747, 2634.4161126399508, 2006.8197435114016,
2634.4161126399508)

LIN (2006.8197435114016, 1428.2950991927046, 5142.7342417925747,
1428.2950991927046)

LIN (5142.7342417925747, 1428.2950991927046, 5142.7342417925747,
2634.4161126399508)

LIN (2006.8197435114016, 2634.4161126399508, 2006.8197435114016,
1428.2950991927046)

END

Al ejecutar el programa "Cortador_Bi_Comp", se debe realizar la conversión de los puntos por velocidades y puntos finales para generar líneas sucesivas que parten desde la posición más cercana al origen.

A continuación se muestra aquellos datos generados al hacer la conversión a extensión CBC.

4.2.2 RECTANGULO.CBC

15	15
2006	1428
15	0
5142	1428
0	15
5142	2634
15	0
2006	2634
0	15
2006	1428
15	15
0	50

4.3 ESTRELLA. CDD.

Como segunda prueba a recortar, se diseñó una estrella, la misma que se la dibujo en el primer cuadrante (requisito indispensable, para que el programa interprete correctamente los datos y empezar el proceso de corte).

Las dimensiones de la estrella que se muestra a continuación, es de aproximadamente 15 centímetros, y el tiempo que se tomó para la realización del corte es de un minuto y medio, por lo cual, se puede concluir, que éste ejemplo contiene condiciones óptimas para obtener un resultado confiable.

A continuación, se muestra el esquema del gráfico diseñado mediante el programa CIMCO v.5.

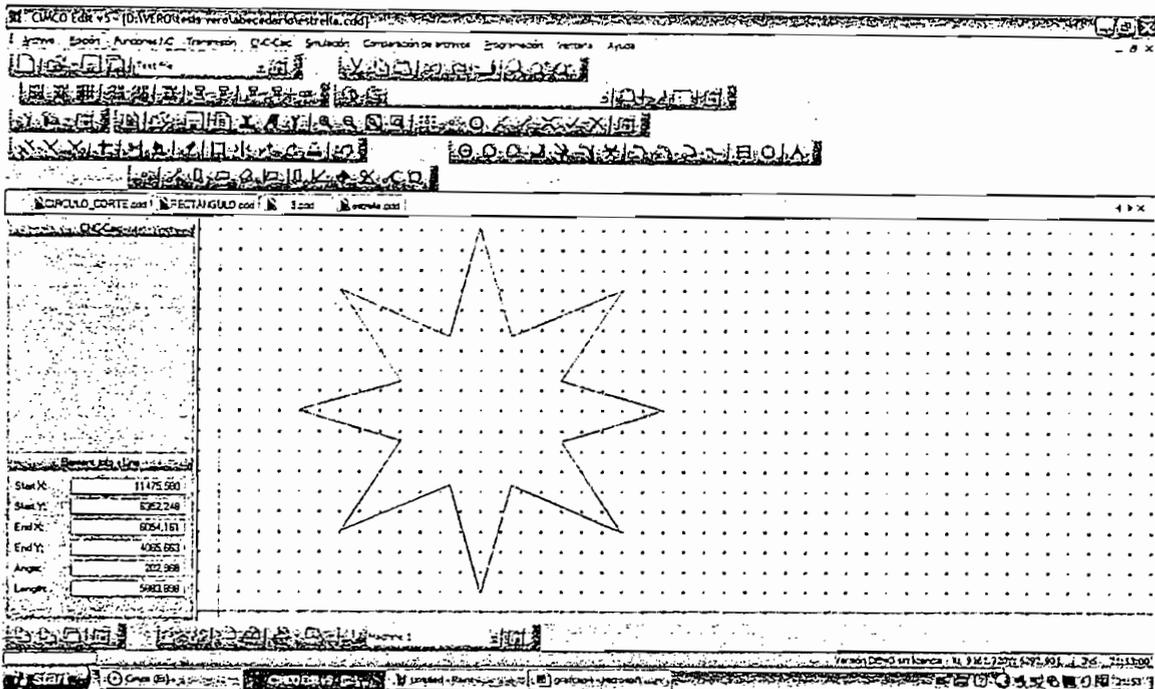


Fig. 4. 2 Estrella.cdd.

4.3.1 ESTRELLA.TXT

Diseñado el gráfico de la estrella, se procede a guardar la información con extensión TXT (Estrella.txt), la misma que se expone a continuación:

CNC-Calc 2000

Text File

LIN (19990.2996765974060, 16030.3740336954890, 14510.9609576835340,
13697.9528246869830)

LIN (19942.7458936514410, 3999.2669608544120, 14529.5974367845670,
6335.077593927197100)

LIN (19990.2996765974060, 16030.3740336954890, 16954.4661733269530,
11476.6237835165880)

LIN (21987.5585603279690, 10038.5973887232520, 16954.4661733269530,
11476.6237835165880)

LIN (21987.5585603279690, 10038.5973887232520, 16994.6787932730880,
8510.16480855826740)

LIN (19942.7458936514410, 3999.2669608544120, 16994.6787932730880,
8510.16480855826740)

LIN (6054.1614186496890, 4065.6628086723140, 9025.1195766552046,
 8558.33123806605540)
 LIN (6009.4874904834687, 16030.3740336954890, 11434.5917927338310,
 13689.4043714557320)
 LIN (12909.4547755831250, 19172.8693019984530, 11434.5917927338310,
 13689.4043714557320)
 LIN (12909.4547755831250, 19172.8693019984530, 14510.9609576835340,
 13697.9528246869830)
 LIN (3964.6748238069413, 10038.5973887232520, 9025.1195766552046,
 8558.33123806605540)
 LIN (3964.6748238069413, 10038.5973887232520, 9026.4621503005528,
 11450.3884102934500)
 LIN (6009.4874904834687, 16030.3740336954890, 9026.4621503005528,
 11450.3884102934500)
 LIN (11475.5801595007580, 6352.2475813516758, 6054.1614186496890,
 4065.66280867231400)
 LIN (12999.8935835404370, 908.27107257508578, 14529.597436784567,
 6335.07759392719710)
 LIN (12999.8935835404370, 908.27107257508578, 11475.5801595007580,
 6352.24758135167580)
 END

4.3.2 ESTRELLA.CBC

A partir de los datos del archivo Estrella.txt, y a través de la conversión del archivo con extensión TXT a CBC ejecutados en la opción "ARCHIVOS" localizado en la pantalla "Menú Principal" (presentación), del programa diseñado en Visual Basic.

En la siguiente tabla, se muestran las velocidades con las cuales los motores deben desplazarse hasta una cierta posición.

15	15
12999	908
4	14
14529	6335
14	6
19942	3999
4	6
16994	8510
13	4
21987	10038
14	4
16954	11476

4	6
19990	16030
14	6
14510	13697
4	14
12909	19172
3	11
11434	13689
14	6
6009	16030
4	6
9026	11450

14	4
3964	10038
14	4
9025	8558
4	6
6054	4065
14	6
11475	6352
4	14
12999	908
15	15
0	50

4.4 CIRCULO_CORTE.CDD.

Con respecto a la construcción de los círculos, los datos que se ingresan son: posición del centro del círculo tanto en el eje X como en el eje Y, y radio del círculo a construir. Adicionalmente, para reutilizar la codificación del programa diseñado en Visual Basic, se añaden los datos Angulo Inicial (Ang_I) y Angulo Final (Ang_F) para asociarlo con la codificación para la ejecución de un ARCO.

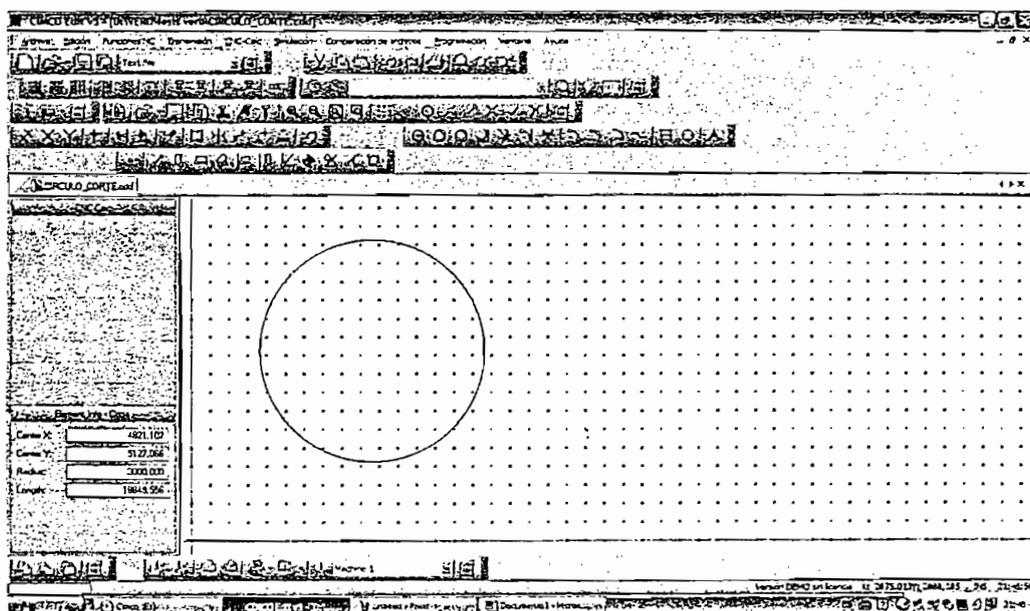


Fig. 4. 3 Circulo_Corte.cdd.

El tiempo estimado para recortar a un círculo de radio 3000 pasos (4 cm o 1.6 pulg) es de 3 minutos.

4.4.1 CIRCULO_CORTE.TXT

Como se explicó anteriormente, la codificación que entrega el programa CIMCO v.5 para el diseño de un círculo contiene los datos de: Posición del Centro del círculo X, Posición del Centro del círculo en Y, Radio del círculo y por software se adjuntan los valores Ang_I=0° (ángulo inicial) y Ang_F=360° (ángulo final).

CNC-Calc 2000

Text File

CIR (4821.101985558604600, 5127.066207097596800, 3000:000000000000000)

END

4.4.2 CIRCULO_CORTE.CBC

Mediante la utilización del programa diseñado en Visual Basic, se convirtió el archivo a un formato reconocido por el programa y de extensión CBC, el cual ha sido llamado CIRCULO_CORTE.CBC, y contiene la siguiente información:

15	15	0	15	8	2	6	14
4647	2132	7801	5475	3961	8001	2140	3781
15	0	2	12	11	4	4	7
4995	2132	7740	5819	3633	7882	2315	3478
8	1	3	10	4	2	6	8
5342	2173	7640	6153	3321	7725	2523	3199
8	2	6	14	6	4	2	2
5681	2253	7502	6473	3030	7533	2762	2945
11	4	4	7	6	5	6	5
6009	2372	7327	6776	2762	7309	3030	2721
4	2	6	8	2	2	6	4
6321	2529	7119	7055	2523	7055	3321	2529
6	4	2	2	6	8	4	2
6612	2721	6880	7309	2315	6776	3633	2372
6	5	6	5	4	7	11	4
6880	2945	6612	7533	2140	6473	3961	2253
2	2	6	4	6	14	8	2
7119	3199	6321	7725	2002	6153	4300	2173
6	8	4	2	3	10	8	1
7327	3478	6009	7882	1902	5819	4647	2132
4	7	11	4	2	12	15	15

7502	3781	5681	8001	1841	5475	0	50
6	14	8	2	0	15		
7640	4101	5342	8081	1821	5127		
3	10	8	1	0	15		
7740	4435	4995	8122	1841	4779		
2	12	15	0	2	12		
7801	4779	4647	8122	1902	4435		
0	15	8	1	3	10		
7821	5127	4300	8081	2002	4101		

Cabe mencionar que para la construcción de círculos y arcos, es necesario generar líneas sucesivas, las cuales dependen del valor del radio del círculo o semicírculo a construir, por tal razón, en el cuadro antes presentado, se observan las velocidades y puntos de destino que deben alcanzar los ejes. Con respecto a la resolución que presenta la figura recortada, se indicará que la herramienta de corte, en este caso el cortador de espumaflex disipa calor, razón por la cual los errores que pueden generarse en la construcción del gráfico se pierden.

4.5 OCHO.CDD.

En el caso del presente ejemplo, se construyó el perfil del número OCHO, que está compuesto por dos arcos. La codificación que presenta el archivo Ocho.txt consiste en el centro tanto en X como en Y del semi-círculo diseñado, así como el ángulo inicial (Ang_I) y ángulo final (Ang_F) de su construcción. Tiempo aproximado para su corte es de 4 minutos.

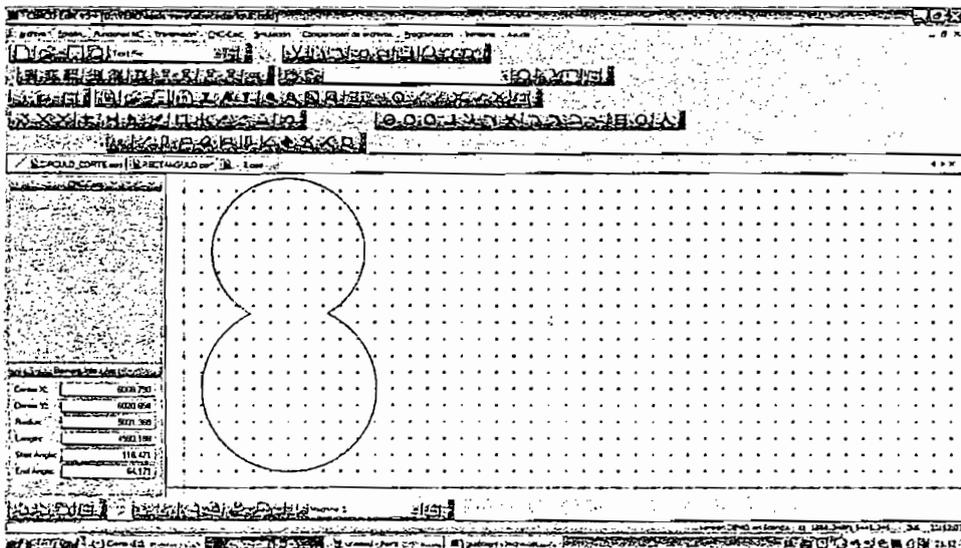


Fig. 4. 4 Ocho.cdd.

4.5.1 OCHO.TXT

Como se puede apreciar en los datos que a continuación se muestran, la información registrada para reconocer que el tipo de gráfico es un ARCO, se observa que en la primera línea se presenta el texto ARC. Los valores que se presentan son: coordenadas del centro del círculo en el eje X, coordenadas del centro del círculo en el eje Y, radio del semi círculo, ángulo inicial del semi círculo y ángulo final del semi círculo.

CNC-Calc 2000

Text File

ARC (6068.7904196542386, 6020.6579763065038, 5031.9694185243061, 2.032799143861561, 1.119997874295723)

ARC (6022.1986095328757, 14337.2960900178240, 4399.6584869533735, 5.246283821380802, 4.189698503956068)

END

4.5.2 OCHO.CBC

Al igual que con los círculos, para la construcción de arcos, se construyen líneas sucesivas, las cuales requieren de los valores de velocidad y destino para completar al semi círculo. A continuación se dan los datos que toman los microcontroladores para el movimiento de sus motores.

Para escoger el punto del cual se inicia el proceso de corte, se construyen todas las líneas (posición de origen y destino que representan a la línea) que conforman al semi círculo, después se procede a escoger de todas las líneas generadas cuál es la más cercana a la posición de origen, comparando en primer lugar, que posición inicial en el eje Y de cada una de las líneas está más cerca del cero, y en caso de tener dos o más líneas en las que coincidan la posición de origen en el eje Y, se procede a comparar las posiciones en el eje X (la más cercana a cero).

15	15	2	14	1	8	13	4	0	15	2	6
6159	990	11016	6930	10408	13996	4574	18491	2799	11344	1325	4342
15	0	2	9	0	15	10	4	0	15	4	10

6508	1008	10941	7271	10421	14345	4249	18363	3046	11097	1453	4017
8	1	4	14	0	15	4	2	13	11	2	4
6854	1051	10843	7606	10407	14694	3936	18210	3312	10872	1603	3702
10	2	3	8	1	8	10	6	7	5	5	9
7197	1117	10721	7933	10364	15040	3635	18032	3595	10668	1774	3398
15	0	5	11	2	10	10	7	10	6	4	6
7534	1207	10577	8251	10295	15382	3350	17831	3863	10542	1966	3107
6	2	7	13	4	14	6	5	15	0	6	8
7864	1321	10412	8558	10199	15718	3081	17609	3554	10378	2178	2829
12	5	7	11	3	8	2	2	8	5	7	8
8186	1456	10225	8853	10076	16044	2831	17365	3258	10193	2409	2567
4	2	8	11	0	15	7	8	11	8	2	2
8497	1614	10019	9135	9928	16360	2601	17103	2976	9988	2657	2322
10	6	11	13	4	7	6	8	6	5	15	0
8796	1793	9793	9401	9755	16664	2393	16823	2708	9765	2921	2094
13	9	2	2	4	6	5	8	2	2	8	6
9083	1992	9550	9651	9559	16952	2207	16527	2457	9523	3201	1886
5	4	10	9	4	5	2	4	10	11	14	9
9355	2211	9290	9884	9341	17224	2045	16218	2222	9264	3495	1697
13	12	9	7	2	2	6	14	7	9	11	6
9611	2448	9014	10098	9102	17478	1909	15897	2007	8989	3801	1529
2	2	6	4	11	10	2	6	4	6	13	6
9850	2702	8725	10292	8843	17713	1798	15566	1811	8701	4117	1383
9	11	7	4	13	10	2	8	7	12	13	5
10071	2973	8288	10566	8566	17926	1714	15227	1635	8399	4444	1258
5	7	6	4	6	4	2	12	2	4	10	3
10273	3257	8579	10758	8274	18116	1657	14883	1481	8086	4778	1157
6	10	9	7	11	6	1	12	4	10	9	2
10454	3556	8855	10972	7967	18283	1627	14535	1349	7763	5118	1080
2	4	11	10	9	4	0	15	2	6	13	2
10614	3866	9113	11207	7648	18424	1626	14186	1239	7432	5463	1026
6	14	0	15	6	2	1	14	2	8	12	1
10753	4186	9351	11462	7319	18540	1651	13838	1153	7094	5810	996
2	6	4	5	15	0	2	13	2	11	15	0
10869	4515	9569	11735	6982	18630	1705	13493	1090	6750	6159	996
3	11	4	6	11	2	2	8	1	9	15	15
10961	4852	9764	12024	6639	18693	1785	13154	1052	6403	0	50
2	10	5	9	10	1	2	6	0	15		
11031	5194	9935	12328	6292	18728	1892	12822	1037	6055		
1	8	0	15	15	0	5	12	0	15		
11076	5540	10082	12644	5943	18735	2025	12499	1047	5706		
0	15	3	8	15	0	2	4	1	10		
11097	5888	10204	12972	5594	18715	2184	12188	1081	5358		
0	15	4	14	14	2	8	13	2	12		
11094	6237	10299	13307	5249	18667	2366	11891	1139	5014		
1	13	2	10	14	3	8	11	2	8		
11067	6585	10367	13650	4908	18593	2571	11608	1220	4675		

Con respecto a la construcción de los arcos, la codificación entrega la posición del centro del semi círculo (en X y Y), el radio del semi círculo, ángulo inicial en radianes y ángulo final en radianes.

CNC-Calc 2000

Text File

ARC (5883.397360179389600, 11906.1356147794700, 4119.8458081952458, 4.811609601909327, 1.653556168507370)

LIN (951.986066082170510, 16046.3131047687890, 951.98606608217051, 994.467830052064530)

LIN (5552.1831687848025, 16046.3131047687890, 951.986066082170510, 16046.313104768789000)

LIN (7981.0872390117911, 994.46783005206453, 10041.9755410225700, 994.467830052064530)

LIN (10041.9755410225700, 994.467830052064530, 6325.0162820388432, 7839.561280167739800)

LIN (3012.8743680929492, 957.666252363270360, 3049.6759449145702, 7655.553391723769900)

LIN (4448.1358641361694, 7655.5533917237699, 3049.67594491457020, 7655.553391723769900)

LIN (4448.1358641361694, 7655.5533917237699, 7981.0872390117911, 994.467830052064530)

LIN (951.98606608217051, 994.467830052064530, 3012.8743680929492, 957.666252363270360)

LIN (4356.2832832560662, 1078.5982772493860, 4353.752515038871900, 1078.614818218578000)

END

4.6.2 R.CBC

Con ayuda de éstos datos, el programa diseñado en Visual Basic, construye un conjunto finito de líneas sucesivas, las mismas que se comparan para determinar cuál se encuentra más cerca del origen (primero se comparan las posiciones en Y

10	6	3	11	5	9	13	4
9060	18223	11207	15153	10619	11453	7626	9199
10	7	2	10	4	6	13	3
9346	18023	11273	14810	10429	11161	7281	9120

4.7 CAMPANA.CDD

A continuación se presenta otro ejemplo que contiene una combinación de arcos y líneas y que lleva la forma de una campana.

Como se puede observar, una aplicación importante que se consigue con el actual Proyecto, es el de generar el contorno de figuras usadas para decoración, manualidades, para días festivos, etc.

Un ejemplo que demuestra lo explicado anteriormente, es el diseño de una campana Navideña, la misma que puede ser decorada y utilizada para complementar la decoración de una fecha festiva. Y como puede apreciarse, a continuación se muestran los datos que entrega el archivo CAMPANA.TXT (que se consiguen a partir del programa CIMCO, a través de la construcción de una determinada figura que se desea recortar), y los valores de velocidades y posiciones que se generan al realizar la conversión a un archivo CAMPANA.CBC.

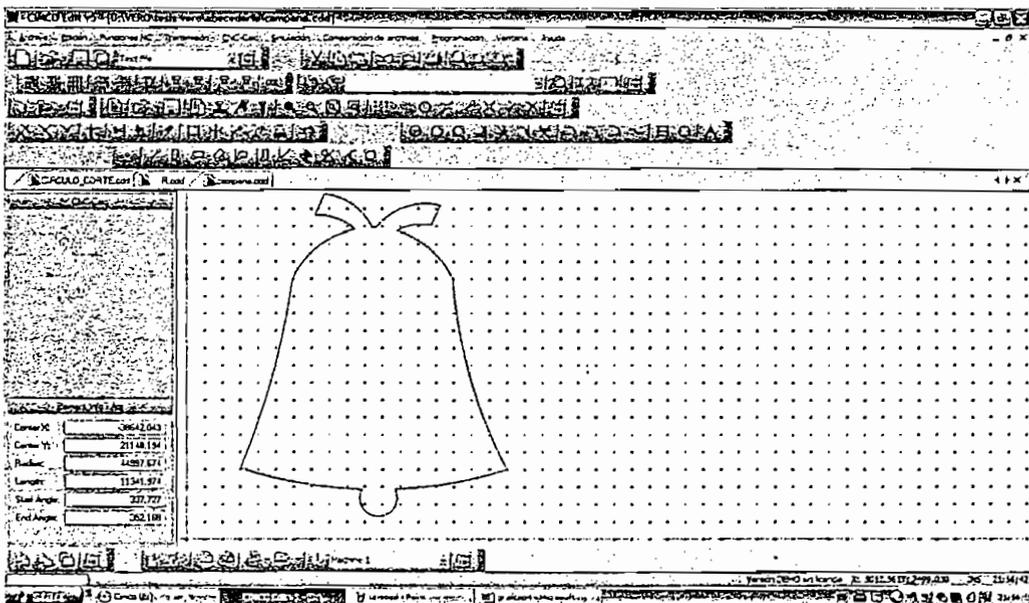


Fig. 4. 6 Campana.cdd.

El corte de una campana de Navidad de una altura de 20 centímetros aproximadamente fue de seis minutos y medio, el problema que se da con éste gráfico, son los lazos de la campana, que por disipación de calor disminuye sus dimensiones (tamaño original de los lazos son muy pequeños).

4.7.1 CAMPANA.TXT

CNC-Calc 2000

Text File

ARC (43659.0978953215600, 17050.1797232747490, 28793.7779059132810, 3.212581187679142, 3.611744842742465)

ARC (-38642.0430419527770, 21148.1939841717980, 44997.6742219109150, 5.894440900886432, 6.146497789603115)

ARC (7491.6446055513607, 16652.2266312445720, 3186.6862014235262, 0.415612119937429, 1.496034326735221)

ARC (10434.9229002048120, 13014.5386650990710, 4924.3914754023981, 1.526194721046324, 1.576457313498491)

ARC (13413.1948523218290, 15818.9245921188520, 3476.2276225511696, 1.330009873341613, 2.487487539233102)

LIN (13703.6754568537040, 18019.9932070985010, 14242.1585648067560, 19194.865593850216000)

LIN (7729.6657640006724, 19830.0112171587210, 7207.4272831701173, 18643.105426140854000)

ARC (7305.1491564553544, 15352.3202996382060, 3292.2357621120377, 0.857270453591119, 1.600483213069320)

ARC (10434.9229002048120, 13014.5386650990710, 4924.3914754023981, 1.770105985703333, 2.722882768949783)

ARC (13167.5341111177790, 15284.2762345472070, 2822.5382714488951, 1.386670874662394, 2.080176392137989)

ARC (10434.9229002048120, 13014.53866509907104 924.3914754023981, 0.416745805983531, 1.291777415582369)

ARC (10626.4564768156050, 26931.2551538786280, 24078.6786147241550, 4.755997239085771, 5.023156516964704)

ARC (10626.4564768156050, 26931.2551538786280, 24078.6786147241550,
4.390027725481899, 4.680035947618672)

ARC (10764.6921061616450, 2367.7020396599091, 1043.3537394700861,
2.644577489051446, 0.508270390473409)

END

4.7.2 CAMPANA.CBC

15	15	6	14	4	6	2	4	7	13	3	8
10710	1325	17107	5913	14408	15921	10242	18258	6016	15188	3335	4941
8	1	4	10	11	14	4	6	4	9	4	10
11055	1366	16974	6236	14192	16195	10050	18550	5893	14714	3208	4616
4	2	4	10	9	10	5	6	2	14	4	10
11368	1517	16845	6560	13958	16454	9827	18818	5841	14368	3185	4031
2	2	5	13	2	2	2	2	2	13	6	2
11614	1762	16720	6886	13705	16694	9577	19061	5787	14023	3518	3926
2	4	3	8	6	5	13	10	2	12	10	3
11765	2075	16598	7213	13436	16917	9301	19274	5731	13679	3852	3825
1	8	5	14	7	5	10	6	2	12	14	4
11806	2420	16481	7542	13152	17120	9003	19456	5672	13335	4188	3730
2	9	2	6	13	8	4	2	2	11	11	3
11815	2882	16368	7872	12855	17302	8687	19605	5610	12991	4525	3639
15	0	2	6	4	2	6	2	2	10	8	2
12164	2902	16259	8204	12545	17463	8357	19718	5545	12648	4863	3553
14	1	2	6	14	6	9	2	2	10	8	2
12512	2927	16153	8537	12224	17601	8017	19794	5478	12306	5203	3472
11	1	3	10	6	2	9	1	2	10	9	2
12860	2957	16052	8871	12000	17853	7729	19830	5408	11964	5544	3395
10	1	4	14	4	2	4	9	3	14	10	2
13207	2992	15955	9206	12000	17853	7217	18643	5335	11622	5885	3324
9	1	3	11	13	5	15	0	2	9	10	2
13554	3032	15862	9543	12326	17978	7566	18634	5260	11282	6228	3258
8	1	2	8	8	2	15	0	3	13	12	2
13900	3077	15773	9880	12664	18061	7912	18588	5182	10941	6572	3197
14	2	2	8	8	1	8	2	2	8	12	2
14245	3127	15688	10219	13011	18102	8251	18505	5102	10602	6916	3141
12	2	2	8	15	0	11	4	2	8	14	2
14590	3182	15607	10558	13360	18099	8579	18387	5018	10263	7262	3089
12	2	2	9	8	1	4	2	2	8	15	0
14934	3241	15531	10899	13703	18019	8893	18235	4933	9924	7607	3043
10	2	3	14	5	11	8	5	2	8	8	1
15277	3306	15458	11240	14254	19191	9190	18051	4844	9587	7954	3002
10	2	2	10	10	2	9	7	3	11	9	1
15619	3376	15390	11583	13912	19258	9494	17848	4753	9250	8301	2965
9	2	2	10	10	1	13	3	3	11	11	1
15960	3451	15326	11926	13564	19291	9154	17769	4660	8913	8649	2934
8	2	2	12	15	0	13	4	4	14	13	1

16300	3531	15266	12270	13215	19288	8821	17666	4564	8578	8997	2908
8	2	2	12	9	1	13	5	3	10	15	0
16638	3616	15210	12614	12868	19251	8495	17540	4465	8243	9345	2887
8	2	2	14	10	2	4	2	4	13	15	0
16976	3705	15158	12959	12527	19179	8180	17392	4363	7909	9843	2857
14	4	2	14	6	2	9	5	2	6	2	6
17312	3800	15111	13305	12194	19073	7875	17221	4259	7576	9733	2527
10	3	1	8	14	6	6	4	2	6	0	15
17646	3899	15067	13652	11874	18935	7584	17029	4153	7243	9738	2180
6	2	1	9	9	5	8	6	2	6	4	11
17987	4013	15028	13998	11569	18765	7306	16817	4044	6912	9856	1853
2	4	1	10	10	7	9	8	2	6	9	11
17830	4325	14993	14346	11283	18565	7045	16586	3932	6581	10076	1583
2	4	1	11	7	6	2	2	2	6	13	8
17678	4639	14963	14693	11019	18338	6800	16337	3818	6251	10371	1401
2	4	1	13	2	2	6	7	5	14	9	2
17529	4955	14932	15017	10778	18085	6574	16071	3701	5922	10710	1325
5	11	2	4	7	9	6	8	4	11	15	15
17385	5273	14779	15330	10692	17931	6367	15790	3581	5594	0	50
4	9	4	7	15	0	5	8	3	8		
17244	5592	14604	15632	10402	17948	6181	15495	3459	5267		

4.8 RESULTADOS GENERALES.

Las primeras pruebas realizadas con el software desarrollado en Visual Basic, ayudaron a encontrar las dimensiones apropiadas para el diseño de figuras.

Al realizar varios cortes de letras, números y figuras básicas, se llegó a la conclusión de que el grosor (ancho) mínimo de un determinado gráfico es de 2 centímetros (0.79 pulgadas), equivalente a 2000 pasos (valor fundamental en la ejecución del programa). Si el usuario mantiene esta condición, el error que se comete por la herramienta de corte (disipación de calor) y por el desplazamiento de los ejes X y Y generado por el movimiento de los motores (si el primer comando que se envía hace referencia al eje X hará que su respectivo motor se mueve antes que los que controlan al eje Y), será mínimo o imperceptible.

Pero, el valor del grosor (ancho de la figura) no es el único parámetro importante a considerar para el diseño de las figuras a recortar, también debe considerarse el tamaño máximo y mínimo que deben tener las figuras. Debido a la longitud de la

herramienta de corte y el tamaño de los ejes, se determinó que las dimensiones máximas que deben tener las figuras son de 27 centímetros (10.63 pulgadas) en el eje Y o expresado en número de pasos equivale a 20300; y de 36 centímetros (14, 17 pulgadas) en el eje X equivalente a 27000 pasos.

Con respecto al tiempo que toma la ejecución del proceso de corte de las figuras, es importante mencionar, que el tiempo depende de la complejidad de la figura y el tamaño del mismo.

Para el desarrollo del presente capítulo, se realizó el corte de cuadrados y círculos de diferentes dimensiones, para encontrar la relación tiempo de corte vs dimensión de la figura; adicionalmente, permite determinar la complejidad de un corte, según la composición de sus trazos (línea horizontales, líneas verticales, diagonales, arcos, círculos).

De los resultados obtenidos, se determinó que las líneas horizontales y verticales requieren cortes sencillos, rápidos y finos, debido a que el movimiento se da en un solo eje a la máxima velocidad (15 pasos/segundo). Los gráficos más complejos a recortar, son aquellos conformados por arcos o círculos, debido a que por interpolación lineal, los arcos y círculos se transforman en un conjunto de líneas sucesivas (horizontales, verticales y diagonales), cuyas velocidades óptimas de corte se las determina por software, lo cual hace lento al proceso de corte.

Con respecto a la relación entre el tiempo de corte y la dimensión de la figura, se escogió a los cuadrados y círculos como figuras para analizar, debido a que los primeros son los más simples, mientras que los círculos (arcos) son las figuras más complejas a recortar.

Por ejemplo, al recortar un círculo de radio 15000 pasos, el tiempo para ejecutarlo fue de un minuto y medio, mientras que para recortar un círculo de radio doble (30000 pasos) se demoró aproximadamente 3 minutos. Con respecto a los

cuadrados, el efecto es el mismo, existe una relación directa entre el aumento de las dimensiones, con el tiempo para la ejecución del proceso de corte.

Como se ha mencionado anteriormente, los gráficos que consumen un periodo de tiempo alto para la ejecución de un corte, son aquellos conformados por arcos y círculos, y sus dimensiones oscilaran en el límite superior. El tiempo aproximado para realizar un gráfico con las características antes mencionadas (peores condiciones) es de aproximadamente ocho minutos.

Con estos datos, se concluye que bajo las peores condiciones, en veinte y cuatro horas, se producen aproximadamente 180 figuras (campana de Navidad). Esto nos lleva a buscar formas para optimizar el proceso de corte y disminuir el tiempo de corte, Para ello se recomienda diseñar gráficos con un grado de complejidad bajo, y cuyas dimensiones no sobrepasen los 15000 pasos (20 centímetros o 7.86 pulgadas) tanto para X como para Y.

CAPÍTULO 5.

CONCLUSIONES Y RECOMENDACIONES.

5.1 CONCLUSIONES GENERALES.

En el transcurso de los dos últimos siglos la manufactura y automatización han sido los motores del desarrollo económico, debido a que estos procesos se encargan de la fabricación de bienes y artículos, controlados y operados por maquinarias.

Control numérico (CN) es todo dispositivo capaz de dirigir posicionamientos de un órgano mecánico móvil, en el que las órdenes para el desplazamiento son elaboradas por las instrucciones codificadas (combinación de letras y números) en un programa, haciendo que las máquinas desarrollen su trabajo automáticamente.

El corazón de un sistema CNC (Control Numérico por Computadora) es el computador que se encarga de realizar los cálculos necesarios y las conexiones lógicas, mientras que el sistema CNC es el puente de unión entre el operador y la máquina-herramienta.

Los CNC incluyen una memoria interna que permite el almacenamiento del programa de la pieza, de los datos de la máquina y de las compensaciones de las herramientas.

El programa diseñado en Visual Basic 6.0 comienza definiendo la secuencia óptima de operaciones y las condiciones del proceso.

En el caso de sistemas de referencia fijos (utilizado en el presente Proyecto de Titulación), el origen siempre se localiza en la misma posición con respecto a la

mesa de trabajo. Normalmente, esta posición es la esquina inferior de la izquierda de la mesa de trabajo y todas las posiciones se localizan a lo largo de los ejes XY positivos y relativos a ese punto fijo de referencia.

Las curvas son producidas con la interpolación lineal interrumpiéndose por trazos pequeños de líneas rectas. Este método tiene limitaciones porque es necesario programar un número muy grande de puntos para describir la curva en orden para producir la forma del contorno.

La programación de un contorno con interpolación lineal requiere la coordenada de las posiciones para el inicio y fin de cada segmento de línea. Por consiguiente, el punto extremo de una línea o segmento se convierte en el punto de salida del siguiente segmento, y así sucesivamente, a lo largo de todo el programa.

El microcontrolador de motores a pasos CY-545 ejecuta cada comando luego de recibido, es decir, que el CY545 irá a ocupado por un periodo largo luego de recibido el último carácter de un comando. Por tal razón, por momentos el usuario notará que los movimientos de los ejes se paran cuando se envía el dato para el corte de la siguiente línea.

El microcontrolador CY545, se comunica con el computador a través del pórtico serial mediante un Controlador Inteligente de Redes Locales CY233, es decir, el CY233 espera por un mensaje serial de la computadora antes de iniciar la transferencia paralela de datos hacia el microcontrolador CY-545.

La principal función del CY233 es la de transferir información entre dos sistemas separados y asincrónicos en tiempos multiplexados o con característica serial, manejando el flujo de datos que se transfiere entre varios sistemas.

Los niveles de voltaje del CY233 son TTL, siendo necesario adaptarlos a niveles RS-232, para ello, se utiliza el integrado MAX-233, que posee un circuito interno de generación de voltaje y dos juegos de receivers y drivers.

Debido a las diferentes pruebas que se realizaron con diferentes herramientas de corte como bisturí, estilete, cortador de espumaflex y por la información recaudada para el uso de un laser de baja potencia, se llegó a la conclusión de que la mejor herramienta de corte por costos y facilidad para su manipulación es el cortador de espumaflex.

De los resultados obtenidos, se determinó que las líneas horizontales y verticales requieren cortes sencillos, rápidos y finos, debido a que el movimiento se da en un solo eje a la máxima velocidad (15 pasos/segundo). Los gráficos más complejos a recortar, son aquellos conformados por arcos o círculos, debido a que por interpolación lineal, los arcos y círculos se transforman en un conjunto de líneas sucesivas (horizontales, verticales y diagonales), cuyas velocidades óptimas de corte se las determina por software, lo cual hace lento al proceso de corte.

5.2 RECOMENDACIONES.

Se recomienda comprar una mesa de granito para montar el equipo, y lograr así la disminución de errores que se producen por los cambios de temperatura y humedad.

Se recomienda mantener la forma de la pieza lo más sencilla que se pueda para reducir el número de procesos a seguir, o en caso contrario, pueden usarse formas complejas si aquello permite la consolidación de varias partes y/o la eliminación de uno o más pasos de manufactura.

Es necesario que el usuario tenga la precaución de construir contornos o perfiles de figuras, ya que si no se cumple con ésta condición, al ejecutarse el proceso de corte, la herramienta atravesará la figura para realizar aquellos cortes internos, con lo cuál, la figura se rompería.

Para que los motores que controlan los movimientos de los ejes se desplacen favorablemente, fue necesario encontrar valores de velocidad inicial, final y

pendiente, los mismos que bordean los valores de 8 a 16 pasos por segundo. Adicionalmente, no debe existir una variación de velocidades muy grandes.

Se recomienda que el usuario realice los contornos de los gráficos a recortar en el primer cuadrante de la pantalla del programa CIMCO, debido a que el software no reconoce posiciones negativas ya que trabaja con un sistema de referencia fijo, en donde el origen se ubica en la parte inferior izquierda del tablero de los ejes.

Debido a que el microcontrolador CY-545 ejecuta cada comando luego de recibido, por lo que se tienen el CY545 irá a ocupado por un periodo largo luego de recibido el último carácter de un comando, se recomienda eliminar todas aquellas líneas o comandos dentro del software que no ayudan o no ejecutan ningún proceso, ya que éste exceso de líneas hace más lento el proceso de lectura de datos por parte del CY-545.

Se recomienda utilizar el programa CIMCO v.5 para el diseño de gráficos y para la adquisición de los puntos que lo conforman para que posteriormente sean transformados en valores de líneas (Archivo.CBC), debido que al realizar diferentes pruebas con los programas COREL DRAW y AUTOCAD, los datos que proporcionan éstos no satisfacen las necesidades para transformar los datos por la complejidad de la información entregada por éstos dos últimos programas.

Con respecto al cortador de espumaflex, es necesario implementar un sistema que sostenga a la figura que se recorta en el espumaflex, debido a que al ser una herramienta que corta por calor, la figura no queda sujeta a nada. Por tal razón, el usuario visualizará una plancha de cartón prensado, para sostener a la figura recortada.

Se recomienda que los gráficos que se desean recortar deben tener un grosor (ancho) mínimo de 2 centímetros (0.78 pulg) equivalente a dar 2000 pasos, con lo cual se reducen los posibles errores que se dan en el proceso de corte por disipación de calor generado por la herramienta de corte.

Pero, el valor del grosor (ancho de la figura) no es el único parámetro importante a considerar para el diseño de las figuras a recortar, también debe considerarse el tamaño máximo y mínimo que deben tener las figuras. Debido a la longitud de la herramienta de corte y el tamaño de los ejes, se determinó que las dimensiones máximas que deben tener las figuras son de 27 centímetros (10.63 pulgadas) en el eje Y o expresado en número de pasos equivale a 20300; y de 36 centímetros (14, 17 pulgadas) en el eje X equivalente a 27000 pasos.

Se concluye que bajo las peores condiciones de corte, la producción de un determinado gráfico baja, calculando que en veinte y cuatro horas, se producirían aproximadamente 180 figuras. Esto nos lleva a buscar formas para optimizar el proceso de corte y disminuir el tiempo de corte, Para ello se recomienda diseñar gráficos con un grado de complejidad bajo, y cuyas dimensiones no sobrepasen los 15000 pasos (20 centímetros o 7.86 pulgadas) tanto para X como para Y.

BIBLIOGRAFÍA

1. SCHEY, John A., "Procesos de Manufactura", McGRAW-HILL, Tercera Edición.
2. CORDOVA, Francisco, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos", Noviembre 1996.
3. COMER, Douglas E., "Redes Globales de información con Internet y TCP/IP", Tercera Edición.
4. AGUIAR URBINA, Byron Hernán, "Motor de Pasos: Control de Posición y Velocidad mediante un microcontrolador INTEL 8751", Quito 1996.
5. U.S. CYBERLAB, INC, "Neuractor CNC-III".
6. STEVE KRAR, Arthur Gill, "Computer Numerical Control Programming Basic", Industrial Press, 1999.
7. J. SABATER, "Control Numérico", Universitat Miguel Hernandez, Enero 2001.
8. CASS, "Manufactura Integrada Por Computadora", Enero 2004.
9. HIDALGO, Pablo, "Comunicación Digital", Quito, 2005.
10. ECG SEMICONDUCTORS, "Master Replacement Guide", Philips ECG, Barcelona, 1981
11. MICROSOFT, "Visual Basic help".
12. DAGA Pedro, "Transformadores Convertidores", Primera Edición, Ediciones CEAC S.A., Barcelona España 1974.
13. FITGERALD A, "Máquinas Eléctricas", Sexta Edición, Mc Graw Hill, México 1997.
14. Ras Enrique, "Transformadores de Potencia de Medida y de Protección", Séptima Edición, Alfaomega - Marcombo, 1991.
15. DORF Richard, "Circuitos Eléctricos", Segunda Edición, Alfaomega, México 1995.
16. KOSOW I, "Maquinas Electricas y Transformadores", Segunda Edición, Prentice may, México 1993
17. VEINOT, Cyril G., "Motores Eléctricos de Potencia Fraccionaria y Subfraccionaria", 1978

BIBLIOGRAFÍA ELECTRÓNICA

18. http://www.sapiensman.com/motores_por_pasos/
19. <http://www.arcelect.com/rs232.htm>
20. <http://www.doc.ic.ac.uk/~ih/doc/stepper/>
21. <http://crm.ii.uam.es/web/index.php?seccion=4&pagina=2>
22. http://es.wikipedia.org/wiki/Motor_paso_a_paso
23. <http://www.redeya.com/electronica/tutoriales/mpp/mpp.htm>
24. <http://www.controlchips.com/manuals/cy545/545-15.htm>
25. <http://www.controlchips.com/cy233cmd.htm>
26. <http://www.ortodoxism.ro/datasheets/maxim/MAX220-MAX249.pdf>
27. http://www.ortodoxism.ro/datasheets/philips/74HC_HCT244_CNV_2.pdf
28. http://www.tranzistoare.ro/datasheets/270/387520_DS.pdf
29. http://www.ortodoxism.ro/datasheets/philips/74HC_HCT04_CNV_2.pdf
30. <http://www.ortodoxism.ro/datasheets/philips/74hc373.pdf>
31. <http://www.doc.ic.ac.uk/~ih/doc/stepper/>
32. <http://crm.ii.uam.es/web/index.php?seccion=4&pagina=2>
33. http://es.wikipedia.org/wiki/Motor_paso_a_paso
34. <http://www.redeya.com/electronica/tutoriales/mpp/mpp.htm>
35. <http://www.cimco-software.com>
36. <http://cache.national.com/ds/LM/LM7512C.pdf>
37. <http://cstep.luberth.com/>
38. <http://www.cs.uiowa.edu/~jones/step/>
39. <http://www.todorobot.com.ar>
40. <http://www.agapea.com/MOTORES-ELECTRICOS-ACCIONAMIENTO-DE-MaQUINAS-30-tipos-de-motores-n218773i.htm>
41. <http://www.etsit.upm.es/departamentos/teat/asignaturas/lab-ingel/motores%20asincronos%20monofasicos.pdf>
42. <http://www.minas.upm.es/dep/Sistemas-Energeticos/Tema3-2.PDF>

ANEXOS

CAPÍTULO I

ANEXO 1.1

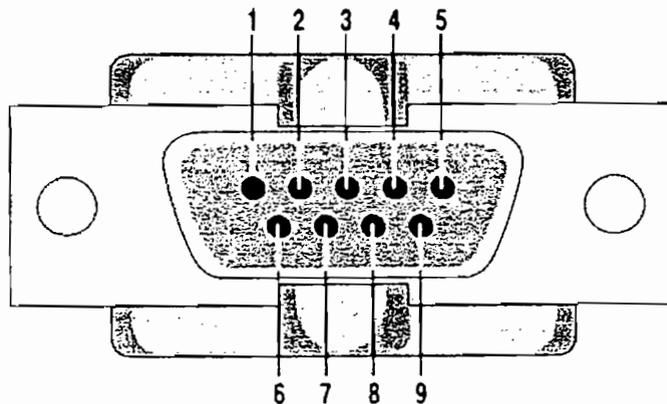
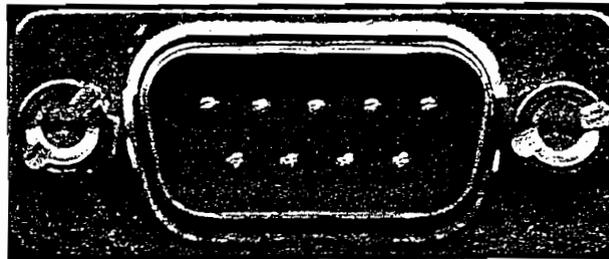
TRANSMISIÓN SERIAL.

CABLE SERIAL

This is a standard 9 to 25 pin cable layout for async data on a PC AT serial cable

Description	Signal	9-pin DTE	25-pin DCE	Source DTE or DCE
Carrier Detect	CD	1	8	from Modem
Receive Data	RD	2	3	from Modem
Transmit Data	TD	3	2	from Terminal/Computer
Data Terminal Ready	DTR	4	20	from Terminal/Computer
Signal Ground	SG	5	7	from Modem
Data Set Ready	DSR	6	6	from Modem
Request to Send	RTS	7	4	from Terminal/Computer
Clear to Send	CTS	8	5	from Modem
Ring Indicator	RI	9	22	from Modem

This a DTE port as on the back of a PC Com Port - EIA-574 RS-232/V.24 pin out on a DB-9 pin used for Asynchronous Data



Pin	Signal	Pin	Signal
1	Data Carrier Detect	6	Data Set Ready
2	Received Data	7	Request to Send
3	Transmitted Data	8	Clear to Send
4	Data Terminal Ready	9	Ring Indicator
5	Signal Ground		

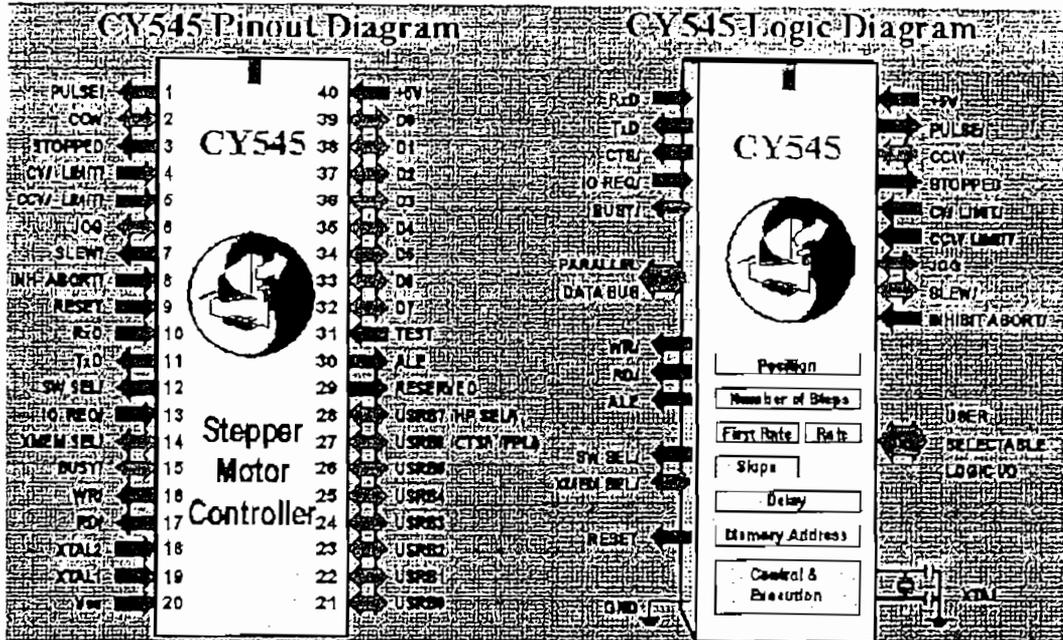
ANEXO 1.2

CÓDIGOS G

ANEXOS

CAPÍTULO II

CY-545 SUMMARY



CY545B Dip Pin#	CY545B/J PLCC Pin#	CY545B/Q QFP* Pin#	Pin Name	I/O	Description
-	1	39	NC		no internal connection
1	2	40	Pulse/	O	Step pulse output
2	3	41	CCW	I/O	Step direction
3	4	42	Stopped	O	Motion status, Lo=stepping
4	5	43	CW Limit/	I	Clockwise step limit reached
5	6	44	CCW Limit/	I	Counter Clockwise step limit reached
6	7	1	Jog	I/O	Manual stepping control
7	8	2	Slew/	I/O	Slew indicator signal
8	9	3	Inhibit_Abort/	I	External motion control
9	10	4	Reset	I	CY545 hardware reset when high
10	11	5	RxD	I	Received Serial Data line
-	12	6	NC		no internal connection
11	13	7	TxD	O	Transmitted Serial Data Line
12	14	8	Sw_Select/	O	Low when reading Switches
13	15	9	IO Request/	I	Parallel handshake input; else Serial baud select: Lo=300, Hi=2400, Z=9600
14	16	10	Xmem Sel/	I/O	External local memory select
15	17	11	Busy/	I/O	Parallel handshake acknowledgement
16	18	12	WR/	O	Parallel Write Strobe
17	19	13	RD/	O	Parallel Read Strobe
18	20	14	Xtal2	I	Crystal oscillator connection
19	21	15	Xtal1	I	Crystal oscillator connection
20	22	16	Vss	I	Ground connection for CY545
-	23	17	NC		no internal connection
21	24	18	UserBit0	I/O	User selectable function, bit 0

22	25	19	UserBit1	I/O	User selectable function, bit 1
23	26	20	UserBit2	I/O	User selectable function, bit 2
24	27	21	UserBit3	I/O	User selectable function, bit 3
25	28	22	UserBit4	I/O	User selectable function, bit 4
26	29	23	UserBit5	I/O	User selectable function, bit 5
27	30	24	UserBit6 or FPL/ or CTS/	I/O	User selectable function, bit 6 or FPL/ or CTS/
28	31	25	UserBit7 or HP_Sel/	I/O	User selectable function, bit 7 or HP_Sel/
29	32	26	Reserved	O	do not connect
30	33	27	ALE	O	Address Latch enable
-	34	28	NC		no internal connection
31	35	27	Test	I	+5v
32	36	30	D 7	I/O	Parallel Data Bus bit 7 (MSB)
33	37	31	D 6	I/O	Parallel Data Bus bit 6
34	38	32	D 5	I/O	Parallel Data Bus bit 5
35	39	33	D 4	I/O	Parallel Data Bus bit 4
36	40	34	D 3	I/O	Parallel Data Bus bit 3
37	41	35	D 2	I/O	Parallel Data Bus bit 2
38	42	36	D 1	I/O	Parallel Data Bus bit 1
39	43	37	D 0	I/O	Parallel Data Bus bit 0 (LSB)
40	44	38	Vcc (+ 5V)	I	+5 volt power supply

Absolute Maximum Ratings

Ambient Temperature under bias 0°C to +70°C
 Storage Temperature -65°C to +150°C
 Voltage on any pin with respect to ground ... -0.5V to Vcc+0.5V
 Power dissipation 0.3 Watts

DC & Operating Characteristics

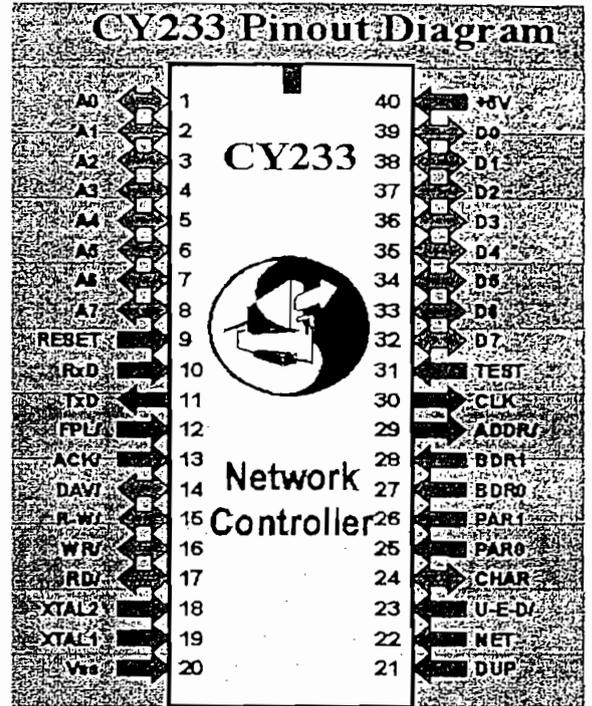
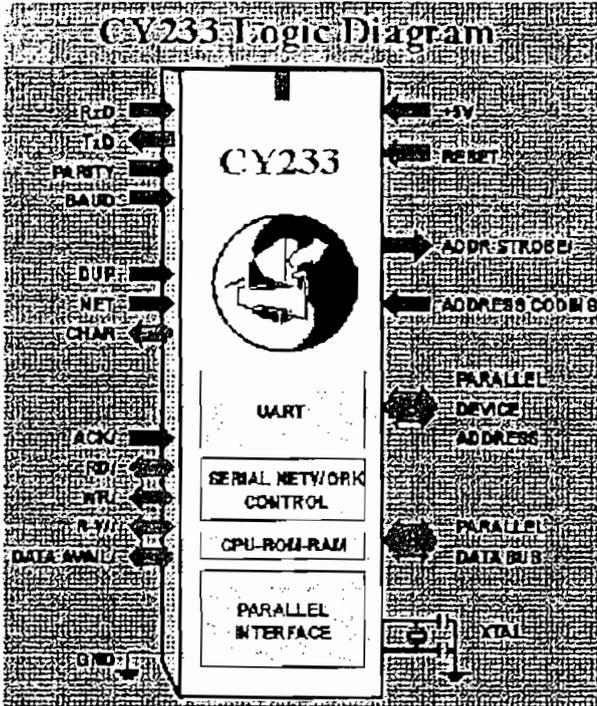
(TA = 0°C to +70°C, Vcc = +5V +/-10%)

SYM	PARAMETER	MIN	MAX	UNIT	REMARKS
I CC	pwr supply current		26	mA	
V IH	input high level	1.9	Vcc	V	(3.5V for XTAL, Reset)
V IL	input low level	-0.5	0.9	V	
I LO	data bus leakage		10	uA	high impedance state
V OH	output high level	2.4		V	I OH = -60 uA
V OL	output low level		0.45	V	I OL = 1.6 mA
F CY	crystal frequency	3.5	16	MHz	see clock circuits

ANEXO 2.2

**ESPECIFICACIONES ELÉCTRICAS DEL
CONTROLADOR INTELIGENTE DE REDES
LOCALES CY-233.**

CY-233 SUMMARY



CY233 Dip Pin#	CY233/J PLCC Pin#	CY233/Q QFP* Pin#	Pin Name	I/O	Description
-	1	39	NC		no internal connection
1	2	40	A0	I/O	Parallel device address bit 0, LSB
2	3	41	A1	I/O	Parallel device address bit 1
3	4	42	A2	I/O	Parallel device address bit 2
4	5	43	A3	I/O	Parallel device address bit 3
5	6	44	A4	I/O	Parallel device address bit 4
6	7	1	A5	I/O	Parallel device address bit 5
7	8	2	A6	I/O	Parallel device address bit 6
8	9	3	A7	I/O	Parallel device address bit 7
9	10	4	Reset	I	CY233 hardware reset when high
10	11	5	RxD	I	Received Serial Data into CY233
-	12	6	NC		no internal connection
11	13	7	TxD	O	Transmitted Serial Data from CY233
12	14	5	FPL/	I	Forced parallel load
13	15	9	ACK/	I	Acknowledge of parallel transfer
14	16	10	DAV/	I/O	Data available to parallel device
15	17	11	R-W/	I/O	Read/Write selection for transfer
16	18	12	WR/	I/O	Parallel Write Strobe
17	19	13	RD/	I/O	Parallel Read Strobe or alternate select
18	20	14	Xtal2	I	Crystal oscillator connection
19	21	15	Xtal1	I	Crystal oscillator connection
20	22	16	Vss	I	Ground connection for CY233

-	23	17	NC		no internal connection
21	24	18	DUP	I	Duplex mode selection
22	25	19	NET	I	Network mode selection
23	26	20	U-E-D/	I	Address coding mode selection
24	27	21	CHAR	I/O	Character mode selection
25	28	22	PAR0	I	Parity/Data length selection, LSB
26	29	23	PAR1	I	Parity/Data length selection, MSB
27	30	24	BDR0	I	Baud rate selection, LSB
28	31	25	BDR1	O	Baud rate selection, MSB
29	32	26	ADDR/	O	Address strobe for parallel device
30	33	27	CLK	O	Clock output, 1/6 crystal rate
-	34	28	NC		no internal connection
31	35	29	Test	I	Internal test signal, connect to Vcc
32	36	30	D 7	I/O	Parallel Data Bus bit 7 (MSB)
33	37	31	D 6	I/O	Parallel Data Bus bit 6
34	38	32	D 5	I/O	Parallel Data Bus bit 5
35	39	33	D 4	I/O	Parallel Data Bus bit 4
36	40	34	D 3	I/O	Parallel Data Bus bit 3
37	41	35	D 2	I/O	Parallel Data Bus bit 2
38	42	36	D 1	I/O	Parallel Data Bus bit 1
39	43	37	D 0	I/O	Parallel Data Bus bit 0 (LSB)
40	44	38	Vcc (+5V)	I	+5 volt power supply

Absolute Maximum Ratings

Ambient Temperature under bias 0°C to +70°C
Storage Temperature -65°C to +150°C
Voltage on any pin with respect to ground ... -0.5V to Vcc+0.5V
Power dissipation 0.2 Watts

DC & Operating Characteristics

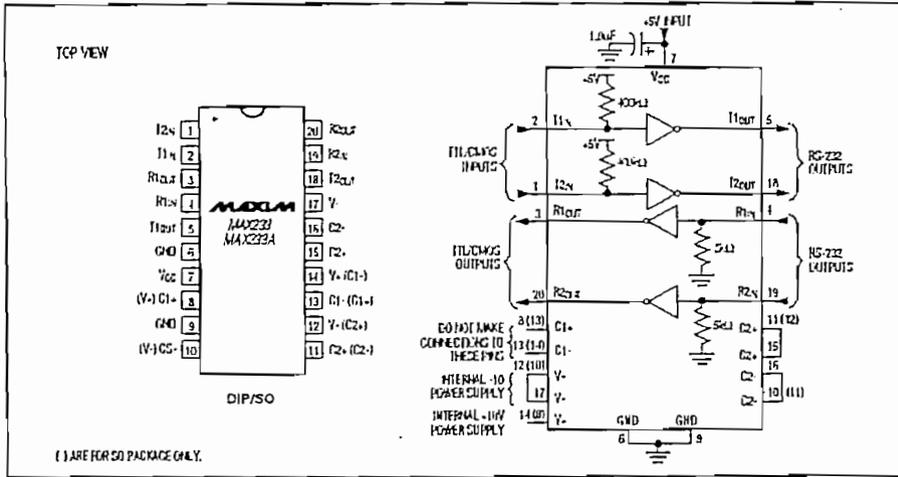
(TA = 0°C to +70°C, Vcc = +5V +/-10%)

SYM	PARAMETER	MIN	MAX	UNIT	REMARKS
I CC	pwr supply current		20	mA	
V IH	input high level	1.9	Vcc	V	(3.5V for XTAL, Reset)
V IL	input low level	-0.5	0.9	V	
I LO	data bus leakage		10	uA	high impedance state
V OH	output high level	2.4		V	I OH = -60 uA
V OL	output low level		0.45	V	I OL = 1.6 mA
F CY	crystal frequency	3.5	12	MHZ	see clock circuits

ANEXO 2.3

ESPECIFICACIONES ELÉCTRICAS DEL MAX233.

MAX233 SUMMARY



ABSOLUTE MAXIMUM RATINGS—MAX220/222/232A/233A/242/243

Supply Voltage (Vcc)-0.3V to +6V
 Input Voltages
 I_{IN}-0.3V to (V_{CC} - 0.3V)
 R_{IN} (Except MAX220)±20V
 R_{IN} (MAX220)±25V
 T_{OUT} (Except MAX220) (Note 1)±15V
 T_{OUT} (MAX220)±13.2V
 Output Voltages
 T_{OUT}±15V
 R_{OUT}-0.3V to (V_{CC} + 0.3V)
 Driver/Receiver Output Short Circuited to GND.....Continuous
 Continuous Power Dissipation (T_A = +70°C)
 16-Pin Plastic DIP (derate 10.53mW/°C above +70°C)....842mW
 18-Pin Plastic DIP (derate 11.11mW/°C above +70°C)....889mW

20-Pin Plastic DIP (derate 8.00mW/°C above +70°C) ..440mW
 16-Pin Narrow SO (derate 8.70mW/°C above +70°C) ...696mW
 18-Pin Wide SO (derate 9.52mW/°C above +70°C).....762mW
 19-Pin Wide SO (derate 9.52mW/°C above +70°C).....762mW
 20-Pin Wide SO (derate 10.00mW/°C above +70°C).....800mW
 20-Pin SSOP (derate 8.00mW/°C above +70°C)640mW
 16-Pin CERDIP (derate 10.00mW/°C above +70°C).....800mW
 18-Pin CERDIP (derate 10.53mW/°C above +70°C).....842mW
 Operating Temperature Ranges
 MAX2_AC __, MAX2_C __0°C to +70°C
 MAX2_AE __, MAX2_E __-40°C to +85°C
 MAX2_AM __, MAX2_M __-55°C to +125°C
 Storage Temperature Range-65°C to +120°C
 Lead Temperature (soldering, 10s)+300°C

Note 1: Input voltage measured with T_{OUT} in high-impedance state, SHDN or V_{CC} = 0V.
Note 2: For the MAX220, V₊ and V₋ can have a maximum magnitude of 7V, but their absolute difference cannot exceed 13V.
 Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243

(V_{CC} = +5V ±10%, C₁-C₄ = 0.1µF, MAX220, C₁ = 0.047µF, C₂-C₄ = 0.33µF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	
RS-232 TRANSMITTERS						
Output Voltage Swing	All transmitter outputs loaded with 3kΩ to GND	±5	±8		V	
Input Logic Threshold Low			1.4	0.8	V	
Input Logic Threshold High	All devices except MAX220 MAX220: V _{CC} = 5.0V	2	1.4		V	
Logic Pull-Up/Input Current	All except MAX220, normal operation SHDN = 0V, MAX222/242, shutdown, MAX220		5	40	µA	
Output Leakage Current	V _{CC} = 5.5V, SHDN = 0V, V _{OUT} = ±15V, MAX222/242 V _{CC} = SHDN = 0V, V _{OUT} = ±15V		±0.01	±10	µA	
Data Rate			200	116	kbps	
Transmitter Output Resistance	V _{CC} = V ₊ = V ₋ = 0V, V _{OUT} = ±2V	360	10M		Ω	
Output Short-Circuit Current	V _{OUT} = 0V	±7	±22		mA	
RS-232 RECEIVERS						
RS-232 Input Voltage Operating Range				±30	V	
RS-232 Input Threshold Low	V _{CC} = 5V	All except MAX243 R2 _{II} MAX243 R2 _{II} (Note 2)	0.8	1.3	-3	V
RS-232 Input Threshold High	V _{CC} = 5V	All except MAX243 R2 _{II} MAX243 R2 _{II} (Note 2)		1.8	-0.5	V
RS-232 Input Hysteresis	All except MAX243, V _{CC} = 5V, no hysteresis in shdn. MAX243		0.2	0.5	1	V
RS-232 Input Resistance			3	5	7	kΩ
TTL/CMOS Output Voltage Low	I _{OUT} = 3.2mA		0.2	0.4		V
TTL/CMOS Output Voltage High	I _{OUT} = -1.6mA		3.5	V _{CC} - 0.2		V
TTL/CMOS Output Short-Circuit Current	Sourcing V _{OUT} = GND Sinking V _{OUT} = V _{CC}	-2	-10	10	30	mA

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243 (continued)

(VCC = +5V ±10%, C1-C4 = 0.1µF, MAX220, C1 = 0.047µF, C2-C4 = 0.33µF, TA = TMIN to TMAX, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
TTL/CMOS Output Leakage Current	SHDN = VCC or EN = VCC (SHDN = 0V for MAX222), 0V ≤ VOUT ≤ VCC			±0.05	±10	µA
EN Input Threshold Low	MAX242			1.4	0.8	V
EN Input Threshold High	MAX242		2.0	1.4		V
Operating Supply Voltage			4.5		5.5	V
VCC Supply Current (SHDN = VCC), Figures 5, 6, 11, 19	No load	MAX220		0.5	2	mA
		MAX222/232A/233A/242/243		4	10	
	3kΩ load both inputs	MAX220		12		
		MAX222/232A/233A/242/243		15		
Shutdown Supply Current	MAX222/242	TA = +25°C		0.1	10	µA
		TA = 0°C to +70°C		2	50	
		TA = -40°C to +85°C		2	50	
		TA = -55°C to +125°C		35	100	
SHDN Input Leakage Current	MAX222/242				±1	µA
SHDN Threshold Low	MAX222/242			1.4	0.8	V
SHDN Threshold High	MAX222/242		2.0	1.4		V
Transition Slew Rate	CL = 50pF to 2500pF, RL = 3kΩ to 7kΩ, VCC = 5V, TA = +25°C, measured from +3V to -3V or -3V to +3V	MAX222/232A/233A/242/243	6	12	30	V/µs
		MAX220	1.5	3	30	
Transmitter Propagation Delay TLL to RS-232 (Normal Operation), Figure 1	tPHLT	MAX222/232A/233A/242/243		1.3	3.5	µs
		MAX220		4	10	
	tPLHT	MAX222/232A/233A/242/243		1.5	3.5	
		MAX220		5	10	
Receiver Propagation Delay RS-232 to TLL (Normal Operation), Figure 2	tPHLR	MAX222/232A/233A/242/243		0.5	1	µs
		MAX220		0.6	3	
	tPLHR	MAX222/232A/233A/242/243		0.6	1	
		MAX220		0.8	3	
Receiver Propagation Delay RS-232 to TLL (Shutdown), Figure 2	tPHLS	MAX242		0.5	10	µs
	tPLHS	MAX242		2.5	10	
Receiver-Output Enable Time, Figure 3	tER	MAX242		125	500	ns
Receiver-Output Disable Time, Figure 3	tDR	MAX242		160	500	ns
Transmitter-Output Enable Time (SHDN Goes High), Figure 4	tET	MAX222/242, 0.1µF caps (includes charge-pump start-up)		250		µs
Transmitter-Output Disable Time (SHDN Goes Low), Figure 4	tDT	MAX222/242, 0.1µF caps		600		ns
Transmitter + to - Propagation Delay Difference (Normal Operation)	tPHLT - tPLHT	MAX222/232A/233A/242/243		300		ns
		MAX220		2000		
Receiver + to - Propagation Delay Difference (Normal Operation)	tPHLR - tPLHR	MAX222/232A/233A/242/243		100		ns
		MAX220		225		

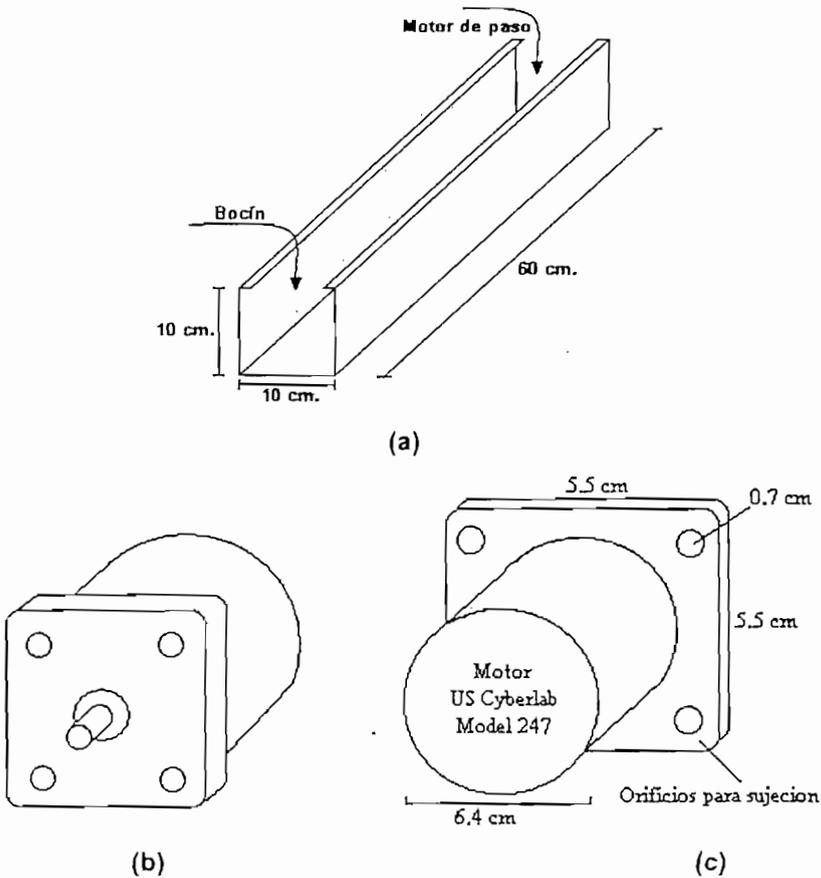
CONSTRUCCIÓN DEL SISTEMA MECÁNICO NEURACTOR CNC-III.

La construcción del NEURACTOR CNC-III, a pesar de no ser complicada, debe realizarse con cuidado siguiendo cada una de las instrucciones que se indican en el folleto del equipo.

Cabe aclarar, que para la realización del presente Proyecto de Titulación, los ejes se encontraban armados, por lo que se ha tomado como base el folleto de instrucciones "U.S. CYBERLAB, INC. Assembly and Operation Manual" para explicar en forma resumida la construcción del sistema mecánico.

Los ejes del sistema mecánico NEURACTOR CNC-III están montados sobre un canal C (Fig. 1a) de 60 cm de largo y 10 cm de ancho. EL canal C constituye el soporte principal para las partes del sistema, sobre el cual se montará el motor a pasos y en su extremo se montará el bocín, el mismo que sostiene la varilla principal del sistema que tiene forma de un extenso tornillo; además, cuenta con un soporte final que permite sostener varias varillas lisas que tienen la función de proporcionar el alineamiento y soporte en el sistema.

Los motores a paso que se utilizan en éste trabajo son los US-CYBERLAB modelo 247 (Fig. 1b). Para montar éstos motores a paso en el canal, se diseñó un soporte especial el mismo que se puede observar en la figura 1c y 1d.



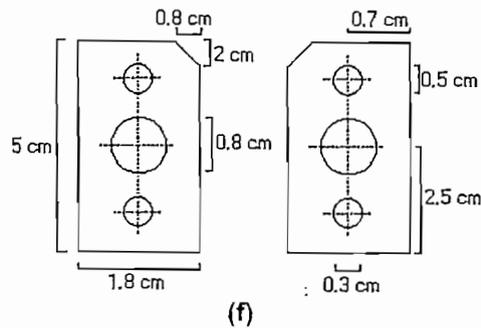
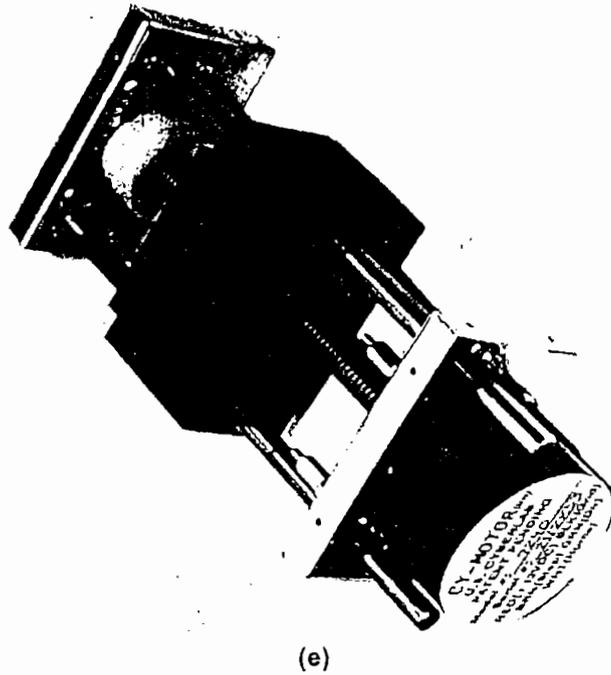
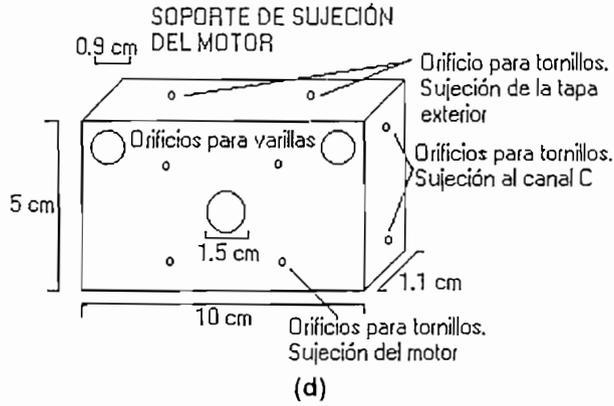


FIG. 1 (a) Canal C; (b) Motor a paso US-CYBERLAB; (c) Características físicas del motor; (d) Soporte para montar el motor sobre el canal; (e) Motor montados sobre el soporte; (f) Placas lisas del soporte del motor.¹

¹ CORDOVA FRANCISCO, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos", pág. 56-59

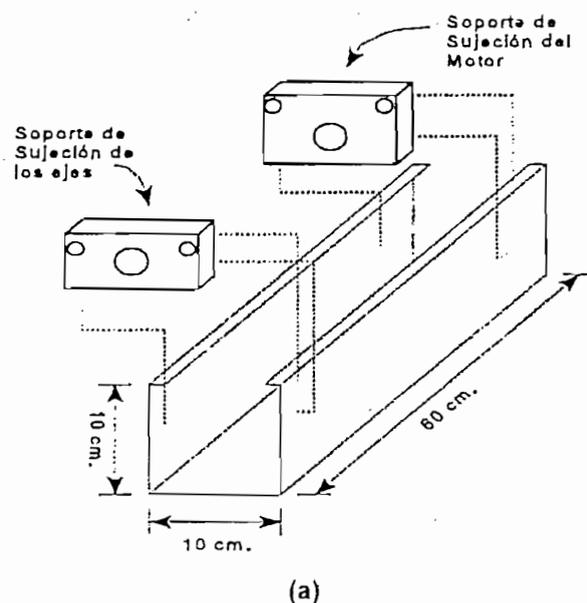
U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 7

Es necesario indicar que el soporte del motor dispone de dos agujeros sobre los cuales se montan dos placas lisas (Fig. 1e) que disponen de agujeros por los cuales pasan las varillas lisas y se las aseguran mediante tornillos, tuercas y arandelas.

Las funciones principales de éstos agujeros son las de servir de soporte para las varillas lisas antes mencionadas que pasan a través de los mismos; de igual forma sirven para la alineación y calibración de las posiciones de las varillas debido a que los agujeros para los tornillos que las sujetan al soporte del motor poseen un mayor diámetro que los tornillos; lográndose de éste modo que el sistema se deslice libremente.

Posteriormente se coloca el motor y el soporte sobre el canal C (Fig. 2A), los agujeros para el montaje del soporte en el canal tienen un diámetro mayor que los tornillos para el ajuste en el proceso de alineación. Luego de montar el motor y su soporte, se coloca una lámina en forma de C de 43 cm de largo y 10 cm de ancho en el interior del canal C, que sirve de base para que el bloque deslizante se mueva a través del canal, siendo de éste modo un soporte adicional para el bloque (Figura 2b).

En el eje principal, se coloca el bloque deslizante (de color verde) que está construido de un material de alto peso molecular que permite una auto lubricación; esta característica ayuda a que se deslice suavemente el bloque a través de las dos varillas lisas (Fig. 2c). Dentro del bloque deslizante se coloca una tuerca especial y de diseño único; el bloque tiene dos agujeros adicionales a través de los cuales pasarán dos varillas lisas (Fig. 2d). A través de la tuerca se enrosca el sinfín, insertando de éste modo el bloque deslizante en el sistema.



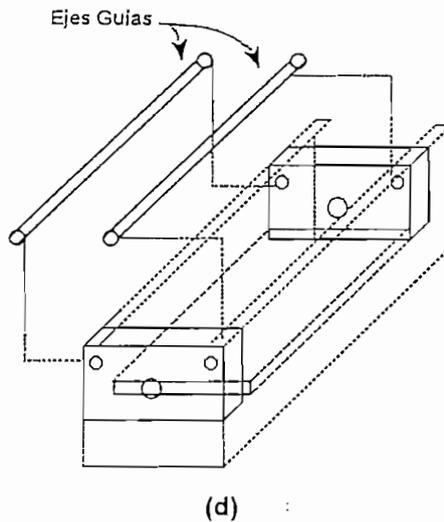
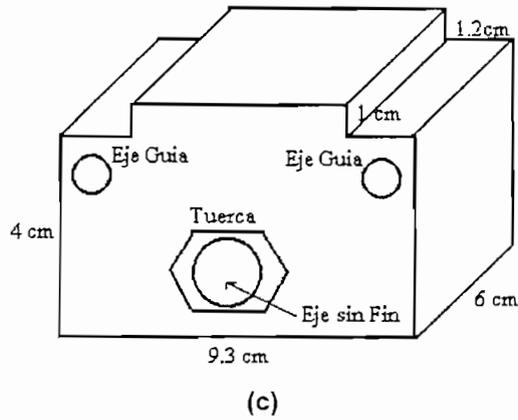
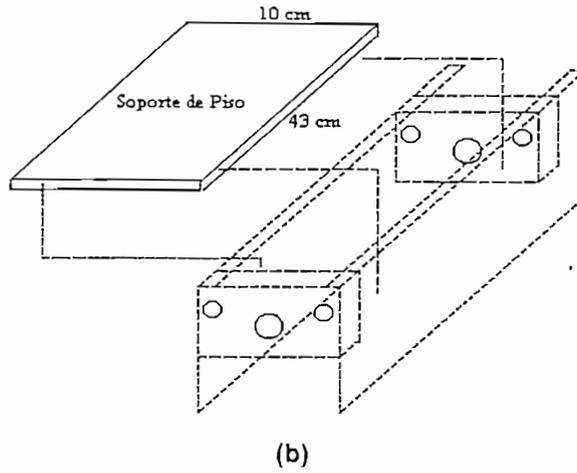


FIG. 2 (a) Soportes de sujeción de los ejes y motor; (b) Lámina interna del canal C; (c) Bloque deslizable; (d) Bloque completo con sus varillas.²

²CORDOVA FRANCISCO, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos", pág. 60-64

Se inserta el bloque algunos centímetros en las varillas, las mismas que se deslizan en los agujeros provistos en el bloque, y en el soporte del motor. Adicionalmente, en los extremos del canal se montan tapas, con el objetivo de cubrir el motor y el final del canal. La tapa del motor dispone de orificios para colocar un conector tipo DIN que conectan los cables correspondientes al motor (Fig. 3). De ésta manera se completa la construcción de los sistemas móviles correspondientes a los ejes X y Y (el eje Y dispone de dos sistemas similares).

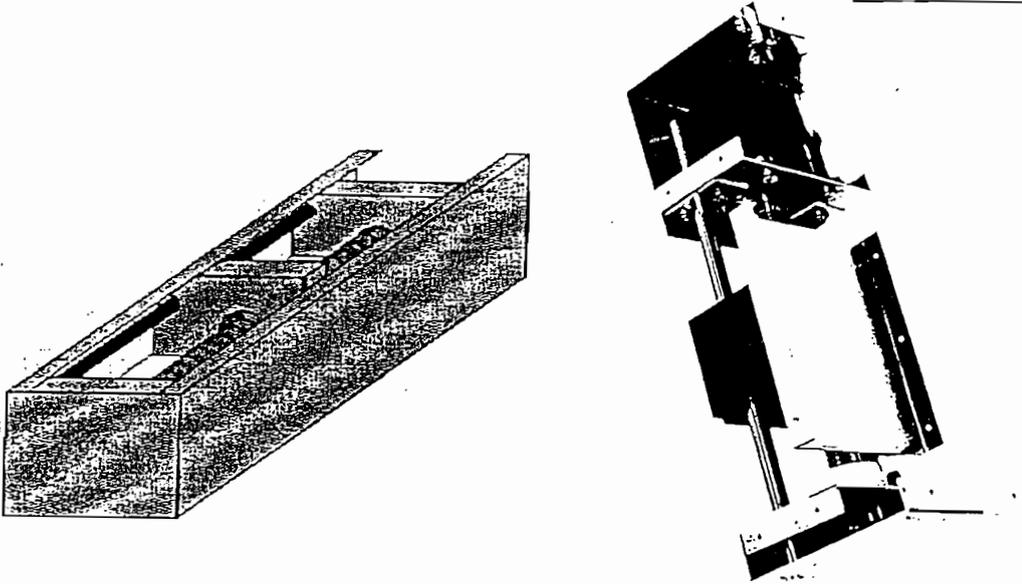


FIG. 3 Sistema completo fuera del canal C.³

Después de la construcción del dispositivo electromecánico NEURACTOR CNC-III, se realiza la calibración del sistema en todos los ejes con el objetivo de lograr la mayor precisión en su trabajo y obtener un buen nivel de perpendicularidad entre los ejes; con lo cuál se dispone a montar los sistemas sobre una base de madera, diseñada para mantener fijos a los ejes. Después de alinear las varillas guías, se procede a alinear las placas del soporte del bloque deslizante, ajustando su altura y distancia de cada lado de tal forma que el bloque toque exactamente la superficie de la placa (tornillos de soporte).

En resumen, se montan los ejes Y sobre la plancha de madera mediante el uso de tornillos manteniendo paralelismo entre éstos ejes. Con los ejes Y como base de referencia se fija el eje X, asegurándolo en uno de los ejes con los tornillos de soporte. Luego, con el uso del programa se mueven los motores para asegurar que los dos ejes queden perpendiculares. Finalmente se monta el eje Z sobre los otros ejes manteniendo su cuadratura para lograr así la mayor exactitud posible.

Con respecto al diseño y construcción del sistema mecánico en el eje Z, se puede indicar que las características y procedimientos para su montaje son similares a

³ CORDOVA FRANCISCO, "Taladro con Control Numérico para la Fabricación de Circuitos Impresos, pág. 65

U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 8

los utilizados en los ejes X y Y, cuyas diferencias con respecto a los otros ejes es que el canal C tiene una longitud de 28 cm. Después del proceso de montaje del sistema mecánico en el eje Z, el sistema adquiere la siguiente forma (Fig. 4):

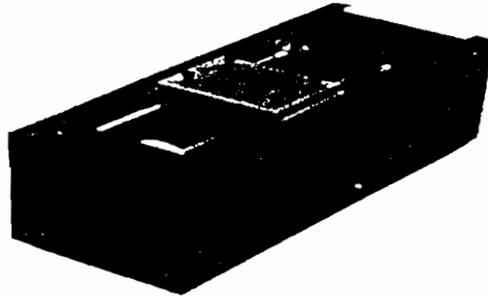


FIG. 4 Montaje del sistema correspondiente al eje Z.⁴

Debido a que en el presente Proyecto de Titulación, se trabajará únicamente con los ejes X y Y, la herramienta de corte (cortador de espumaflex), se sujetará sobre el bloque deslizante del eje X para el control de su movimiento.

Al finalizar el proceso de construcción del sistema mecánico NEURACTOR CNC-III, el producto final adquiere la siguiente forma (Fig. 5):

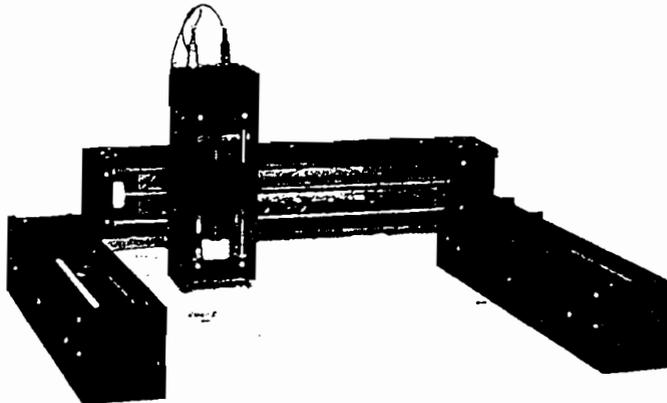


FIG. 5 NEURACTOR CNC-III.⁵

CALIBRACIÓN DE LOS SISTEMAS.

Usando el programa base, se mueve el bloque deslizante de cada uno de los ejes hacia el final del canal C. Por medio del uso de una regla se mide la altura desde la superficie sobre la que está colocado el canal, hasta la parte superior del bloque deslizante. Luego se lleva el bloque deslizante hacia el inicio del canal C, cerca al motor de paso, verificando en todo momento que la altura antes tomada no varíe (usando como referencia el centro del bloque). Teniendo los valores de altura constante y al calibrar los tornillos de orientación del motor y los tornillos de montaje del motor, se asegura que el movimiento horizontal del bloque deslizante sea constante.

⁴ U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 16

⁵ U.S. CYBERLAB, INC., "Assembly and Operation Manual", pág. 3

Luego, con el bloque deslizante fijo, se procede a alinear a las varillas guías, ajustando primero el final de una de ellas. Mediante el programa se lleva el bloque deslizante a lo largo del canal, para observar el lado opuesto de la varilla guía, el mismo que sube o baja mientras el bloque se mueve. Luego se ajusta uno de los finales de la segunda varilla, y se repite el procedimiento anterior, para conseguir una alineación definitiva de las dos varillas.

Por último, se alinea la placa de soporte del bloque deslizante, ajustando la altura de ésta placa de final a final, y de lado a lado, de tal forma que el bloque toque exactamente a la superficie de la placa. Este procedimiento se repite con cada uno de los restantes ejes.

ENSAMBLAJE FINAL.

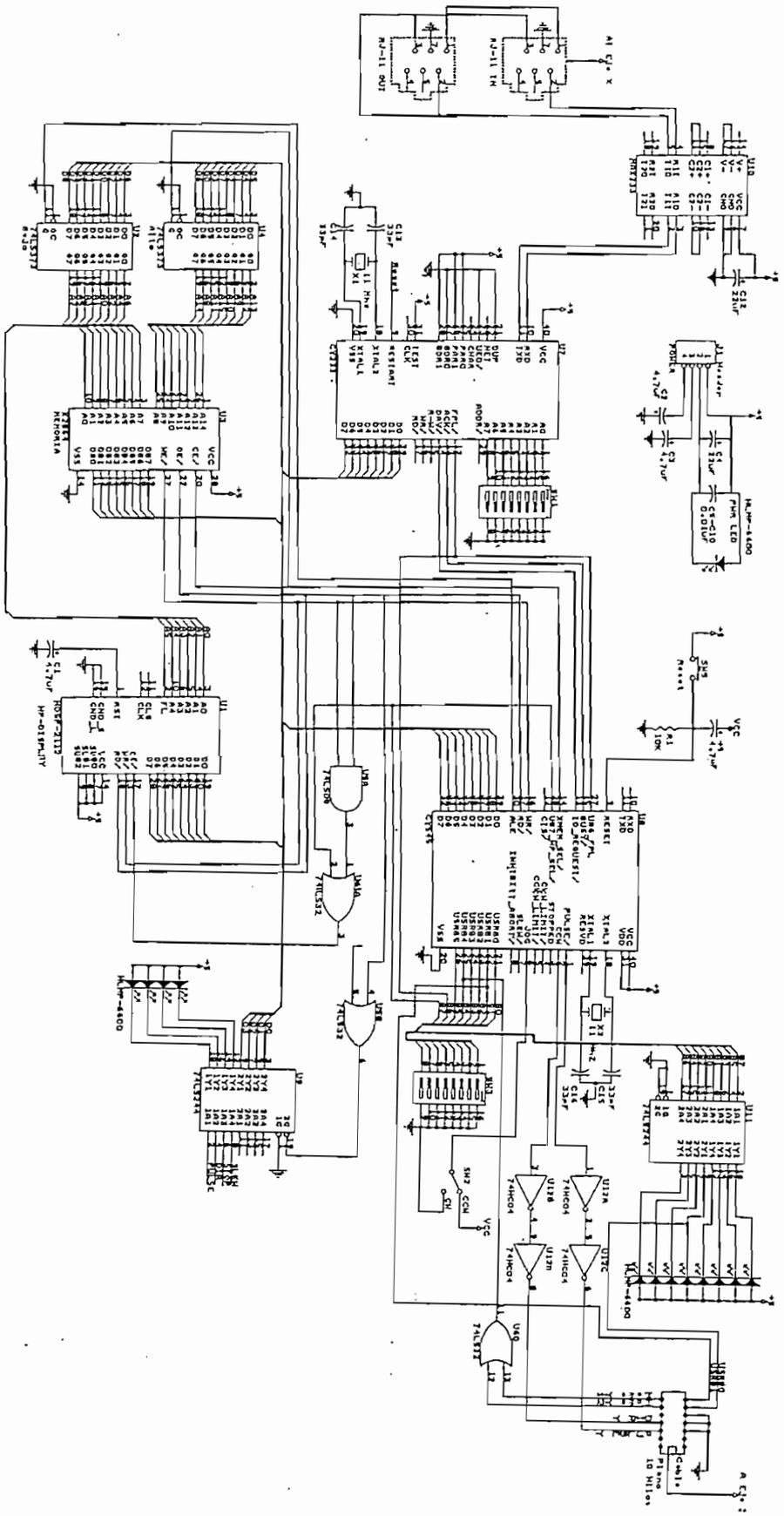
Para el ensamblaje final del sistema mecánico NEURACTOR CNC-III, se monta el eje Y sobre la base de madera; asegurados los dos finales del eje, se establece una base de referencia. Mediante una regla en T se trazan dos líneas perpendiculares a este eje (desde el inicio y fin del eje), posteriormente se coloca el segundo eje Y en paralelo con el primer eje (entre las líneas trazadas).

Se recomienda comprar una mesa de granito para montar el equipo, y lograr así la disminución de errores que se producen por los cambios de temperatura y humedad; pero como el costo de una mesa de granito es muy elevado, se ha utilizado una plancha de madera que se comporta aceptablemente con los cambios generados. Después se fija el eje X sobre los ejes deslizantes de estos dos ejes; usando una regla T se asegura que todo el sistema esté en escuadra alrededor de la base. Luego, mediante un software implementado en Visual Basic, se mueven los motores asegurando que los dos ejes hayan quedado perpendiculares.

Mediante un programa básico se pueden verificar movimientos, ajustes, distancias, y con la finalización del equipo se pueden realizar varias aplicaciones como la producción de partes exactas, perforación de agujeros con variación de profundidades, aplicación de spray o pegamento con movimientos en tres dimensiones, cortado de materiales con diferentes tipos de herramientas, corte de artículos de cuero o tejidos siguiendo un patrón, estampado de hojas de metal, automatización del movimiento de una fotografía en un stand, control de movimiento para fotografía, etc., además éstas aplicaciones se las puede realizar con materiales como madera, plástico, corcho, metales ligeros, etc.

ANEXO 2.5

CIRCUITO.



ANEXOS

CAPÍTULO III

ANEXO 3.1

DESARROLLO DEL SOFTWARE.

MODULE1. BAS

```
Public Sub Enviar(ByRef Forma1 As Form) ' RUTINA BASICA DE ENVIO
Dim Ch As String ' DE DATOS VIA SERIAL
Dim J As Integer
  J = 1
  For J = 1 To Len(Dato)
    Ch = Mid(Dato, J, 1)
    Forma1.MSComm1.Output = Ch
    Call Retardo(0.002)
  Next J
End Sub
Public Sub Inicio1(ByRef Forma1 As Form, N As Integer)
  ' INICIALIZA MICROS
  Dato = "W" + "01" + "I" + Chr$(13) ' BUSCA HOME
  Call Enviar(Forma1) ' RETIRA 50 PASOS DEL HOME
  Call Retardo(0.3)
  Dato = "W" + "01" + "B 15H" + Chr$(13)
  Call Enviar(Forma1)
  Dato = "W" + "02" + "I" + Chr$(13)
  Call Enviar(Forma1)
  Call Retardo(0.3)
  Call HomeX(Forma1)
  Call HomeY(Forma1)
  Call Retardo(0.3)
  Do While True
    Dato = "W" + "01" + "O 085H" + Chr$(13) ' RESPUESTA AL DISPLAY
    Call Enviar(Forma1)
    Dato = "W" + "01" + "? B" + Chr$(13)
    Call Enviar(Forma1)
    Dato = "W" + "01" + Chr$(13)
    Call Enviar(Forma1)
    Call Enviar(Forma1)
    Dato = "W" + "01" + "O 081H" + Chr$(13) ' RESPUESTA AL COMPUTADOR
    Call Enviar(Forma1)
    Dato = "W01? B" + Chr$(13)
    Call Enviar(Forma1)
    Call Retardo(1)
    Ch1 = ""
    Do While Forma1.MSComm1.InBufferCount > 0
      Ch1 = Forma1.MSComm1.Input
      Call Retardo(0.4)
    Loop
    If Len(Ch1) > 0 Then
      Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
      Ch1 = Left(Ch1, Len(Ch1) - 1)
    End If
    If Ch1 = "65499" Or Ch1 = "65503" Then ' HOME X
      Exit Do
    End If
  Loop
  Do While True
    Dato = "W" + "02" + "O 081H" + Chr$(13)
    Call Enviar(Forma1)
    Dato = "W02? B" + Chr$(13)
    Call Enviar(Forma1)
    Call Retardo(1)
    Ch1 = ""
  Do While Forma1.MSComm1.InBufferCount > 0
    Ch1 = Forma1.MSComm1.Input
    Call Retardo(0.5)
  Loop
  If Len(Ch1) > 0 Then
    Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
    Ch1 = Left(Ch1, Len(Ch1) - 1)
  End If
  If Ch1 = "65535" Or Ch1 = "65519" Then ' HOME Y
    Exit Do
  End If
  Loop
  If N = 1 Then ' SI VIENE DEL MANUAL
    Call Inicializa_Y(Forma1)
  End If
  If N = 1 Then ' SI VIENE DEL MANUAL
```

```

    Call Inicializa_X(Forma1)
End If
Dato = "W" + "01" + "N 50" + Chr$(13) ' AVANZA 50 PASOS DEL HOME
Call Enviar(Forma1)
Dato = "W" + "01" + "F 12" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "01" + "S 4" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "01" + "R 16" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "01" + "-" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "01" + "G" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "02" + "N 50" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "02" + "F 12" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "02" + "S 4" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "02" + "R 16" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "02" + "+" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "02" + "G" + Chr$(13)
Call Enviar(Forma1)
'Dato = "W" + "02" + "A 00" + Chr$(13)
'Call Enviar(Forma1)
End Sub
Public Sub Retardo(N As Single) ' RETARDO EN SEGUNDOS Y
Dim Tiempo1, Tiempo2 As Single ' FRACCIONES
    Tiempo1 = Timer
    Do
        Tiempo2 = Timer
    Loop Until Tiempo2 >= Tiempo1 + N
End Sub
Public Sub HomeX(ByRef Forma1 As Form) ' HOME X
    Dato = Chr$(13) + Chr$(13)
    Call Enviar(Forma1)
    Dato = "W" + "01" + "F 100" + Chr$(13)
    Call Enviar(Forma1)
    Dato = "W" + "01" + "H 12H" + Chr$(13) ' BIT 2
    Call Enviar(Forma1)
End Sub
Public Sub HomeY(ByRef Forma1 As Form) ' HOME Y
    Dato = "W" + "02" + "F 100" + Chr$(13)
    Call Enviar(Forma1)
    Dato = "W" + "02" + "H 04H" + Chr$(13) ' BIT 4
    Call Enviar(Forma1)
End Sub
Public Sub Inicializa_X(ByRef Forma1 As Form) ' INICIALIZA X
    MousePointer = 11
    Call ActualizacionX(Forma1)
    Dato = "W01" + "B 15H" + Chr$(13)
    Call Enviar(Forma1)
    MousePointer = 0
End Sub
Public Sub Inicializa_Y(ByRef Forma1 As Form) ' INICIALIZA Y
    MousePointer = 11
    Call ActualizacionY(Forma1)
    MousePointer = 0
End Sub
Public Sub ActualizacionX(ByRef Forma1 As Form) ' ACTUALIZA PARAMETROS DE X
Dim Ch As String
Dim K As Integer
Dim antiguo As Integer
Dim Ch1 As String
    For J = 0 To 6
        Forma1.posicion(J).Caption = ""
    Next J
    MousePointer = 11
    Forma1.MSComm1.Output = "W" + "01" + Chr$(13)
    Forma1.MSComm1.Output = "W" + "01" + Chr$(13)
    Dato = "W" + "01" + "O 085H" + Chr$(13)

```

```

Call Enviar(Forma1)
' ENVIO DE DATOS PARA LEER POSICION
Ch1 = ""
For J = 0 To 6
Dato = "W" + "01" + "O 085H" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "01" + "? " + Parametro(J) + Chr$(13)
Call Enviar(Forma1)
Dato = Chr$(13)
Call Enviar(Forma1)
Dato = Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "01" + "O 081H" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "01" + "? " + Parametro(J) + Chr$(13)
Call Enviar(Forma1)
Call Retardo(0.2)
Ch = ""
Do While ManualFor.MSComm1.InBufferCount > 0
Ch = Ch + ManualFor.MSComm1.Input
Call Retardo(0.2)
Loop
ManualFor.posicion(J).Caption = Right(Ch, Len(Ch) - InStr(Ch, "="))
ManualFor.posicion(J).Refresh
Next J
MousePointer = 0
End Sub
Public Sub ActualizacionY(ByRef Forma1 As Form) ' ACTUALIZA PARAMETROS DE Y
Dim Ch As String
Dim K As Integer
Dim antiguo As Integer
Dim Ch1 As String
For J = 7 To 13
Forma1.posicion(J).Caption = ""
Next J
MousePointer = 11
Forma1.MSComm1.Output = Chr$(13)
Forma1.MSComm1.Output = Chr$(13)
'Dato = "W" + "02" + "I" + Chr$(13) 'SE QUITO PARA Q ESTE IGUAL AL EJE X
'Call Enviar(Forma1)
Dato = "W" + "02" + "O 085H" + Chr$(13)
Call Enviar(Forma1)
' ENVIO DE DATOS PARA LEER POSICION
Ch1 = ""
For J = 0 To 6
Dato = "W" + "02" + "O 085H" + Chr$(13)
Call Enviar(Forma1)
Retardo (0.2)
Dato = "W" + "02" + "? " + Parametro(J) + Chr$(13)
Call Enviar(Forma1)
Retardo (0.2)
Dato = Chr$(13)
Call Enviar(Forma1)
Dato = Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "02" + "O 081H" + Chr$(13)
Call Enviar(Forma1)
Dato = "W" + "02" + "? " + Parametro(J) + Chr$(13)
Call Enviar(Forma1)
Call Retardo(0.2)
Ch = ""
Do While Forma1.MSComm1.InBufferCount > 0
Ch = Ch + Forma1.MSComm1.Input
Call Retardo(0.2)
Loop
Forma1.posicion(J + 7).Caption = Right(Ch, Len(Ch) - InStr(Ch, "="))
Forma1.posicion(J + 7).Refresh
Next J
MousePointer = 0
End Sub
Public Sub Ordenar_Lineas() ' ORDENAR LA SECUENCIA DE
Dim I, J, K As Integer ' LAS LINEAS
Dim Salva_Y, Salva_X As Integer
Dim Pl_X, Pl_Y, PT_X, PT_Y As Long
Dim PS_X, PS_Y As Long

```

```

Dim I_C As Long
Dim Arreglo_Usos() As Byte
Dim Distancia As Double
' PRIMERO BUSCAR EL PUNTO MAS CERCANO AL HOME
I = 30000
For J = 1 To Indice_Lineas
    If Matriz_Lineas(J, 2) < I Then
        I = Matriz_Lineas(J, 2)
        Salva_Y = Matriz_Lineas(J, 2)
    End If
Next J
I = 32000
For J = 1 To Indice_Lineas
    If Matriz_Lineas(J, 2) = Salva_Y Then
        If Matriz_Lineas(J, 1) < I Then
            I = Matriz_Lineas(J, 1)
            Salva_X = J
        End If
    End If
Next J
ReDim Arreglo_Usos(1 To Indice_Lineas)
' SALVA_X ME DA EL INDICE DEL MAS CERCANO A HOME
I_C = 1
PI_X = Matriz_Lineas(Salva_X, 1)
PI_Y = Matriz_Lineas(Salva_X, 2)
K = Salva_X
Do While True
    Arreglo_Usos(K) = 1
    Matriz_Copia(I_C, 1) = Matriz_Lineas(K, 1)
    Matriz_Copia(I_C, 2) = Matriz_Lineas(K, 2)
    Matriz_Copia(I_C, 3) = Matriz_Lineas(K, 3)
    Matriz_Copia(I_C, 4) = Matriz_Lineas(K, 4)
    I_C = I_C + 1
    If I_C > Indice_Lineas Then
        Indice_Lineas = Indice_Lineas + 1
        Exit Do
    End If
    PS_X = Matriz_Lineas(K, 3) ' BUSCAR EL INICIAL DE LA SIGUIENTE RECTA
    PS_Y = Matriz_Lineas(K, 4) ' QUE ES EL FINAL DE LA YA PROCESADA
    If PS_X = PI_X And PS_Y = PI_Y Then ' SI YA REGRESO AL INICIAL HAY Q SALIR
        'Matriz_Copia(I_C, 1) = Matriz_Lineas(K, 3)
        'Matriz_Copia(I_C, 2) = Matriz_Lineas(K, 2)
        'Matriz_Copia(I_C, 3) = PS_X
        'Matriz_Copia(I_C, 4) = PS_Y
        Exit Do
    End If
    'K = I 'revisar porque k=Matriz_Lineas(J, 1)
    Cerrado = False
    Diferencia_X = 30000
    Diferencia_Y = 20000
    Distancia = 400000
    For I = 1 To Indice_Lineas
        If Arreglo_Usos(I) = 0 Then
            If Sqr(((CInt(PS_X) - Matriz_Lineas(I, 1)) * (CInt(PS_X) - Matriz_Lineas(I, 1))) + ((CInt(PS_Y) - Matriz_Lineas(I, 2)) *
(CInt(PS_Y) - Matriz_Lineas(I, 2)))) < Distancia Then
                Distancia = Sqr(((CInt(PS_X) - Matriz_Lineas(I, 1)) * (CInt(PS_X) - Matriz_Lineas(I, 1))) + ((CInt(PS_Y) -
Matriz_Lineas(I, 2)) * (CInt(PS_Y) - Matriz_Lineas(I, 2))))
                Cerrado = True
                K = I
            End If
        End If
    Next I
    ' PUEDE ESTAR INVERTIDA LA LINEA
    For I = 1 To Indice_Lineas
        If Arreglo_Usos(I) = 0 Then
            If Sqr(((CInt(PS_X) - Matriz_Lineas(I, 3)) * (CInt(PS_X) - Matriz_Lineas(I, 3))) + ((CInt(PS_Y) - Matriz_Lineas(I, 4)) *
(CInt(PS_Y) - Matriz_Lineas(I, 4)))) < Distancia Then
                Distancia = Sqr(((CInt(PS_X) - Matriz_Lineas(I, 3)) * (CInt(PS_X) - Matriz_Lineas(I, 3))) + ((CInt(PS_Y) -
Matriz_Lineas(I, 4)) * (CInt(PS_Y) - Matriz_Lineas(I, 4))))
                Cerrado = True
            End If
        End If
    Next I
    ' INVERTIR LA LINEA PARA QUE PUEDA CONTINUAR
    PT_X = Matriz_Lineas(I, 3)
    PT_Y = Matriz_Lineas(I, 4)
    Matriz_Lineas(I, 3) = Matriz_Lineas(I, 1)

```

```

Matriz_Lineas(l, 4) = Matriz_Lineas(l, 2)
Matriz_Lineas(l, 1) = PT_X
Matriz_Lineas(l, 2) = PT_Y
K = l
End If
End If
Next l
If Not Cerrado Then
MsgBox "El poligono no está completamente cerrado. Rectifique..", vbExclamation, "Advertencia!"
Exit Sub
End If
Loop
End Sub

```

DECLARACIONES.BAS

```

Global Dato As String ' CADENA DE DATOS A TRANSMITIR
Global Parametro(0 To 6) As String ' LETRAS DE LOS PARAMETROS A CONSULTAR
Global PosX(50000) As Double ' COORDENADA EN X
Global PosY(50000) As Double ' COORDENADA EN Y
Global Matriz_Lineas(1 To 500000, 1 To 6) As Double 'X1,Y1,X2,Y2,Velcd X,Velcd Y
Global Matriz_Copia(1 To 500000, 1 To 4) As Double ' Temporal para ordenamiento
Global Indice_Lineas As Integer ' MANTIENE EL INDICE DE LAS LINEAS
Global NumPuntos As Integer ' Auxiliar número de coordenadas
Global Nombre As String ' Nombre del archivo a procesar
Global Tipo As String ' Tipo de archivo
Global Cerrado As Boolean ' Polígono cerrado?
Global Tiempo1, Tiempo2 As Single

```

```

Public Sub Carga_Parametros() ' PARAMETROS A CONSULTAR
Parametro(0) = "P"
Parametro(1) = "R"
Parametro(2) = "S"
Parametro(3) = "F"
Parametro(4) = "N"
Parametro(5) = "Y"
Parametro(6) = "B"
End Sub

```

INICIAL.FRM

```

Private Sub Automatico_Click()
Form2.Hide
AUTO.Show
End Sub
Private Sub Archivos_Click()
Form2.Hide
Form1.Show
End Sub
Private Sub Fin_Click()
Unload Me
End
End Sub
Private Sub Form_KeyPress(KeyAscii As Integer)
If UCase(Chr(KeyAscii)) = "M" Then Call Manual_Click
If UCase(Chr(KeyAscii)) = "A" Then Call Automatico_Click
If UCase(Chr(KeyAscii)) = "C" Then Call Archivos_Click
If UCase(Chr(KeyAscii)) = "S" Then Unload Me
End Sub
Private Sub Form_Load()
Me.Top = (Screen.Height - Me.Height) / 2
Me.Left = (Screen.Width - Me.Width) / 2
Call Carga_Parametros
End Sub
Private Sub Manual_Click()
Form2.Hide
ManualFor.Show
End Sub

```

MANUALF.FRM

```

Dim PosicionX As Long ' POSICION RELATIVA DE X CON RESPECTO A HOME X
Dim PosicionY As Long ' POSICION RELATIVA DE Y CON RESPECTO A HOME Y
Dim SignoX As Byte ' SIGNO ACTUAL DE EJE X
Dim SignoY As Byte ' SIGNO ACTUAL DE EJE Y

```

```

Dim Cortador_Encendido As Boolean      ' Está prendido el cortador ?
Const Numeros = "0123456789"

Private Sub ActualizarX_Click() ' ACTUALIZAR PARAMETROS DE EJE X
    Call ActualizacionX(Me)
End Sub
Private Sub Borrado() ' BORRADO DE LOS DISPLAY
    Dato = "W" + "01" + "O 081H" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "02" + "O 081H" + Chr$(13)
    Call Enviar(Me)
    Retardo (0.1)
End Sub
Private Sub ActualizarY_Click() ' ACTUALIZAR PARAMETROS DE EJE Y
    Call ActualizacionY(Me)
End Sub
Private Sub BorrarDisplay_Click() ' BORRAR LOS 2 DISPLAYS
    MousePointer = 11
    Dato = "W" + "01" + "O 081H" + Chr$(13)
    Call Enviar(Me)
    Retardo (0.2)
    Dato = "W" + "02" + "O 081H" + Chr$(13)
    Call Enviar(Me)
    Retardo (0.2)
    TextoDisplay.Text = ""
    MousePointer = 0
End Sub
Private Sub DatosPosicion_KeyPress(KeyAscii As Integer)
    If KeyAscii > 24 Then
        If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
    End If
End Sub
Private Sub DatosPosicionY_KeyPress(KeyAscii As Integer)
    If KeyAscii > 24 Then
        If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
    End If
End Sub
Private Sub Enciende_Cortador()
    Dato = "W" + "01" + "O 081H" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "01" + "B 13H" + Chr$(13)
    Call Enviar(Me)
    Shape2.FillColor = &HFF&      ' Rojo
End Sub
Private Sub Apaga_Cortador()
    Dato = "W" + "01" + "O 081H" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "01" + "B 03H" + Chr$(13)
    Call Enviar(Me)
    Shape2.FillColor = &H80000008      ' Negro
End Sub
Private Sub Enciende_Click()      ' Enciende o apaga el cortador
    If Not Cortador_Encendido Then
        Enciende_Cortador
        Cortador_Encendido = True
    Else
        Apaga_Cortador
        Cortador_Encendido = False
    End If
End Sub
Private Sub FirstRateXVal_KeyPress(KeyAscii As Integer)
    If KeyAscii > 24 Then
        If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
    End If
End Sub
Private Sub FirstRateYVal_KeyPress(KeyAscii As Integer)
    If KeyAscii > 24 Then
        If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
    End If
End Sub
Private Sub Form_Activate() ' SE EJECUTA AL ACTIVARSE LA FORMA
    Inicializa_Equipos.Enabled = False
    If 1 = 2 Then
        Timer1.Enabled = True
    End If
End Sub

```

```

Ch1 = ""
Do While True
  DoEvents
  Dato = "W" + "01" + "? B" + Chr$(13)
  Call Enviar(Me)
  Do While MSComm1.InBufferCount > 0
    Ch1 = MSComm1.Input
    Call Retardo(0.2)
  Loop
  If Len(Ch1) > 0 Then
    If InStr(Ch1, "=") > 0 Then
      Timer1.Enabled = False
      Exit Do
    End If
  End If
Loop
End If
Inicializa_Equipo.Enabled = True
Inicializa_Equipo.SetFocus
Apaga_Cortador
Cortador_Encendido = False
End Sub

```

```

Private Sub HomeX1_Click() ' BUSCAR HOME DEL EJE X
' SUSPENDE OTROS CONTROLES
Controles.Enabled = False
ControlesY.Enabled = False
ControlesD.Enabled = False
Call HomeX(Me)
' CONSULTA B PARA ASEGURAR LLEGADA A HOME
Do While True
  Dato = "W" + "01" + "B 15H" + Chr$(13)
  Call Enviar(Me)
  Dato = "W" + "01" + "O 085H" + Chr$(13)
  Call Enviar(Me)
  Dato = "W" + "01" + "? B" + Chr$(13)
  Call Enviar(Me)
  Dato = Chr$(13)
  Call Enviar(Me)
  Dato = Chr$(13)
  Call Enviar(Me)
  Dato = "W" + "01" + "O 081H" + Chr$(13)
  Call Enviar(Me)
  Dato = "W01? B" + Chr$(13)
  Call Enviar(Me)
  Call Retardo(0.2)
  Ch1 = ""
  Do While MSComm1.InBufferCount > 0
    Ch1 = MSComm1.Input
    Call Retardo(0.2)
  Loop
  If Len(Ch1) > 0 Then
    Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
    Ch1 = Left(Ch1, Len(Ch1) - 1)
  End If
  If Ch1 = "65499" Then ' X EN HOME
    Exit Do
  End If
Loop
' RETIRA 50 PASOS DEL HOME (DECLARA POSICION 0)
Dato = "W" + "01" + "N 50" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "F 12" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "S 4" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "R 16" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "-" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "G" + Chr$(13)
Call Enviar(Me)
' HABILITA CONTROLES
Controles.Enabled = True

```

```

ControlesY.Enabled = True
ControlesD.Enabled = True
DatosPosicion.Text = 0
PASOS_X_CM.Text = 0
posicion(0).Caption = 0
PosicionX = 0
' FUERA DE HOME
SignoX = 1
End Sub

```

```

Private Sub HomeY1_Click() ' BUSCAR HOME DEL EJE Y
    MousePointer = 11
    Dato = Chr$(13)
    Call Enviar(Me)
    Dato = Chr$(13)
    Call Enviar(Me)
    Call Retardo(0.2)
    Dato = "W" + "02" + "I" + Chr$(13)
    Call Enviar(Me)
    Call HomeY(Me)
    Call Retardo(0.2)
    ' SUSPENDE CONTROLES
    ControlesY.Enabled = False
    Controles.Enabled = False
    ControlesD.Enabled = False
    ' CONSULTA B PARA ASEGURAR LLEGADA A HOME
    Do While True
        Dato = "W" + "02" + "O 081H" + Chr$(13)
        Call Enviar(Me)
        Dato = "W02? B" + Chr$(13)
        Call Enviar(Me)
        Call Retardo(2)
        Ch1 = ""
        Do While MSComm1.InBufferCount > 0
            Ch1 = MSComm1.Input
            Call Retardo(0.5)
        Loop
        If Len(Ch1) > 0 Then
            Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
            Ch1 = Left(Ch1, Len(Ch1) - 1)
        End If
        If Ch1 = "65535" Then
            Exit Do
        End If
    Loop
    ' RETIRA 50 PASOS DEL HOME (DECLARA POSICION 0)
    Dato = "W" + "02" + "N 50" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "02" + "F 12" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "02" + "S 4" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "02" + "R 16" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "02" + "+" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "02" + "G" + Chr$(13)
    Call Enviar(Me)
    ' HABILITA CONTROLES
    ControlesY.Enabled = True
    Controles.Enabled = True
    ControlesD.Enabled = True
    DatosPosicionY.Text = 0
    PASOS_Y_CM.Text = 0
    posicion(7).Caption = 0
    PosicionY = 0
    ' FUERA DE HOME
    SignoY = 1
    Frame2.Enabled = True
    MousePointer = 0
End Sub

```

```

Private Sub Inicializa_Equipo_Click()
    MousePointer = 11

```

```

Call Inicio1(Me, 1)
DatosPosicion.Text = 0
PASOS_X_CM.Text = 0
DatosPosicionY.Text = 0
PASOS_Y_CM.Text = 0
posicion(0).Caption = 0
posicion(7).Caption = 0
PosicionX = 0
PosicionY = 0
' Inicializa signos positivos (para alejarse de home)
SignoX = 1
SignoY = 1
Frame2.Enabled = True
MousePointer = 0
End Sub

Private Sub MaxRateXVal_KeyPress(KeyAscii As Integer)
If KeyAscii > 24 Then
If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
End If
End Sub

Private Sub MaxRateYVal_KeyPress(KeyAscii As Integer)
If KeyAscii > 24 Then
If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
End If
End Sub

Private Sub OptimoX_Click() ' VALORES OPTIMOS PARA TRABAJO EN EJE X
Dato = "W" + "01" + "B 15H" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "L"
Call Enviar(Me)
SignoX = 1
PasosXVal.Text = 5000
FirstRateXVal.Text = 12
SlopeXVal.Text = 4
MaxRateXVal.Text = 16
' ACTUALIZAR VALORES
Call NumPasosX_Click
Call FirstRateX_Click
Call SlopeX_Click
Call MaxRateX_Click
End Sub

Private Sub OptimoY_Click() ' VALORES OPTIMOS PARA TRABAJO EN EJE Y
Dato = "W" + "02" + "+"
Call Enviar(Me)
SignoY = 1
PasosYVal.Text = 5000
FirstRateYVal.Text = 12
SlopeYVal.Text = 4
MaxRateYVal.Text = 16
' ACTUALIZAR VALORES
Call NumPasosY_Click
Call FirstRateY_Click
Call SlopeY_Click
Call MaxRateY_Click
End Sub

Private Sub CY5451_Click() ' INICIALIZAR EJE X
Call Inicializa_X(Me)
End Sub

Private Sub CY5452_Click() ' INICIALIZAR EJE Y
Call Inicializa_Y(Me)
End Sub

Private Sub Display_Click() ' ENVIAR TEXTO AL DISPLAY SELECCIONADO
Dim ADD As String
' SEGUN EL DISPLAY SELECCIONADO
If Display_X.Value = True Then
ADD = "01"
Else

```

```

    ADD = "02"
End If
Call Borrado
Dato = "W" + ADD + "O 085H" + Chr$(13)
Call Enviar(Me)
Dato = "W" + ADD + "*****" + TextoDisplay.Text + "*****" + Chr$(13)
Call Enviar(Me)
TextoDisplay.Text = ""
End Sub

```

```

Private Sub FirstRateX_Click() ' ACTUALIZA VEL. INICIAL EN X
    MousePointer = 11
    Dato = "W" + "01" + "F " + CStr((FirstRateXVal.Text))
    Call Enviar(Me)
    MSComm1.Output = Chr$(13)
    Retardo (0,2)
    posicion(3).Caption = FirstRateXVal.Text
    MousePointer = 0
End Sub

```

```

Private Sub FirstRateY_Click() ' ACTUALIZA VEL. INICIAL EN Y
    MousePointer = 11
    Dato = "W" + "02" + "F " + CStr((FirstRateYVal.Text))
    Call Enviar(Me)
    MSComm1.Output = Chr$(13)
    Retardo (0,2)
    posicion(10).Caption = FirstRateYVal.Text
    MousePointer = 0
End Sub

```

```

Private Sub Form_Load() ' SE EJECUTA CUANDO CARGA LA PANTALLA
    ' ABRE LA COMUNICACION SERIAL
    MousePointer = 0
    MSComm1.Settings = "2400, n, 8, 1"
    MSComm1.CommPort = 5
    MSComm1.PortOpen = True
    MSComm1.Output = Chr$(13)
    MSComm1.Output = Chr$(13)
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer) ' AL CERRAR LA PANTALLA
    Apaga_Cortador
    MSComm1.PortOpen = False ' CIERRA SERIAL
End Sub

```

```

Private Sub GoX_Click() ' EJECUTE EL MOVIMIENTO ESTABLECIDO
    If SignoX = 0 Then
        If PosicionX - CInt(PasosXVal.Text) < 0 Then
            MsgBox "NO ES POSIBLE ESE MOVIMIENTO", vbExclamation, "Advertencial"
            Exit Sub
        End If
    End If
    If SignoX = 1 Then
        If CLng(PosicionX + CLng(PasosXVal.Text)) > 29750# Then
            MsgBox "EL MOVIMIENTO SE ENCUENTRA FUERA DEL RANGO", vbExclamation, "Advertencial"
            Exit Sub
        End If
    End If
    MousePointer = 11
    Dato = "W" + "01" + "G"
    Call Enviar(Me)
    MSComm1.Output = Chr$(13)
    ' ACTUALIZA PANTALLA Y CONTROL DE POSICION
    If SignoX = 0 Then
        PosicionX = PosicionX - CInt(PasosXVal.Text)
    End If
    If SignoX = 1 Then
        PosicionX = PosicionX + CInt(PasosXVal.Text)
    End If
    DatosPosicion.Text = CStr(PosicionX)
    PASOS_X_CM.Text = (1.33 * CInt(DatosPosicion.Text)) / 1000
    posicion(0).Caption = PosicionX
    Dato = "W" + "01" + "B 03" + Chr$(13)
    Call Enviar(Me)

```

```
MousePointer = 0
End Sub
```

```
Private Sub GOY_Click() ' EJECUTE EL MOVIMIENTO ESTABLECIDO
If SignoY = 0 Then
If PosicionY - CInt(PasosYVal.Text) < 0 Then
MsgBox "NO ES POSIBLE ESE MOVIMIENTO", vbExclamation, "Advertencia!"
Exit Sub
End If
End If
If SignoY = 1 Then
If CLng(PosicionY + CLng(PasosYVal.Text)) > 29750# Then
MsgBox "EL MOVIMIENTO SE ENCUENTRA FUERA DEL RANGO", vbExclamation, "Advertencia!"
Exit Sub
End If
End If
MousePointer = 11
Dato = "W" + "02" + "G"
Call Enviar(Me)
MSComm1.Output = Chr$(13)
' ACTUALIZA PANTALLA Y CONTROL DE POSICION
If SignoY = 0 Then
PosicionY = PosicionY - CInt(PasosYVal.Text)
End If
If SignoY = 1 Then
PosicionY = PosicionY + CInt(PasosYVal.Text)
End If
DatosPosicionY.Text = CStr(PosicionY)
PASOS_Y_CM.Text = ((1.33 * (CStr(DatosPosicionY.Text)) / 1000))
posicion(7).Caption = PosicionY
DatosPosicionY.Text = CStr(PosicionY)
PASOS_Y_CM.Text = ((1.33 * (CStr(DatosPosicionY.Text)) / 1000))
Dato = "W" + "01" + "B 03" + Chr$(13)
Call Enviar(Me)
MousePointer = 0
End Sub
```

```
Private Sub InicializarX_Click() ' INICIALIZA EJE X
MousePointer = 11
Dato = Chr$(13)
Call Enviar(Me)
Dato = Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "I" + Chr$(13)
Call Enviar(Me)
Call ActualizacionX(Me)
Dato = "W01" + "B 15H" + Chr$(13)
Call Enviar(Me)
MousePointer = 0
End Sub
```

```
Private Sub InicializarY_Click() ' INICIALIZA EJE X
MousePointer = 11
Dato = Chr$(13)
Call Enviar(Me)
Dato = Chr$(13)
Call Enviar(Me)
Dato = "W" + "02" + "I" + Chr$(13)
Call Enviar(Me)
Call ActualizacionY(Me)
Dato = "W" + "02" + "-" + Chr$(13)
Call Enviar(Me)
MousePointer = 0
End Sub
```

```
Private Sub MasX_Click() ' ALEJARLE DE HOME X
MousePointer = 11
Dato = "W" + "01" + "-"
Call Enviar(Me)
MSComm1.Output = Chr$(13)
Retardo (0.1)
MousePointer = 0
SignoX = 1
End Sub
```

```
Private Sub MasY_Click() ' ALEJARLE DE HOME Y
```

```
    MousePointer = 11  
    Dato = "W" + "02" + "+"  
    Call Enviar(Me)  
    MSComm1.Output = Chr$(13)  
    Retardo (0,1)  
    MousePointer = 0  
    SignoY = 1
```

```
End Sub
```

```
Private Sub MaxRateX_Click() ' FIJA VEL. MAX EN X
```

```
    MousePointer = 11  
    Dato = "W" + "01" + "R " + CStr((MaxRateXVal.Text))  
    Call Enviar(Me)  
    MSComm1.Output = Chr$(13)  
    Retardo (0,2)  
    posicion(1).Caption = MaxRateXVal.Text  
    MousePointer = 0
```

```
End Sub
```

```
Private Sub MaxRateY_Click() ' FIJA VEL. MAX EN Y
```

```
    MousePointer = 11  
    Dato = "W" + "02" + "R " + CStr((MaxRateYVal.Text))  
    Call Enviar(Me)  
    MSComm1.Output = Chr$(13)  
    Retardo (0,2)  
    posicion(8).Caption = MaxRateYVal.Text  
    MousePointer = 0
```

```
End Sub
```

```
Private Sub MenosX_Click() ' HACIA HOME X
```

```
    MousePointer = 11  
    Dato = "W" + "01" + "+"  
    Call Enviar(Me)  
    MSComm1.Output = Chr$(13)  
    Retardo (0,1)  
    MousePointer = 0  
    SignoX = 0
```

```
End Sub
```

```
Private Sub MenosY_Click() ' HACIA HOME Y
```

```
    MousePointer = 11  
    Dato = "W" + "02" + "-"  
    Call Enviar(Me)  
    MSComm1.Output = Chr$(13)  
    Retardo (0,1)  
    MousePointer = 0  
    SignoY = 0
```

```
End Sub
```

```
Private Sub MenuInicial_Click() ' SALE DE PANTALLA, REGRESA AL MENU
```

```
    Unload Me  
    Form2.Show
```

```
End Sub
```

```
Private Sub MSComm1_OnComm() ' EVENTO EN COMUNICACIONES
```

```
    MsgBox "Error en la comunicación serial..", vbExclamation, "Advertencia!"
```

```
End Sub
```

```
Private Sub NumPasosX_Click() ' FIJA NUMERO DE PASOS EJE X
```

```
    MousePointer = 11  
    Dato = "W" + "01" + "N " + CStr(PasosXVal.Text)  
    Call Enviar(Me)  
    MSComm1.Output = Chr$(13)  
    Retardo (0,2)  
    posicion(4).Caption = PasosXVal.Text  
    MousePointer = 0
```

```
End Sub
```

```
Private Sub NumPasosY_Click() ' FIJA NUMERO DE PASOS EJE Y
```

```
    MousePointer = 11  
    Dato = "W" + "02" + "N " + CStr(PasosYVal.Text)  
    Call Enviar(Me)  
    MSComm1.Output = Chr$(13)
```

```
Retardo (0,2)
posicion(11).Caption = PasosYVal.Text
MousePointer = 0
End Sub
```

```
Private Sub Parametros_Click() ' DESPLIEGA EN DISPLAY EL PARAMETRO ELEGIDO
```

```
Dim ADD As String
Dim ParametroDisplay As String
Dim I As Integer
```

```
MousePointer = 11
Call Borrado
' QUE DIRECCION ?
If Display_X.Value = True Then
ADD = "01"
Else
ADD = "02"
End If
' QUE PARAMETRO ?
For I = 1 To 6
If Option3(I).Value Then
Exit For
End If
Next I
Select Case I
Case Is = 1
ParametroDisplay = "P"
Case Is = 2
ParametroDisplay = "R"
Case Is = 3
ParametroDisplay = "S"
Case Is = 4
ParametroDisplay = "F"
Case Is = 5
ParametroDisplay = "N"
Case Is = 5
ParametroDisplay = "Y"
End Select
' ENVIO DE DATOS AL DISPLAY
Dato = "W" + ADD + "O 085H"
Call Enviar(Me)
MSComm1.Output = Chr$(13)
Retardo (0,2)
' ENVIO DE DATOS PARA IMPRIMIR PARAMETROS
Dato = "W" + ADD + "? " + ParametroDisplay
Call Enviar(Me)
MSComm1.Output = Chr$(13)
Retardo (0,2)
MousePointer = 0
End Sub
```

```
Private Sub PasosXVal_KeyPress(KeyAscii As Integer)
If KeyAscii > 24 Then
If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
End If
End Sub
```

```
Private Sub PasosYVal_KeyPress(KeyAscii As Integer)
If KeyAscii > 24 Then
If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
End If
End Sub
```

```
Private Sub SlopeX_Click() ' FIJA SLOPE EN EJE X
MousePointer = 11
Dato = "W" + "01" + "S " + CStr(SlopeXVal.Text) + "H" + Chr$(13)
Call Enviar(Me)
posicion(2).Caption = CStr(SlopeXVal.Text)
MousePointer = 0
End Sub
```

```
Private Sub SlopeXVal_KeyPress(KeyAscii As Integer)
If KeyAscii > 24 Then
If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
End If
```

End Sub

```
Private Sub SlopeY_Click() ' FIJA SLOPE EN EJE Y
    MousePointer = 11
    Dato = "W" + "02" + "S " + CStr(SlopeYVal.Text) + "H" + Chr$(13)
    Call Enviar(Me)
    posicion(9).Caption = CStr(SlopeYVal.Text)
    MousePointer = 0
End Sub
```

```
Private Sub SlopeYVal_KeyPress(KeyAscii As Integer)
    If KeyAscii > 24 Then
        If InStr(Numeros, Chr(KeyAscii)) = 0 Then KeyAscii = 0
    End If
End Sub
```

```
Private Sub Timer1_Timer()
    Timer1.Enabled = False
    MsgBox "HAY ERROR DE COMUNICACION SERIAL..", vbExclamation, "Advertencia!"
End Sub
End Sub
```

DIBUJO.FRM

```
Dim N As Integer
Dim Indice As Long ' CONTROLA LO QUE SE GUARDA EN LAS MATRICES X y Y
Dim Escala As Integer
```

```
Private Sub Dibujar()
    Picture1.Cls
    For I = 1 To Indice_Lineas
        Picture1.Line (Matriz_Lineas(I, 1) / Escala, Matriz_Lineas(I, 2) / Escala)-(Matriz_Lineas(I, 3) / Escala, Matriz_Lineas(I, 4) / Escala)
    Next I
End Sub
```

```
Private Sub Linea_horizontal(ByVal X1 As Long, ByVal Y1 As Long, ByVal X2 As Long, ByVal Y2 As Long)
    Matriz_Lineas(Indice, 1) = X1
    Matriz_Lineas(Indice, 2) = Y1
    Matriz_Lineas(Indice, 3) = X2
    Matriz_Lineas(Indice, 4) = Y2
    Matriz_Lineas(Indice, 5) = 15
    Matriz_Lineas(Indice, 6) = 0
    Indice = Indice + 1
End Sub
```

```
Private Sub Linea_Vertical(ByVal X1 As Long, ByVal Y1 As Long, ByVal X2 As Long, ByVal Y2 As Long)
    Matriz_Lineas(Indice, 1) = X1
    Matriz_Lineas(Indice, 2) = Y1
    Matriz_Lineas(Indice, 3) = X2
    Matriz_Lineas(Indice, 4) = Y2
    Matriz_Lineas(Indice, 5) = 0
    Matriz_Lineas(Indice, 6) = 15
    Indice = Indice + 1
End Sub
```

```
Private Sub Linea(ByVal X1 As Long, ByVal Y1 As Long, ByVal X2 As Long, ByVal Y2 As Long)
    Dim Difx As Long
    Dim Dify As Long
    Dim Razon As Currency
    Dim RazonTemp As Currency
    Dim Factor As Integer
    Dim FactorTemp As Integer
    Dim Velocidad_Menor As Integer
    ' DEBEN SALIR 6 PARAMETROS GRABADOS EN MATRIZ_LINEAS
    Difx = (X2 - X1)
    Dify = (Y2 - Y1)
    If Dify = 0 Then
        Call Linea_horizontal(X1, Y1, X2, Y2)
        Exit Sub
    End If
    If Difx = 0 Then
        Call Linea_Vertical(X1, Y1, X2, Y2)
        Exit Sub
    End If
```

```

End If
If Difx < 0 Then
  Difx = X1 - X2
End If
If Dify < 0 Then
  Dify = Y1 - Y2
End If
If Difx > Dify Then
  Razon = ((Dify * 15) / Difx)
  RazonTemp = Abs(Round(Razon) - Razon)
  If Razon < 1 Then
    Call Linea_horizontal(X1, Y1, X2, Y1)
    Exit Sub
  End If
  FactorTemp = 15
  Factor = 14
  For J = 1 To 14
    Razon = ((Dify * Factor) / Difx)
    If ((Abs(Round(Razon) - Razon)) <= RazonTemp) Or ((Abs(Round(Razon) - Razon)) - RazonTemp <= 0.02) Then
      RazonTemp = Abs(Round(Razon) - Razon)
      FactorTemp = Factor
      Velocidad_Menor = Round(Razon)
    End If
    Factor = Factor - 1
  Next J
  Matriz_Lineas(Indice, 1) = X1
  Matriz_Lineas(Indice, 2) = Y1
  Matriz_Lineas(Indice, 3) = X2
  Matriz_Lineas(Indice, 4) = Y2
  If Velocidad_Menor < 4 And FactorTemp <= 7 Then
    Velocidad_Menor = Velocidad_Menor * 2
    FactorTemp = FactorTemp * 2
  End If
  Matriz_Lineas(Indice, 5) = FactorTemp
  Matriz_Lineas(Indice, 6) = Velocidad_Menor
  Indice = Indice + 1
Else
  Razon = ((Difx * 15) / Dify)
  RazonTemp = Abs(Round(Razon) - Razon)
  If Razon < 1 Then
    Call Linea_Vertical(X1, Y1, X1, Y2)
    Exit Sub
  End If
  FactorTemp = 15
  Factor = 14
  For J = 1 To 14
    Razon = ((Difx * Factor) / Dify)
    If ((Abs(Round(Razon) - Razon)) <= RazonTemp) Or ((Abs(Round(Razon) - Razon)) - RazonTemp <= 0.02) Then
      RazonTemp = Abs(Round(Razon) - Razon)
      FactorTemp = Factor
      Velocidad_Menor = Round(Razon)
    End If
    Factor = Factor - 1
  Next J
  Matriz_Lineas(Indice, 1) = X1
  Matriz_Lineas(Indice, 2) = Y1
  Matriz_Lineas(Indice, 3) = X2
  Matriz_Lineas(Indice, 4) = Y2
  If Velocidad_Menor < 4 And FactorTemp <= 7 Then
    Velocidad_Menor = Velocidad_Menor * 2
    FactorTemp = FactorTemp * 2
  End If
  Matriz_Lineas(Indice, 5) = Velocidad_Menor
  Matriz_Lineas(Indice, 6) = FactorTemp
  Indice = Indice + 1
End If
End Sub

```

```

Private Sub Arco(ByVal XC As Long, ByVal YC As Long, ByVal R As Long, ByVal Ang_I As Integer, ByVal Ang_F As Integer) ' CENTRO X, CENTRO Y, RADIO, ANG INICIAL ,ANG FINAL
Dim Angulo As Double
Dim Numpasos As Long
Dim Inc_Angulo As Double
Dim Ang_Radianes As Double

```

```

Dim Dif_Ang As Long
Dim I As Integer
Const Factor = 20000
Angulo = (Factor / R) 'ENCUENTRA EL ANGULO DE GIRO (grados)
If Ang_I > Ang_F Then
  Ang_F = Ang_I + 360
End If
Dif_Ang = (Ang_F - Ang_I)
If Dif_Ang = 0 Then Exit Sub
Matriz_Lineas(Indice, 1) = XC + Round(R * Cos(Ang_I * (2 * 3.14159265358979) / 360))
Matriz_Lineas(Indice, 2) = YC + Round(R * Sin(Ang_I * (2 * 3.14159265358979) / 360))
Numpasos = Abs(Dif_Ang) * R / Factor
Inc_Angulo = Ang_I
For J = 2 To Numpasos
  Inc_Angulo = Inc_Angulo + Angulo
  If Inc_Angulo > 360 Then
    Inc_Angulo = Inc_Angulo - 360
  End If
  Ang_Radianes = Inc_Angulo * (2 * 3.14159265358979) / 360 ' FUNCIONES SENOS Y COSENO EL ANGULO EN
RADIANTES
  X = R * Cos(Ang_Radianes)
  Y = R * Sin(Ang_Radianes)
  Matriz_Lineas(Indice, 3) = XC + Round(X)
  Matriz_Lineas(Indice, 4) = YC + Round(Y)
  Indice = Indice + 1
  Matriz_Lineas(Indice, 1) = Matriz_Lineas(Indice - 1, 3)
  Matriz_Lineas(Indice, 2) = Matriz_Lineas(Indice - 1, 4)
Next J
Inc_Angulo = Inc_Angulo + Angulo
Ang_Radianes = Inc_Angulo * (2 * 3.14159265358979) / 360 ' FUNCIONES SENOS Y COSENO EL ANGULO EN
RADIANTES
X = R * Cos(Ang_Radianes)
Y = R * Sin(Ang_Radianes)
Matriz_Lineas(Indice, 3) = XC + Round(X)
Matriz_Lineas(Indice, 4) = YC + Round(Y)
Indice = Indice + 1
End Sub

Private Sub Cimco_Memoria_Click()
Dim Linea1 As String
Dim Auxiliar As String
Dim Long_Aux As Integer
Dim X1, X2, Y1, Y2, Ang_I, Ang_F, N As Long
Dim Nombre As String
Archivo_CBC.Visible = False
Indice = 1
Indice_Lineas = 1
CommonDialog1.Filter = ".txt File (*.txt)|*.text|All Files (*.*)|*.*"
CommonDialog1.DialogTitle = "Selección de Archivo CIMCO"
CommonDialog1.FileName = "*.txt"
CommonDialog1.CancelError = -1
On Error GoTo Cancelo
CommonDialog1.ShowOpen
If Err <> 0 Then
  MsgBox "Error en apertura de Archivo..", vbExclamation
  Exit Sub
End If
Nombre = CommonDialog1.FileName
On Error GoTo Cancelo
Open Nombre For Input As #1
Input #1, Tipo
If Tipo <> "CNC-Calc 2000" Then
  MsgBox "Este archivo no ha sido generado por CIMCO", vbExclamation, "Archivo Erróneo"
  Close
  Exit Sub
End If
Input #1, Tipo
If Tipo <> "Text File" Then
  MsgBox "Este archivo no está en formato de Texto", vbExclamation, "Archivo Erróneo"
  Close
  Exit Sub
End If
Auxiliar = Nombre
Do While InStr(Auxiliar, "\") > 0

```

```

If Err <> 0 Then
    MsgBox "Error en grabación de Archivo..", vbExclamation
    Exit Sub
End If
Nombre = CommonDialog1.FileName
Open Nombre For Output As #1
Print #1, "CBC"
Print #1, CStr(15) + vbTab + CStr(15)
For l = 1 To Indice_Lineas - 1
    Print #1, CStr(Matriz_Lineas(l, 1)) + vbTab + CStr(Matriz_Lineas(l, 2))
    Print #1, CStr(Matriz_Lineas(l, 5)) + vbTab + CStr(Matriz_Lineas(l, 6))
Next l
Print #1, CStr(Matriz_Lineas(l - 1, 3)) + vbTab + CStr(Matriz_Lineas(l - 1, 4))
Print #1, CStr(15) + vbTab + CStr(15)
Print #1, CStr(0) + vbTab + CStr(50)
Close #1
Cancelo:
End Sub

Private Sub Construye_Grafico()
    MousePointer = 11
    Call Ordenar_Lineas ' SALE ORDENADO EN MATRIZ_COPIA
    If Not Cerrado Then
        l = MsgBox("El Poligono no está cerrado. Desea continuar de todas maneras?", vbQuestion + vbYesNo, "Advertencial")
        If l = vbNo Then
            Exit Sub
        End If
    End If
    ' LIMPIAR MATRIZ_LINEAS
    For N = 1 To Indice_Lineas
        Matriz_Lineas(N, 1) = 0
        Matriz_Lineas(N, 2) = 0
        Matriz_Lineas(N, 3) = 0
        Matriz_Lineas(N, 4) = 0
    Next N
    Indice = 1
    For l = 1 To Indice_Lineas - 1 ' PARA CALCULO DE VELOCIDADES OPTIMAS
        Call Linea(Matriz_Copia(l, 1), Matriz_Copia(l, 2), Matriz_Copia(l, 3), Matriz_Copia(l, 4))
    Next l
    Call Dibujar
    Archivo_CBC.Visible = True
    MousePointer = 0
End Sub

Private Sub Form_Load()
    Me.Top = 0
    Me.Left = 0
    Picture1.Width = 14175
    Picture1.Height = 9420
    Escala = 3
End Sub

Private Sub MenuInicial_Click()
    Unload Me
    Form2.Show
End Sub

AUTO.FRM
Dim Interrumpir As Boolean ' Interrumpir o no el corte
Dim Salva_Punto As Integer ' Auxiliar salva el último punto cortado
Dim Desplazamiento_X As Double ' Primero o segundo cuadrante

Private Sub Enciende_Cortador() ' ACTIVA EL CORTADOR
    Dato = "W" + "01" + "O 081H" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "01" + "B 13H" + Chr$(13)
    Call Enviar(Me)
End Sub

Private Sub Apaga_Cortador() ' DESACTIVA EL CORTADOR
    Dato = "W" + "01" + "O 081H" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "01" + "B 03H" + Chr$(13)
    Call Enviar(Me)

```

End Sub

```
Private Sub Parametros_Iniciales() ' CARGA PARAMETROS OPTIMOS
Dim Ch1 As String
'EJE X
Dato = "W" + "01" + "B 15H" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "F 12" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "S 4" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "R 16" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "-" + Chr$(13)
Call Enviar(Me)
'EJE Y
Dato = "W" + "02" + "F 12" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "02" + "S 4" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "02" + "R 16" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "02" + "-" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "B 15H" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "O 081H" + Chr$(13)
Call Enviar(Me)
Dato = "W01? F" + Chr$(13)
Call Enviar(Me)
Retardo (0.2)
Ch1 = ""
Retardo (0.5)
Do While MSComm1.InBufferCount > 0
    Ch1 = MSComm1.Input
    Call Retardo(0.3)
Loop
Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
FirstRateX.Caption = Left(Ch1, Len(Ch1) - 1)
Dato = "W01? S" + Chr$(13)
Call Enviar(Me)
Retardo (0.3)
Ch1 = ""
Do While MSComm1.InBufferCount > 0
    Ch1 = MSComm1.Input
    Call Retardo(0.3)
Loop
Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
SlopeX.Caption = Left(Ch1, Len(Ch1) - 1)
Dato = "W01? R" + Chr$(13)
Call Enviar(Me)
Retardo (0.3)
Ch1 = ""
Do While MSComm1.InBufferCount > 0
    Ch1 = MSComm1.Input
    Call Retardo(0.3)
Loop
Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
MaxRateX.Caption = Left(Ch1, Len(Ch1) - 1)
Dato = "W" + "02" + "O 081H" + Chr$(13)
Call Enviar(Me)
Dato = "W02? F" + Chr$(13)
Call Enviar(Me)
Call Retardo(0.3)
Ch1 = ""
Do While MSComm1.InBufferCount > 0
    Ch1 = MSComm1.Input
    Call Retardo(0.3)
Loop
If Len(Ch1) > 0 Then
    Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
    FirstRateY.Caption = Left(Ch1, Len(Ch1) - 1)
End If
Dato = "W02? S" + Chr$(13)
```

```

Call Enviar(Me)
Call Retardo(0.3)
Ch1 = ""
Do While MSComm1.InBufferCount > 0
    Ch1 = MSComm1.Input
    Call Retardo(0.3)
Loop
If Len(Ch1) > 0 Then
    Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
    SlopeY.Caption = Left(Ch1, Len(Ch1) - 1)
End If
Dato = "W02? R" + Chr$(13)
Call Enviar(Me)
Call Retardo(0.3)
Ch1 = ""
Do While MSComm1.InBufferCount > 0
    Ch1 = MSComm1.Input
    Call Retardo(0.3)
Loop
If Len(Ch1) > 0 Then
    Ch1 = Right(Ch1, Len(Ch1) - InStr(Ch1, "="))
    MaxRate.Caption = Left(Ch1, Len(Ch1) - 1)
End If
Contenedor.Enabled = True
End Sub

```

```

Private Sub Form_Activate()
Dim Ch1 As String
Inicializa_Equipo.Enabled = False
If 1 = 2 Then
    Timer1.Enabled = True
    Ch1 = ""
Do While True
    DoEvents
    Dato = "W" + "01" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "01" + Chr$(13)
    Call Enviar(Me)
    Dato = "W" + "01" + "? B" + Chr$(13)
    Call Enviar(Me)
Do While MSComm1.InBufferCount > 0
    Ch1 = MSComm1.Input
    Call Retardo(0.3)
Loop
If Len(Ch1) > 0 Then
    If InStr(Ch1, "=") > 0 Then
        Timer1.Enabled = False
        Exit Do
    End If
End If
Loop
End If
Inicializa_Equipo.Enabled = True
Inicializa_Equipo.SetFocus
Call Apaga_Cortador
Inicializa_Equipo.SetFocus
End Sub

```

```

Private Sub Form_Load()
MSComm1.Settings = "2400, n, 8, 1" ' ESTABLECE COMUNICACION SERIAL
MSComm1.CommPort = 5
MSComm1.PortOpen = True
MSComm1.Output = "W" + "01" + Chr$(13)
MSComm1.Output = "W" + "01" + Chr$(13)
MSComm1.Output = "W" + "02" + Chr$(13)
MSComm1.Output = "W" + "02" + Chr$(13)
I_Cuadrante.Value = True ' Primer cuadrante (default)
Interrumpir = False
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
MSComm1.PortOpen = False
End Sub

```

```

Private Sub Inicializa_Equipo_Click() ' INICIALIZA EQUIPO
    MousePointer = 11
    Call Inicio1(Me, 2)
    Call Parametros_Iniciales
    Iniciar.Visible = True
    MousePointer = 0
End Sub

Private Sub Iniciar_Click() ' CARGA ARCHIVO CBC A MEMORIA
Dim I As Integer
Dim Impar As Boolean ' Discrimina velocidades y coordenadas
' ARCHIVO DE CORTES
CommonDialog1.CancelError = -1
CommonDialog1.Filter = "CBC File (*.CBC)|*.CBC|All Files (*.*)|*.*|"
CommonDialog1.DialogTitle = "Selección de Archivo"
On Error GoTo Cancelo
CommonDialog1.ShowOpen
If Err <> 0 Then
    MsgBox "Error en apertura de Archivo..", vbExclamation
    Exit Sub
End If
Nombre = CommonDialog1.FileName
NumPuntos = 0
' OBTENER INFORMACION DEL ARCHIVO
Open Nombre For Input As #1
Input #1, Tipo
If Tipo <> "CBC" Then
    MsgBox "Este archivo no ha sido generado para el sistema CBC", vbExclamation, "Archivo Erróneo"
    Close
    Exit Sub
End If
NombreArchivo.Caption = Nombre
' PRIMER PAR ES DE VELOCIDADES, SIGUIENTE SON COORDENADAS
If I_Cuadrante.Value Then
    Desplazamiento_X = 0
Else
    Desplazamiento_X = 17000
End If
Impar = True
Do While Not EOF(1)
    NumPuntos = NumPuntos + 1
    If Impar Then
        ' PAR DE VELOCIDADES
        Input #1, PosX(NumPuntos), PosY(NumPuntos)
    Else
        ' PAR DE COORDENADAS
        Input #1, PosX(NumPuntos), PosY(NumPuntos)
        PosX(NumPuntos) = PosX(NumPuntos) + Desplazamiento_X
    End If
    If Impar Then
        Impar = False
    Else
        Impar = True
    End If
Loop
Close #1
NumPuntos = NumPuntos / 2
If NumPuntos = 0 Then
    MsgBox "El archivo no tiene información para cortes..", vbInformation, "Archivo vacío"
    Exit Sub
End If
Call Cortar(1, NumPuntos)
Cancelo:
End Sub

Private Sub Interrupcion_Click() ' INTERRUMPE EL CORTE
    Interrumpir = True
    Reiniciar.Visible = True
End Sub

Private Sub MSComm1_OnComm() ' EVENTO EN COMUNICACIONES
    MsgBox "Error en la comunicación serial..", vbExclamation, "Advertencia!"
End Sub

```

```

Private Sub Regresoinicial_Click() ' REGRESA A MENU PRINCIPAL
Unload Me
Form2.Show
End Sub

' RUTINA DE CORTES
Private Sub Cortar(Desde As Integer, Hasta As Integer)
Dim X_OK, Y_OK As Boolean ' Alcanzo la posicion ?
Dim X_Destino, Y_Destino, Velocidad_X, Velocidad_Y As Integer
Iniciar.Visible = False
Regresoinicial.Visible = False
Reiniciar.Visible = False
Interrupcion.Visible = True
SSFrame4.Refresh
' FIJA MODO DE TRABAJO EN AMBOS CY545
Dato = "W" + "02" + "O 081H" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "O 081H" + Chr$(13)
Call Enviar(Me)
TotalPuntos.Caption = CStr(NumPuntos * 2)
I = Desde
Do While I < Hasta * 2
DoEvents
If Interrumpir = True Then
Apaga_Cortador
MsgBox "Se ha interrumpido el trabajo", vbExclamation, "Interrupción"
Interrupcion.Visible = False
Salva_Punto = I
Regresoinicial.Visible = True
SSFrame4.Refresh
Exit Sub
End If

' TOMA DATOS DE MEMORIA
Velocidad_X = PosX(I)
Velocidad_Y = PosY(I)
I = I + 1
X_Destino = PosX(I)
Y_Destino = PosY(I)

' DATOS EJE Y
Dato = "W02R " + CStr(Velocidad_Y) + Chr$(13)
Call Enviar(Me)
Retardo (0.02)
Dato = "W02F " + CStr(Velocidad_Y) + Chr$(13)
Call Enviar(Me)
Retardo (0.02)
FirstRateY.Caption = String(5 - Len(CStr(Velocidad_Y)), "0") + CStr(Velocidad_Y)
MaxRateY.Caption = String(5 - Len(CStr(Velocidad_Y)), "0") + CStr(Velocidad_Y)
Enciende_Cortador
Dato = "W" + "02" + "P " + CStr(Y_Destino) + Chr$(13)
Call Enviar(Me)

' DATOS EJE X
Dato = "W01R " + CStr(Velocidad_X) + Chr$(13)
Call Enviar(Me)
Retardo (0.02)
Dato = "W01F " + CStr(Velocidad_X) + Chr$(13)
Call Enviar(Me)
Retardo (0.02)
FirstRateX = String(5 - Len(CStr(Velocidad_X)), "0") + CStr(Velocidad_X)
MaxRateX = String(5 - Len(CStr(Velocidad_X)), "0") + CStr(Velocidad_X)
Dato = "W" + "01" + "P " + LTrim(CStr(16777166 - X_Destino)) + Chr$(13)
Call Enviar(Me)
Retardo (0.02)
X_OK = False
Y_OK = False
Do While True ' VERIFICA POSICIONES ALCANZADAS o ESPERA
Ch1 = ""
If Not X_OK Then ' X TODAVIA NO ALCANZA
Dato = "W" + "01" + "O 081H" + Chr$(13)
Call Enviar(Me)
Dato = "W" + "01" + "? P" + Chr$(13)
Call Enviar(Me)
Retardo (0.02)
End If
If Not Y_OK Then ' Y TODAVIA NO ALCANZA
Dato = "W" + "02" + "O 081H" + Chr$(13)

```

```

Call Enviar(Me)
Dato = "W" + "02" + "? P" + Chr$(13)
Call Enviar(Me)
Retardo (0.02)
End If
Do While MSComm1.InBufferCount > 0
  Ch1 = MSComm1.Input
Loop
If Len(Ch1) > 0 Then
  If Not X_OK Then
    If InStr(Ch1, "R01P=") > 0 Then
      On Error Resume Next
      If CDBl(Mid(Ch1, InStr(Ch1, "R01P=") + 5, 8)) = CDBl(CStr(16777166 - PosX(I))) Then
        X_OK = True      ' ALCANZO EN X
      End If
    End If
    End If
    End If
    If Not Y_OK Then
      If InStr(Ch1, "R02P=") > 0 Then
        On Error Resume Next
        If CDBl(Mid(Ch1, InStr(Ch1, "R02P=") + 5, 8)) = CDBl(CStr(PosY(I))) Then
          Y_OK = True      ' ALCANZO EN Y
        End If
      End If
    End If
    End If
    If X_OK And Y_OK Then      ' YA ALCANZO EN AMBOS EJES
      Exit Do
    End If
  Loop
  Apaga_Cortador
  ProgressBar1.Value = CInt(100 * I / (NumPuntos * 2))
  PuntosCortados.Caption = CStr(I)
  I = I + 1
Loop
ProgressBar1.Value = 100
MsgBox "Tarea concluída..", vbInformation, "Información"
Interrupcion.Visible = False
Iniciar.Visible = False
RegresoInicial.Visible = True
Reiniciar.Visible = False
NombreArchivo.Caption = ""
ProgressBar1.Value = 0
SSFrame4.Refresh
End Sub

Private Sub Reiniciar_Click()      ' REINICIA LUEGO DE SUSPENSION
Reiniciar.Visible = False
Iniciar.Visible = False
RegresoInicial = False
Interrupcion.Visible = True
Interrumpir = False
SSFrame4.Refresh
Call Cortar(Salva_Punto, NumPuntos) ' Salva_punto: último punto cortado
End Sub

Private Sub Timer1_Timer()      ' Se dispara si no inicia Tx serial
Timer1.Enabled = False
MsgBox "HAY ERROR DE COMUNICACION SERIAL..", vbExclamation, "Advertencia!"
End
End Sub

```

ANEXO 3.2

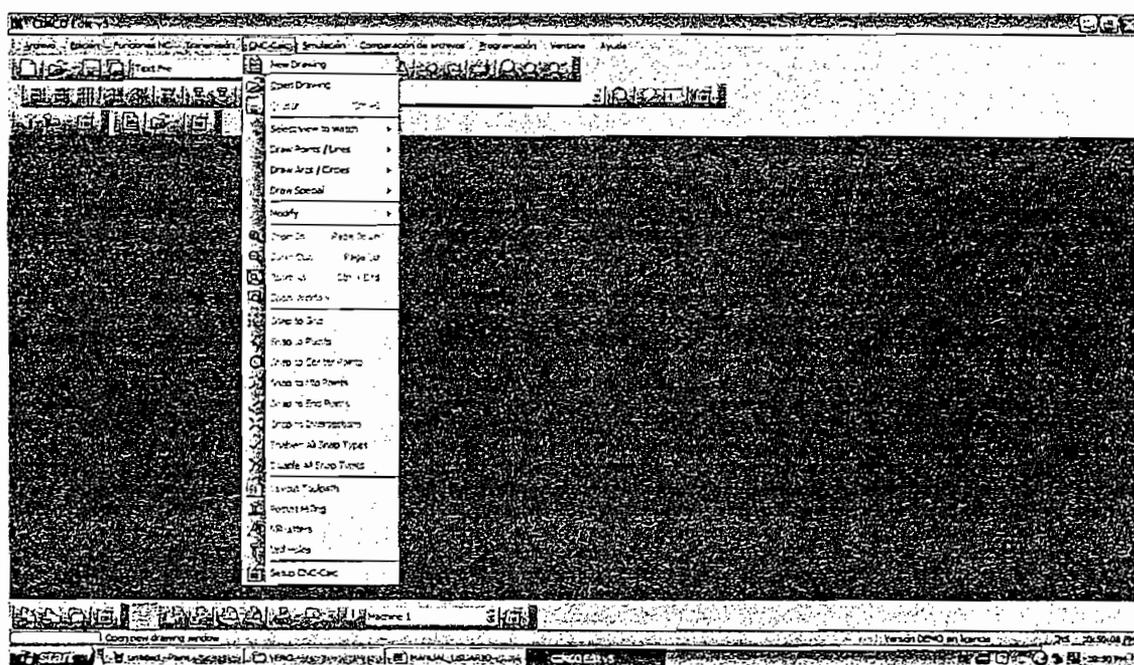
MANUAL DEL USUARIO.

MANUAL DEL USUARIO

CIMCO EDIT V.5

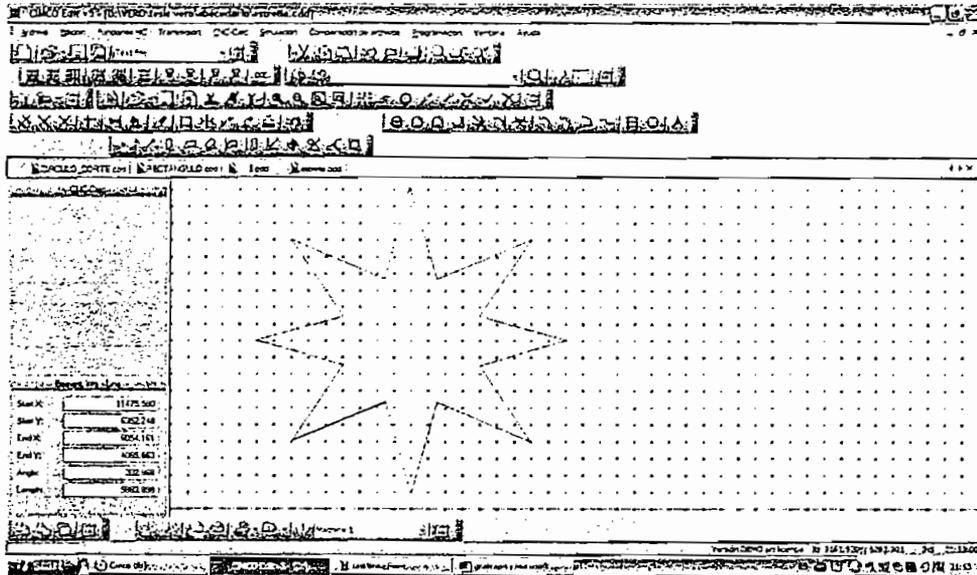
Para cumplir con el objetivo de desarrollar e implementar una aplicación visual que permite implementar un sistema de corte bidimensional controlado por computadora, es necesario escoger un programa que permita realizar el diagrama de la figura que se desea recortar.

El programa escogido para cumplir con la función de diseño de gráficos, es el CIMCO EDIT V.5. Para acceder a la pantalla principal, sobre el cual se realizarán los diseños deseados, se deben seguir los pasos que se muestran en la figura.



Se debe tener la precaución de escoger la opción "**Text File**", que se encuentra en la barra de herramientas del Menú Archivos; si no se mantiene esta condición, el programa "**Cortador_Bi_Comp.exe**" no reconocerá al archivo como uno generado por el programa CIMCO.

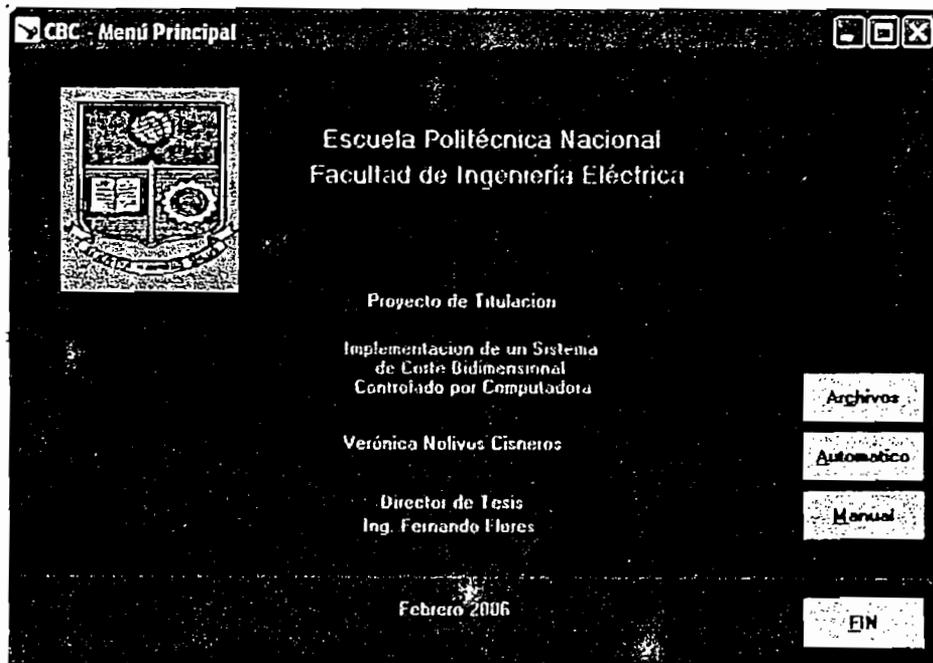
Otra consideración a tomar, es que el área de trabajo está representado por el primer cuadrante, y las dimensiones de los ejes X y Y, son de 30000 y 17000 respectivamente, que representan al plano de corte del equipo.



Al diseñar un gráfico en el programa CIMCO, es necesario guardar el archivo con extensión **"TXT"**, por ejemplo **"Estrella.txt"**.

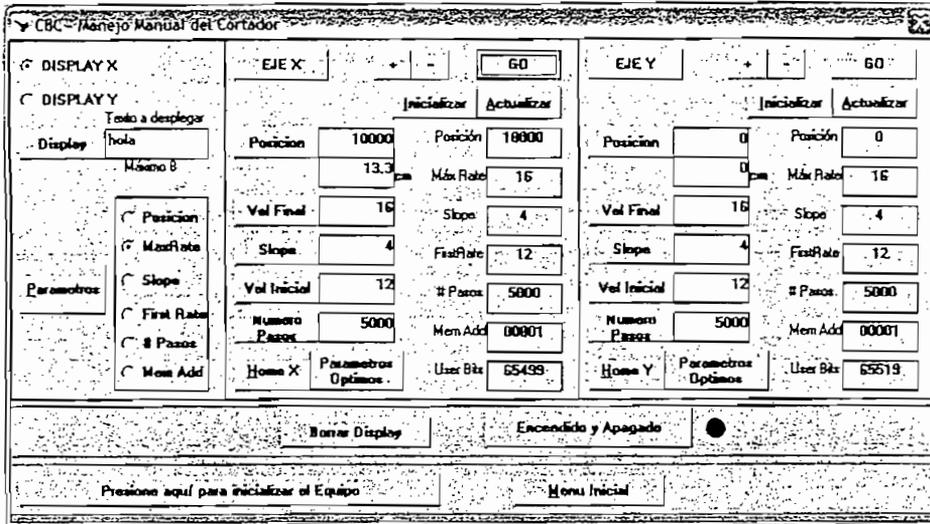
CORTADOR_BI_COMP.EXE

Al ejecutar el programa **"Cortador_Bi_Comp.exe"**, la presentación o Pantalla Principal que se observará es:



En ésta pantalla, el usuario puede acceder a los modos de trabajo **"Manual"** y **"Automático"**, según su necesidad.

El Modo de trabajo Manual, posee la siguiente presentación:



Para iniciar el modo Manual de trabajo, hacer clic en el botón **“Presione aquí para inicializar el equipo”**, con lo cual la herramienta de corte se desplazará hacia el origen del cuadrante. En ese momento se cargan los parámetros de trabajo.

En éste punto el usuario puede escoger parámetros de velocidad (pasos/seg) que deben adquirir los ejes para su desplazamiento, o puede escoger la opción **“Parámetros Optimos”** cargando los datos con los que el equipo trabajará eficientemente, y para producir el movimiento de los ejes se utiliza el botón **“GO”**.

Para escoger la dirección de desplazamiento de los ejes se tienen los botones **“+”** y **“-”**. Con el botón **“Actualizar”** se cargan los parámetros en caso de no trabajar con los parámetros óptimos. El botón **“Inicializar”**, permite resetear los datos cargados en la pantalla.

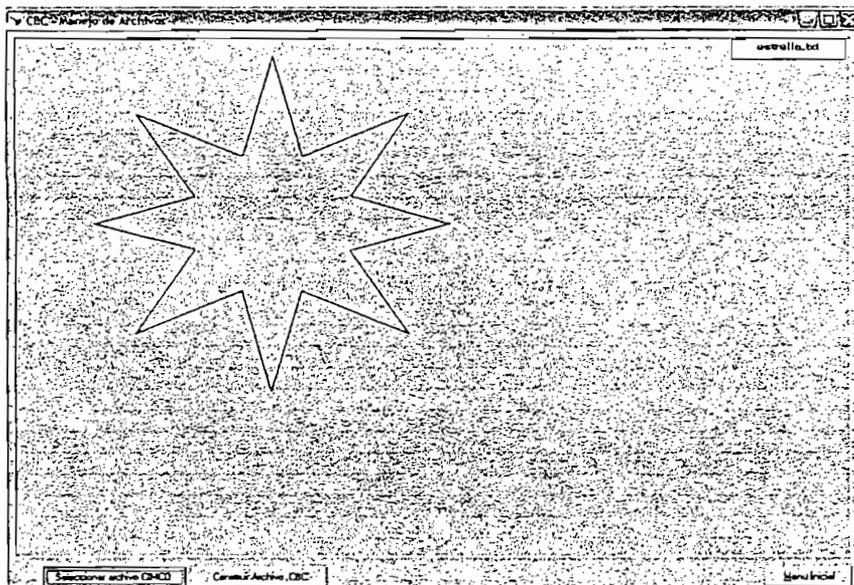
Adicionalmente, el usuario puede desplegar mensajes o parámetros que se han cargado en el equipo, los mismos que se observarán sobre display's que se encuentran en el equipo de corte.

Como se observa, ésta pantalla permite verificar el buen funcionamiento del equipo y de la herramienta de corte, por lo que también se puede acceder a la opción que permite encender y apagar al cortador.

Escogiendo el botón **“Menú Inicial”**, se regresa a la Pantalla Principal para escoger el resto de opciones que permite ejecutar un proceso de corte automático.

MODO DE TRABAJO AUTOMÁTICO.

Antes de escoger el Modo de trabajo Automático, es necesario realizar la conversión de los archivos generados en el programa CIMCO escogiendo el botón **“Archivos”** localizado en el Menú Principal.



La pantalla **“Manejo de Archivos”**, que se despliega da la opción de seleccionar el archivo que contiene la figura que se desea recortar generado por el programa CIMCO y guardado con extensión **“TXT”**. Al seleccionar el archivo (botón **“Seleccionar archivo CIMCO”**) *Estrella.txt*, la figura se despliega sobre la pantalla; en éste momento se debe presionar el botón **“Construir archivo.CBC”** para guardar el nombre del archivo con extensión CBC con el cual desea llamar a la figura.

Realizado el proceso anterior, hay que regresar al Menú Principal y ahora sí escoger el botón **“Automático”** que representa al modo de trabajo automático (Manejo Automático del Cortador).

Al igual que en el modo Manual, el usuario debe inicializar el equipo presionando el botón ubicado en la parte inferior izquierda de la pantalla (el cortador se ubica en el origen); inicializado el equipo hay que seleccionar el archivo con extensión CBC que se guardó anteriormente (opción Archivo del Menú Principal), con ello se inicia el proceso de corte, a través del botón **“Seleccionar Iniciar”**.

El corte del archivo seleccionado se ejecuta automáticamente, con lo cual se observa en la pantalla el avance porcentual del proceso de corte, así como los valores de los parámetros con los cuales se ejecuta el corte de las líneas que conforman el gráfico.

En caso el produzca algún error, el usuario puede interrumpir el proceso con ayuda del botón **"Interrumpir"** y si se desea proseguir el corte, se habilita inmediatamente el botón **"Reiniciar"**, con el cual se prosigue el corte desde el punto en el que se detuvo.

En ésta pantalla, el usuario puede seleccionar la posición sobre la cuál se desea realizar el corte, si se escoge la opción **"Primero"**, el equipo reconoce el origen como la posición 0, mientras que con la opción **"Segundo"**, el cortador se ubicará en la mitad del eje X, considerando a ésta posición como el nuevo origen.