

**MAQUINA HERRAMIENTA CON CONTROL NUMÉRICO POR COMPUTADOR
PARA LA PERFORACIÓN DE CIRCUITOS IMPRESOS**

SANDRA PATRICIA BUENDÍA JURADO

BORIS REINA ROMERO

**UNIVERSIDAD SURCOLOMBIANA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
NEIVA
2007**

**MAQUINA HERRAMIENTA CON CONTROL NUMÉRICO POR COMPUTADOR
PARA LA PERFORACIÓN DE CIRCUITOS IMPRESOS**

SANDRA PATRICIA BUENDÍA JURADO

BORIS REINA ROMERO

**Proyecto de Grado Desarrollado como Requisito
Para Optar el Título
De Ingeniero Electrónico**

**Director
DIEGO JIMENEZ
Ingeniero Eléctrico**

**UNIVERSIDAD SURCOLOMBIANA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
NEIVA
2007**

Nota De Aceptación:

Presidente del Jurado

Jurado

Jurado

Neiva, 13 de Junio 2007

A mis padres, María del Carmen Montealegre y Humberto Buendía Garzón. Por su dedicación con esmero a mi surgir, la tenacidad que me han impartido y la consagración que me han heredado en la búsqueda de conocimiento.

A mi hermana....soporte incondicional.

A Dios quien en su maravillosa gloria ha decidido sumar a mi vida un tesoro más.

Sandra Patricia Buendía Jurado.

A Dios...

Boris Reina Romero

AGRADECIMIENTOS

Jorge Cortés, Ing. Mecánico gerente general Mecanizados CNC, Bogotá D.E; por su colaboración en la elaboración del núcleo de la máquina herramienta y su asesoramiento respecto al funcionamiento de Tornos y Electroerosionadoras de Hilo CNC.

Mauricio Guzmán, Tornero del Taller Industrial Soldatomo del Huila "TISH". Por su asesoría, acompañamiento y fabricación de la base de la máquina herramienta.

Nelson Perdomo, Comerciante, gerente general de Prooficinas.com. Por su colaboración económica en el desarrollo de algunas etapas del proyecto.

CONTENIDO

	Pág.
INTRODUCCIÓN	20
1. MARCO REFERENCIAL	22
1.1. INTRODUCCIÓN AL CAD/CAM	22
1.2. INTRODUCCIÓN AL CONTROL NUMÉRICO COMPUTARIZADO	38
1.2.1. Sistema CNC	39
1.2.2. Máquinas Herramientas de Control Numérico	40
1.2.3. Componentes de un Sistema CNC	41
1.2.4. Programación de CN	42
2. DISEÑO Y DESCRIPCIÓN DEL MECANISMO	43
2.1. TORQUE Y EQUILIBRIO DE CUERPO RÍGIDO	44
2.2. TORQUE DE UNA FUERZA	44
2.3. SISTEMAS DE ENGRANAJE Y CORREA	45
2.3.1. Engranajes Dentados	45
2.3.2. Correa	46
2.3.3. Tornillo Sinfín	47
2.4. RODAMIENTOS	48
2.4.1. Clasificación de los Rodamientos	49
2.5. DESCRIPCIÓN GENERAL DE LA MHCNC	50
2.5.1. Dimensiones	52

2.5.2. Herramientas	52
2.5.3. Actuador para Corte	54
2.5.4. Núcleo	55
2.5.5. Coordenada Z	56
2.5.6. Coordenada X	61
2.5.7. Coordenada Y	64
2.5.8. Soporte de impreso y Base	65
2.5.9. Modelado	66
3. INTERFAZ ELECTRÓNICA	69
3.1. ARQUITECTURA DE LA INTERFAZ ELECTRÓNICA	69
3.1.1. Bloque 1: Microcontrolador / Control del Bus / Selección del Chip	69
3.1.2. Bloque 2: Almacenamiento	77
3.1.3. Bloque 3,4, y 5: Coordenadas	78
3.1.4. Bloque 5: Fuente de Poder y Control AC	83
3.2. PROGRAMA DE MICROCONTROLADOR	85
4. SISTEMAS DE CONTROL	89
4.1. ENCODER	89
4.1.1. Descripción	89
4.1.2. Teoría de Funcionamiento	90
4.2. PROCESADOR DE CONTROL DE MOVIMIENTO LM629	91
4.2.1. Características	91
4.2.2. Teoría de Operación	92
5. DESCRIPCIÓN INTERFAZ GRÁFICA	107

5.1. INTRODUCCIÓN	107
5.2. ARQUITECTURA DE LA INTERFAZ GRÁFICA	107
5.3. ESTRUCTURA DEL SOFTWARE	108
5.3.1. Captación, Acondicionamiento y Presentación	108
6. CONCLUSIONES Y RECOMENDACIONES	127
BIBLIOGRAFÍA	129
ANEXOS	130

LISTAS DE TABLAS

	Pág.
Tabla 1. Diámetros de Brocas Comerciales	53
Tabla 2. Código de Función	86

LISTAS DE CUADROS

	Pág.
Cuadro 1. Velocidad de baudios para el modo asíncrono	76
Cuadro 2. Especificaciones Del LM629	93

LISTAS DE FIGURAS

	Pág.
Figura1 Ventana de Diseño SolidWorks	23
Figura 2. Análisis COSMOSXpress	26
Figura 3. Trazado de antena sobre circuito Impreso de un transmisor de radio frecuencia	28
Figura 4.EAGLE: Easily Applicable Graphical Layout Editor	29
Figura 5. Miva LS Large Area Photo plotter	30
Figura 6. Taladro de Control Numérico	30
Figura 7. Ventana de diálogo del Procesador CAM	31
Figura 8. (a) Archivo de Informe para Estación de Taladrado (.dri). (b) Parte del Archivo de Texto con las coordenadas para las perforaciones	33
Figura 9. VI para Configuración del Puerto Serial	34
Figura 10. Panel Frontal - Interfaz de Usuario	35
Figura 11. Panel de Programación con Estructuras: For, While, Evento, y Caso)	37
Figura 12. Ejecución del Programa Paso a Paso: Visualización del Flujo de Datos	37
Figura 13: Esquema de un sistema CN (a) y CNC (b)	40
Figura 14. Componentes de un sistema CNC	41
Figura 15. Relación entre información geométrica y tecnológica para los pasos de mecanizado	42
Figura 16. Par De Engranajes Dentados	45
Figura 17. Sistema de transmisión por correa	46
Figura 18. Rueda helicoidal accionada por Tornillo Sin Fin.	47
Figura 19.Rodamientos de bolas SKF utilizados en la MHCNC	48
Figura 20. Rodamientos Radiales	50
Figura 21. Vista isométrica frontal y posterior de la MHCNC	51
Figura 27. Ubicación MHCNC	51
Figura 28.Dimensiones MHCNC	52
Figura 29. Herramientas: Brocas, collets y portaherramienta	52
Figura 30.Broca	53
Figura 31. vista inferior del MHCNC	54
Figura 32. Vista Isométrica del Núcleo MHCNC	55
Figura 33. Ensamble del Núcleo (SolidWorks)	55
Figura 34. Portabrocas DREMEL	56
Figura 35. Transmisión Movimiento Eje Z	58
Figura 36. Ensamble Cabeza MHCNC (SolidWorks)	60
Figura 37. Cabeza MHCNC	60

Figura 38. Ensamble Coordenada X (SolidWorks)	61
Figura 39. Ensamble del Sistema para la Transmisión del Movimiento en la Coordenada X.	62
Figura 40. Coordenada X	63
Figura 41. Ensamble Coordenada Y (SolidWorks)	64
Figura 42. Ensamble Base y Soporte de impreso de la MHCNC	65
Figura 43. Modelado del Sistema para la Coordenada X	67
Figura 44. Arquitectura de la Interfaz Electrónica	70
Figura 45. Diagrama esquemático del circuito de control	71
Figura 46. Diagrama de conexiones y tabla de función del CI 74HC245	73
Figura 47. Circuito de Control de Bus utilizando el PSP del uC	75
Figura 48. Diagrama esquemático del bloque de almacenamiento	77
Figura 49: izq. Diagrama de De conexiones del 74HC374.	77
Figura 50. Tarjeta Principal	78
Figura 51. Diagrama Esquemático del bloque correspondiente a una Coordenada.	79
Figura 52. Señal de cuadratura de salida del encoder y su tabla de decodificación fotografía cinta codificada, disco codificado con su Encoder óptico	79
Figura 53. Puente H (h-bridge) L298 y su tabla de verdad	81
Figura 54. Forma de ondas para la magnitud PWM. y Configuración en paralelo del L298 le permite manejar corrientes mas altas	82
Figura 55. Tarjetas de Control PID	82
Figura 56. Fuente de Poder	83
Figura 57. Esquemático del Control del Motor AC	84
Figura 58. Montaje Circuito Electrónico	85
Figura 54. diagrama de flujo del programa principal	87
Figura 59: izq. Diagramas de empaque para el modulo encoder y Fotografía que muestra al encoder HEDS9798 junto a dos tipos de discos y una correa codificada con resolución de 150 Lpi (líneas por pulgada)	89
Figura 60. izq. Diagrama de bloques del modulo encoder. Der. Forma de onda De Salida del encoder	90
Figura 61. Sistema de Control de Motor	92
Figura 63. Ilustra dos perfiles trapezoidales de velocidad típica: trapezoide simple, y en la parte inferior un ejemplo de la forma de la trayectoria, cuando se cambian Velocidad y posición en dos momentos diferentes durante el movimiento.	94
Figura 64. Diagrama de Bloques Simplificado del Filtro PID LM629	96
Figura 65. Sistema de control digital	98
Figura 66 Especificaciones de la respuesta transitoria.	99
Figura 68. Respuesta al paso con parámetros de filtro para la coordenada X $K_p=100, k_i=10, k_d=4000, i_l=1000$	109
Figura 68. Respuesta al paso con parámetros de filtro para la coordenada X $K_p=10, k_i=5, k_d=2000, i_l=1000$	103
Figura 68. Respuesta al paso con parámetros de filtro para la coordenada X	

Kp=100, ki=50 kd=4000,il=1000	104
Figura 68. Respuesta al paso con parámetros de filtro para la coordenada X	
Kp=100, ki=10 kd=4000,il=1000	105
Figura 68. Respuesta al paso con parámetros de filtro para la coordenada Y	
Kp=50, ki=10 kd=4000,il=1000	106
Figura 68. Respuesta al paso con parámetros de filtro para la coordenada Z	
Kp=40, ki=20 kd=8000,il=1000	106
Figura 70. Diagrama de Bloques de la Interfaz Gráfica	107
Figura 71. Información contenida en un archivo *.DRL	109
Figura 72. Apariencia y Contenido del Archivo DRI (Esquemático: hexapodo2)	110
Figura 73. Vista “parcial” del Archivo de Códigos Numéricos (Esquemático: hexapodo2)	111
Figura 74. Visualización Error por Ejecución Estructura Caso 0	112
Figura 75. Mensaje desplegado al no encontrarse el archivo de informe de la estación de taladro, archivo de Códigos Numéricos o archivo de informe de herramientas.	113
Figura 76. Apariencia de la ventana de Adquisición de los Archivos CAM	115
Figura 77. Diagrama de Bloques para la Adquisición de los Archivos CAM	116
Figura 78. Subprograma Conversión de Unidades Brocas	117
Figura 79. Subprograma: Ajuste de Diámetros Brocas	119
Figura 80. Subprograma: Separación de Instrucciones Numéricas por Brocas	120
Figura 81. SubVI's Manejo de Herramientas	120
Figura 82. Panel Frontal: Presentación de Herramientas y Perforaciones	122
Figura 83. Subprograma Zoom Mapa de Perforaciones y Visualización Punto Señalado	122
Figura 84. SubPrograma Funciones de la Máquina	126
Figura 85. Panel de Frontal: Funciones de la Máquina	126

LISTAS DE ANEXOS

Anexo A. Avance de las Brocas para Distintos Materiales	131
Anexo B. Características Principales del LM629	132
Anexo C. Formato Sieb&Meyer	134
Anexo D. Principales Característica del Microcontrolador	136
Anexo E. Descripción de la Memoria RAM 6116	137
Anexo F. Características Encoder Incremental HEDS9798	138
Anexo G. Características Driver L298HN	142
Anexo H. Características Motores DC	144
Anexo I. Comandos de Usuario LM629	145
Anexo J. Esquemáticos Circuito Electrónico	146

INTRODUCCIÓN

Actualmente existe un ambiente de grandes expectativas e incertidumbre debido a los rápidos cambios de la tecnología que no permiten asimilarla en forma adecuada de modo que es muy difícil sacar su mejor provecho. También surgen cambios rápidos en el orden económico y político los cuales en las sociedades de países en desarrollo, inhiben el surgimiento de soluciones autóctonas o propias para los problemas más fundamentales.

Entre todos estos cambios, uno de los de mayor influencia lo será sin duda el desarrollo de las nuevas políticas mundiales de mercados abiertos y globalización. Todo esto habla de una libre competencia y surge la necesidad de adecuar las industrias a fin de que puedan satisfacer el reto de los próximos años. Una opción o alternativa frente a esto es la reconversión de las industrias introduciendo el elemento de la automatización. Sin embargo, debe hacerse en la forma más adecuada, de modo que se pueda absorber la nueva tecnología gradualmente en un tiempo apropiado; todo esto sin olvidar los factores de rendimiento de la inversión y capacidad de producción.

Uno de los elementos importantes dentro de este resurgir de la automatización son las Máquinas de Herramientas de Control Numérico Computarizado, las cuales brindan ventajas adicionales ya que desde los orígenes del control numérico todos los esfuerzos se han encaminado a incrementar la productividad, precisión, rapidez y flexibilidad de las máquinas-herramienta. Su uso ha permitido la mecanización de piezas muy complejas, especialmente en la industria aeronáutica, que difícilmente se hubieran podido fabricar de forma manual.

Las máquinas herramienta de control numérico configuran una tecnología de fabricación que de la mano de la microelectrónica, la automática y la informática industrial ha experimentado en los últimos años un desarrollo acelerado y una plena incorporación a los procesos productivos, desplazando progresivamente a las máquinas convencionales, su capacidad de trabajo automático y de integración de los distintos equipos entre sí y con los sistemas de control, planificación y gestión de formación, hacen del control numérico (CN) la base de apoyo a unas tecnologías de fabricación: el **COM**.- fabricación flexible y el **CIM** fabricación integrado por computadora

La utilización de estos sistemas traen consigo beneficios que permiten mejorar, aumentar la productividad, reducir los costos de producción, tiempo de desarrollo

y un incremento de la flexibilidad en la adaptación de las máquinas-herramienta y células de producción, logrando así, que productos de alta calidad y tecnología sean asequibles a todos los sectores.

1. MARCO REFERENCIAL

1.1. INTRODUCCIÓN AL CAD/CAM

CAD/CAM, proceso en el cual se utilizan los ordenadores o computadoras para mejorar la fabricación, desarrollo y diseño de los productos. Éstos pueden fabricarse más rápido, con mayor precisión o a menor precio, con la aplicación adecuada de tecnología informática.

Los sistemas de Diseño Asistido por Ordenador (CAD, acrónimo de Computer Aided Design) pueden utilizarse para generar modelos con muchas, si no todas, de las características de un determinado producto. Estas características podrían ser el tamaño, el contorno y la forma de cada componente; almacenados como dibujos bi y tridimensionales. Una vez que estos datos dimensionales han sido introducidos y almacenados en el sistema informático, el diseñador puede manipularlos o modificar las ideas del diseño con mayor facilidad para avanzar en el desarrollo del producto.

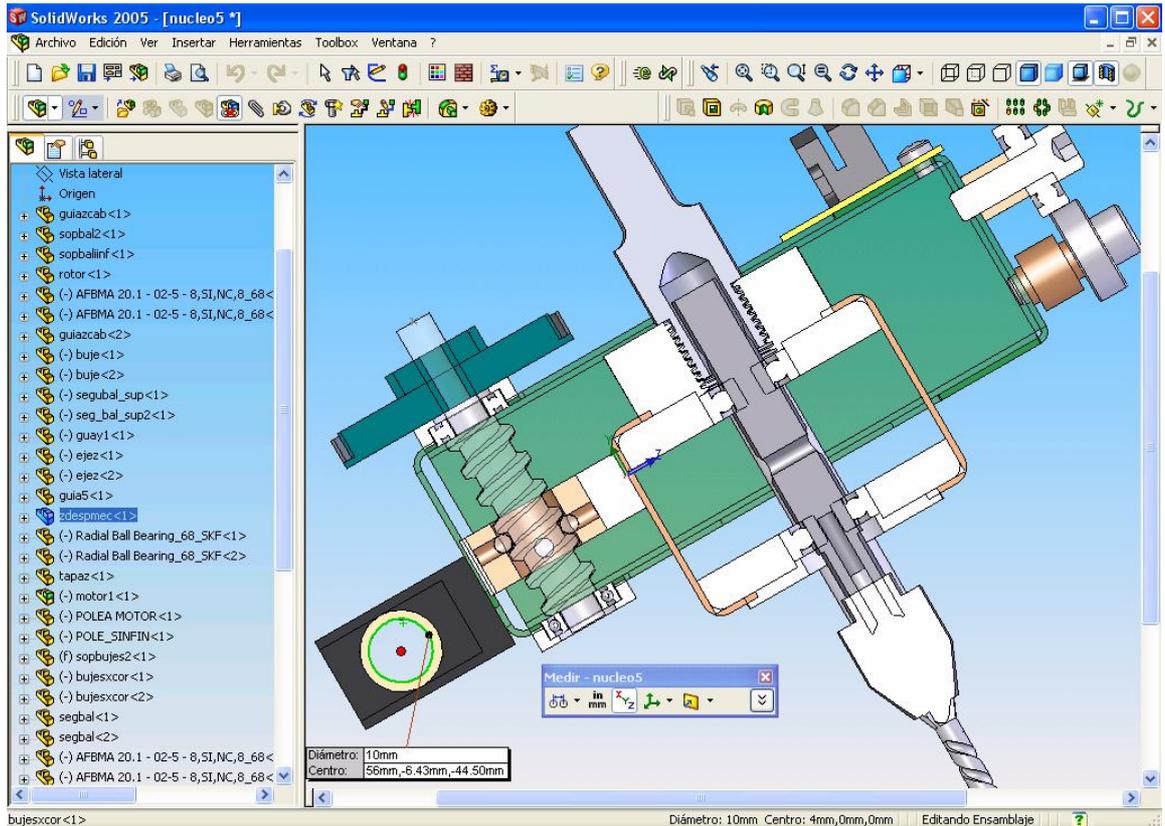
Los sistemas CAD también permiten simular el funcionamiento de un producto. Hacen posible verificar si un circuito electrónico propuesto funcionará tal y como está previsto, si un puente será capaz de soportar las cargas pronosticadas sin peligros e incluso si una salsa de tomate fluirá adecuadamente desde un envase de nuevo diseño.

Cuando los sistemas CAD se conectan a equipos de fabricación también controlados por ordenador conforman un sistema integrado CAD/CAM (CAM, acrónimo de Computer Aided Manufacturing).

Los equipos CAM se basan en una serie de códigos numéricos, almacenados en archivos informáticos, para controlar las tareas de fabricación. Este Control Numérico por Computadora (CNC) se obtiene describiendo las operaciones de la máquina en términos de los códigos especiales y de la geometría de formas de los componentes, creando archivos informáticos especializados o programas de piezas. La creación de estos programas de piezas es una tarea que, en gran medida, se realiza hoy día por software informático especial que crea el vínculo entre los sistemas CAD y CAM.

1.1.1. SOFTWARE CAD/CAM

Figura1. Ventana de Diseño SolidWorks



Solidworks³ SolidWorks es software de automatización de diseño mecánico que utiliza la interfaz de usuario gráfica creada por Microsoft Windows. Esta herramienta facilita a los diseñadores mecánicos esbozar rápidamente las ideas, experimentar con los rasgos, dimensiones, y; producir modelos y dibujos detallados. La Interfaz de Usuario de la aplicación SolidWorks contiene una amplia variedad de herramientas y posibilidades, que facilita la creación y edición de modelos de forma eficiente.

Intención de Diseño. La intención de diseño determina el modo en que desea que reaccione un modelo como resultado de cualquier cambio que se realice en el mismo. La intención de diseño es una cuestión principalmente de planificación. Decidir cómo se debe crear el modelo determina el modo en que los cambios le afectarán posteriormente. Cuanto más compenetrados estén, la implementación

³ SolidWorks 2005 Getting Started

del diseño y la intención del mismo, mayor será la integridad del modelo. Entre los factores que intervienen en el proceso de diseño, se encuentran los siguientes:

- **Necesidades Actuales.** Comprender el propósito del modelo para diseñarlo eficazmente.
- **Consideraciones Futuras.** Anticipar las posibles necesidades futuras para reducir al mínimo las tareas de volver a diseñar y de cambiar el modelo.

El proceso de diseño suele componerse de las siguientes fases:

- Identificación de las necesidades.
- Conceptualización del modelo en función de las necesidades identificadas.
- Desarrollo del modelo según los conceptos.
- Análisis de los resultados del desarrollo del modelo.
- Generación del prototipo del modelo.
- Construcción del modelo.
- Edición del modelo si fuera necesario.

Método de Diseño. En cuanto hayan sido identificados las necesidades y aislado los conceptos adecuados, es posible iniciar el desarrollo del modelo siguiendo las siguientes fases:

- **Croquis.** Se crean los croquis y se decide cómo se desea acotar, dónde se aplicarán relaciones de posición, etc. Un croquis es un perfil o sección transversal en 2D. Además de estos, se pueden crear croquis en 3D que contengan un eje Z, así como los ejes X y Y. Hay varias maneras de crear un croquis. Todos los croquis constan de los siguientes elementos: origen, planos, cotas.
- **Operaciones.** Se selecciona las operaciones adecuadas, y el orden para implementarlas, etc. Una vez finalizado el croquis, puede crearse un modelo en 3D empleando operaciones como: Extruir Superficie, Revolución de Superficie, etc.
- **Ensamblajes.** Pueden crearse varias piezas que se acoplan con el fin de elaborar ensamblajes. Los ensamblajes tienen muchas características, entre las que se destacan:
 - Pueden contener de 2 a mil componentes, que pueden ser piezas u otros ensambles, que se denominan subensamblajes.
 - Reproducen el movimiento entre piezas relacionadas dentro de sus posibilidades.

Las piezas se integran en un ensamblaje mediante Relaciones de Posición. Las relaciones de posición pueden ser de diversos tipos: coincidentes, colineales,

paralelas, coincidentes, concéntricas y de distancia entre otras. Con herramientas como Mover componente o Girar componente, puede verse cómo funcionan las piezas de un ensamblaje en un contexto en 3D.

Tareas De Ingeniería. El SolidWork contiene herramientas y funciones que ayuda a realizar tareas específicas del diseño del modelo. Dichas tareas consisten, entre otras, en crear variaciones de las piezas o importar archivos de sistemas de CAD heredados a los modelos de SolidWorks.

Tablas de Diseño. Las tablas de diseño permiten construir varias configuraciones de una pieza aplicando los valores de la tabla a las cotas de la pieza.

Importación y Exportación. Puede importarse y exportarse varios formatos de archivos desde y hacia el software SolidWorks. SolidWorks facilita una arquitectura abierta que permite compartir archivos con una amplia base de usuarios. El DWGeditor™, permite a los usuarios de AutoCAD® migrar a Solidworks a través de una interface familiar. Algunos de los formatos con los que trabaja SolidWorks son: Archivos IGES, DXF, STL, entre otros.

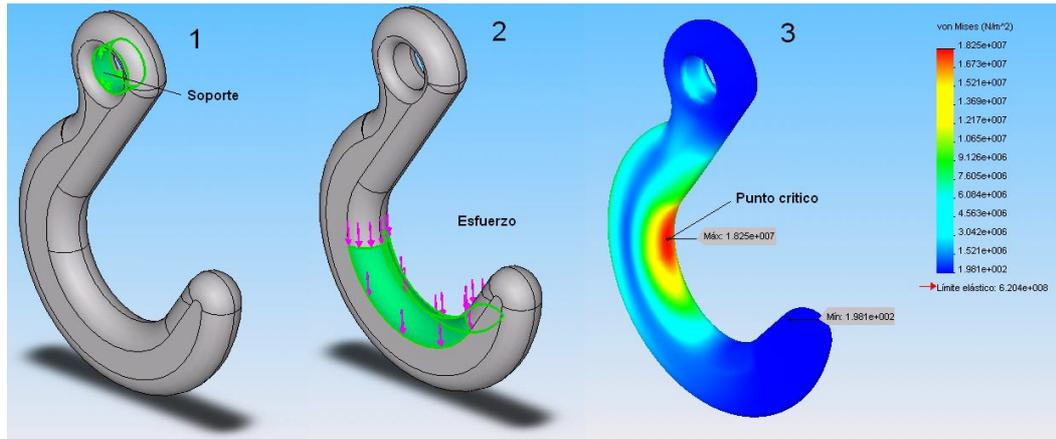
Cargar y Reemplazar. Se puede actualizar documentos compartidos para cargarlos en su última versión, incluido cualquier cambio efectuado. El software SolidWorks actualiza automáticamente los ensamblajes, una vez efectuado cualquier cambio en una pieza correspondiente al mismo.

Reemplazar: Se puede reemplazar un documento al que se hace referencia por otro documento que se encuentre en cualquier ubicación de una red. Cuando se reemplaza un componente, las relaciones posición que se utilizan en la pieza original se aplican a la pieza de sustitución siempre que sea posible. Para asegurarse de que se conserven las relaciones de posición, se debe cambiar el nombre de las caras y aristas correspondientes en la pieza de reemplazo para que coincidan con los nombres de las caras y aristas de la pieza original.

COSMOSXpress. COSMOSXpress ofrece una simulación de los ciclos de diseño para las piezas que se creen con COSMOSXpress, se puede reducir la necesidad de crear prototipos y de realizar pruebas sobre el terreno, prácticas que, de otro modo, resultan caras y lentas.

Por ejemplo, es posible analizar los efectos de un esfuerzo ejercido sobre una pieza. COSMOSXpress simula el ciclo de diseño y permite ver los resultados de aplicar un desplazamiento o una fuerza. Igualmente, indica las zonas críticas y el grado de seguridad de ciertos puntos de la pieza (figura 2).

Figura 2. Análisis COSMOSXpress



Ejemplos del tutorial de Solidworks 2005

En función de estos resultados, es posible reforzar puntos débiles y eliminar material de los puntos que presente una sobre carga.

Interfaz de Programación de Aplicaciones. La interfaz de programación de aplicaciones (API) de SolidWorks es una interfaz de programación OLE para el software SolidWorks. La API contiene miles de funciones que pueden activarse desde Visual Basic, VBA (Excel, Acces, etc.), C, C++ o archivos macros de SolidWorks. Estas funciones ofrecen acceso directo a las opciones funcionales de SolidWorks. Con la API, se puede personalizar la aplicación SolidWorks para reducir el tiempo dedicado al diseño. Puede llevar a cabo operaciones por lotes, hacer que los documentos de dibujo se rellenen automáticamente con las vistas o las cotas del modelo, crear sus propios PropertyManagers, etc.

eDrawings. Los eDrawings eliminan las barreras de comunicación a las que los diseñadores e ingenieros deben hacer frente día a día. Se pueden crear archivos eDrawings a partir de documentos de piezas, de ensamblaje o de dibujo. A menudo, el tamaño del archivo es demasiado grande para enviarlo por correo electrónico. Sin embargo, si el modelo es guardado como archivo eDrawings, se obtendrá una versión mucho más reducida del archivo. Los eDrawings pueden visualizarse con el eDrawings Viewer.

PhotoWorks. Es una aplicación de renderización que permite crear imágenes en realismo fotográfico directamente a partir de modelos de SolidWorks. Con PhotoWorks, se puede especificar las propiedades de la superficie del modelo como el color, la textura, la reflectancia y la transparencia. photoWorks cuenta con una biblioteca de texturas de superficies (metales, plásticos, etc.) y, además permite escanear un propio mapa de bits de superficies, como texturas, materiales, escenarios y logotipos.

SolidWorks Animador. Con la aplicación SolidWorks Animador, es posible animar y capturar los ensamblajes de SolidWorks en movimiento. solidWorks Animator genera archivos .avi que podrá reproducirse en cualquier sistema basado en Windows. Junto con el software PhotoWorks la aplicación SolidWorks Animator pueda producir animaciones en realismo fotográfico. Las animaciones hacen posible que los modelos puedan ser observados en situaciones de la vida real.

SolidWorks Explorer. SolidWorks Explorer es una herramienta de administración de archivos diseñada para facilitar tareas como el cambio de nombre, reemplazo o la copia de documentos de SolidWorks. SolidWorks Explorer permite:

- Visualizar las dependencias de documentos de dibujos, piezas y ensamblajes en una estructura árbol.
- Copiar, cambiar el nombre o reemplazar documentos de referencia. Tiene la opción de detectar y actualizar las referencias a los documentos.
- Visualizar datos y vistas preliminares, o datos de entrada, según la función que tenga activada.

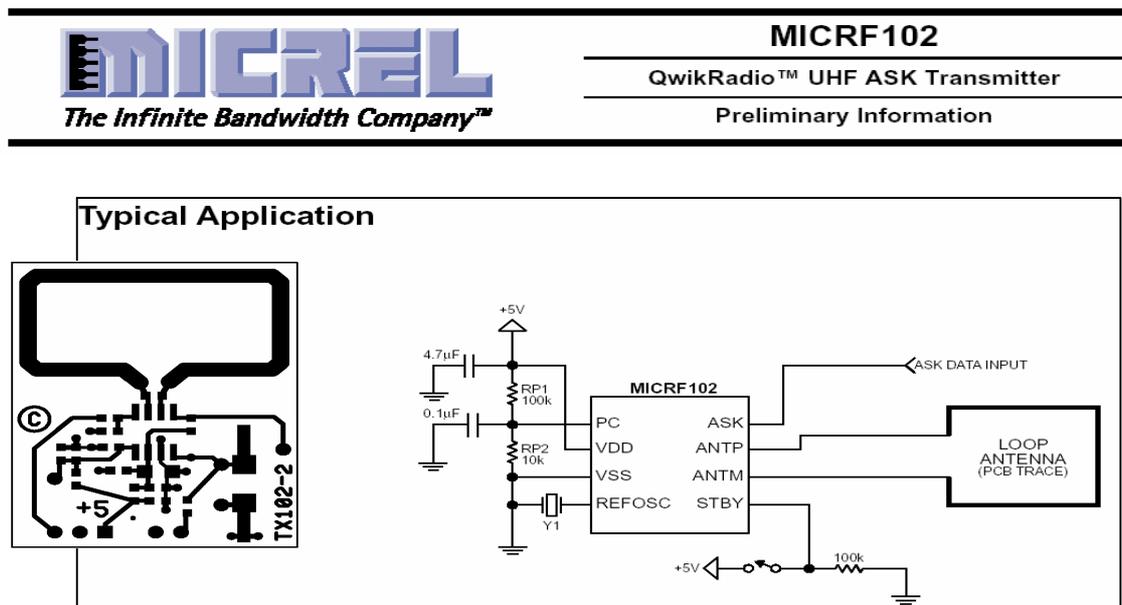
SolidWorks Toolbox. SolidWorks Toolbox contiene una biblioteca de piezas estándar que están integradas con el software SolidWorks, este admite varios estándares internacionales como ANSI, BSI, CISC, DIN, ISO y JIS. Además, SolidWorks Toolbox cuenta con diversas herramientas de ingeniería:

- Calculadora de vigas. Lleva a cabo cálculos sobre la desviación y el esfuerzo en secciones transversales de acero estructural.
- Calculadora de rodamientos. Lleva a cabo cálculos sobre los rodamientos con el fin de determinar los valores de capacidad nominal y de duración básica.
- Levas. Crea levas con trayectos de movimiento y tipos de seguidores completamente definidos. La leva puede ser circular o lineal y permite elegir entre 14 tipos de movimiento. También es posible definir el modo en que se corta el itinerario del seguidor, ya sea como corte ciego o como corte a través de toda la leva.
- Ranuras. Crea ranuras para aros tóricos y ranuras de anillo de retención para el modelo cilíndrico.
- Acero estructural. Lleva el croquis de sección transversal de una viga de acero estructural hasta una pieza. El croquis está completamente acotado para respetar los tamaños estándar del sector.

EAGLE 4.0 EAGLE es un potente editor de gráficos y esquemas para el diseño de placas circuitos con PC. Este software permite crear artes exactos y precisos de cualquier número de capas en pocos minutos, liberando al usuario de la tediosa

tarea de dibujar y enrutar manualmente las pistas y pads (por ensayo y error). En un circuito simple el circuito impreso puede diseñarse sencillamente siguiendo un diagrama esquemático y traduciendo las líneas a un patrón de trazos y *pads* de interconexión, que se transferirá directamente o mediante procesos foto-químicos, a una tarjeta virgen de cobre de una o dos caras.

Figura 3. Trazado de antena sobre circuito impreso de un transmisor de radio Frecuencia ASK



www.micrel.com MICRF102 QwikRadio™ UHF ASK Transmitter data sheet

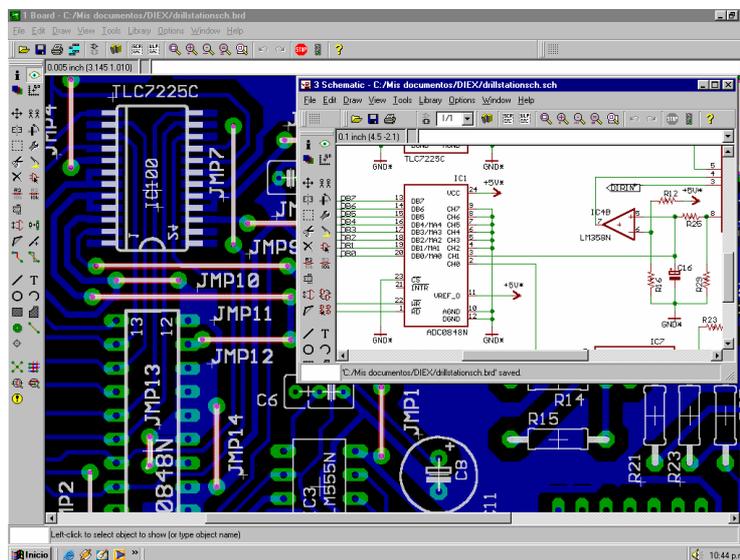
Los circuitos impresos han alcanzado un gran desarrollo industrial, al punto que se pueden encontrar con caminos conductores galvanizados en oro, de varias capas, de montaje superficial; pasando de ser un mecanismo de soporte para convertirse en algunas ocasiones, en parte integral del circuito electrónico, desempeñando funciones de algún componente como: condensadores y bobinas, ó emitiendo señales electromagnéticas, comportándose como una antena. Figura 3.

Manufactura. La manufactura de circuitos impresos se divide fundamentalmente en dos partes:

- Proceso de Diseño
- Proceso de Fabricación

Proceso de Diseño. El software EAGLE es la herramienta fundamental durante el desarrollo del proceso de diseño del circuito impreso. Este proceso consiste en generar una gráfica (Artes) de los caminos conductores de forma tal que se adecuen a las dimensiones de los dispositivos, proporcionando las conexiones eléctricas requeridas y cumpliendo con factores que garantizan el adecuado funcionamiento del circuito electrónico, como lo es tolerar las corrientes eléctricas que circulan por determinado camino conductor (figura 4).

Figura 4. EAGLE: Easily Applicable Graphical Layout Editor



Para diseñar una tarjeta de circuito impreso por computador es preciso seguir un proceso sistemático de tres pasos, con el fin de lograr que el proceso transcurra de una manera uniforme y ordenada. Estos pasos son:

- Dibujo y captura del esquema
- Colocación de los componentes
- Enrutamiento de las pistas

Proceso de Fabricación. Una vez realizado el proceso de diseño. El procesador CAM del software Eagle, permite combinar los archivos de estas dos escenas: dibujo y captura del esquema, colocación de los componentes y enrutamiento de las pistas, para producir un juego completo de archivos de salida que contiene la información necesaria para la manufactura del circuito impreso a través de diversos equipos como: fotoplotters, estaciones de taladro, etc. Bajo formatos de comandos: Gerber, Excellon, SM1000, entre otros.

Una fotográfica o fotoplotter es una máquina que dibuja, en forma exacta y precisa, líneas y formas de luz sobre un medio fotosensible (película). La fotográfica lee la información del archivo GERBER y la utiliza para crear una imagen virtual del arte, incluyendo los patrones individuales de cada capa, un patrón para el antisolder y un patrón para el Screen. Esta imagen se desarrolla fotográficamente produciendo un arte de alta calidad y gran precisión ($\pm 0.0002''$, típicamente) que se utiliza para producir la tarjeta final por medios fotoquímicos. (Figura 5)

Figura 5. Miva LS Large Area Photo plotter



El procesador CAM genera un archivo para una estación de taladro, en el cual se especifican varios datos como: coordenadas de las perforaciones, el número y diámetro de las herramientas (brocas), etc. Este archivo es leído y ejecutado a través de una máquina herramienta de Taladrado por control numérico (Figura 6) que deja como resultado una placa con perforaciones de diferentes diámetros para el montaje de cada componente.

Figura 6. Taladro de Control Numérico

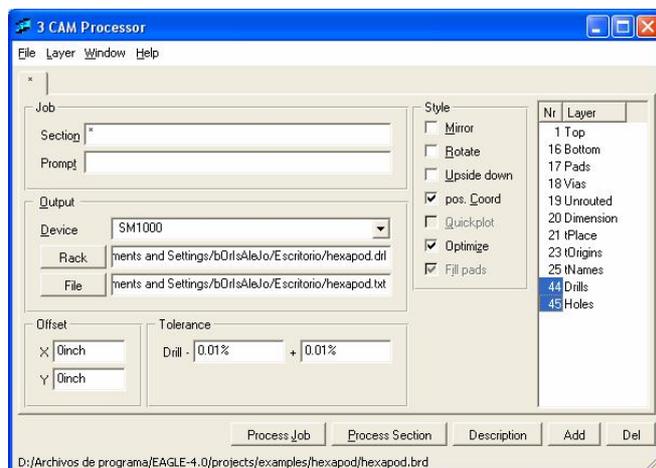


Procesador CAM. Para llevar a cabo el proceso de manufactura de un circuito impreso, se requiere la creación de un archivo que contiene los parámetros de las herramientas a utilizar:

- Creación del archivo de taladro. Una estación de taladro necesita saber qué diámetros del taladro se asignan a los códigos usados en el archivo de salida (archivo de Códigos Numéricos). Estas asignaciones se definen en un Archivo de Armario de Taladro. El archivo se genera con la ayuda de un “Programa de Idioma de Usuario” llamado DRILLCFG.ULP que se guarda en el directorio ULP del software Eagle.

Una vez realizado este procedimiento, es posible acceder a la ventana de diálogo del procesador CAM, en donde se definirán las características del archivo de control numérico a ejecutar por alguna máquina herramienta especializada.

Figura 7. Ventana de diálogo del Procesador CAM



- Archivo de Armario de Taladros “Rack”. Archivo DRL: “Drill Rack File” Archivo de Armario de Taladros presenta la información concerniente a las herramientas a utilizar en la fabricación del circuito impreso, junto con el diámetro de las mismas.

T01 0.50mm
T02 0.81mm
T03 0.91mm
T04 1.02mm
T05 1.12mm
T06 1.32mm
T07 3.30mm; T: herramienta y diámetro.

- Archivo. Genera un archivo de Códigos Numéricos que la máquina herramienta ejecuta en el proceso de manufactura. Para el desarrollo del proyecto, el archivo es guardado como documento de texto (figura 8).

Finalizado la programación del Procesador CAM, este genera un archivo de Informe de la Estación de Taladro “Drill Station Info File”, que contiene los datos seleccionados en la ventana de diálogo del Procesador CAM (figura 8):

- Parámetros Programados: tolerancia, capas, etc.
 - Información de las Perforaciones: tipos de datos, unidades.
 - Número y diámetro de las herramientas
 - Perforaciones totales, etc.
- El Dispositivo de salida. Define el tipo de archivo de salida que el Procesador CAM produce. Este depende del equipo seleccionado: Excellon, Gerber, SM1000, etc.
 - Tolerancia. Programa las tolerancias para las perforaciones. El Procesador CAM usa los diámetros del taladro dentro de las tolerancias, puesto que un taladro con el valor exacto no es disponible.
 - El desplazamiento. Establece la compensación necesaria en la dirección X y la dirección Y (en pulgadas o milímetros), para posicionar el origen de las coordenadas (Punto Cero) en un lugar a partir del dato suministrado.
 - Estilo
 - *Pos. Coord.* Elimina las coordenadas negativas y el dibujo es referenciado al origen del dispositivo de salida.
 - *Quickplot Plot.* Muestra sólo los contornos de objetos (sujeto a la disponibilidad en el dispositivo de salida seleccionado).
 - Capas. Selecciona la capa adecuada para la generación del archivo de control numérico, según el proceso de manufactura a desarrollar.

Figura 8. Archivo de Informe para Estación de Taladrado (.dri), y parte del Archivo de Texto con las coordenadas para las perforaciones.

```

Parameter settings:
Tolerance Drill + : 0.50 %
Tolerance Drill - : 0.50 %
Rotate           : no
Mirror           : no
Optimize         : yes
Auto fit        : yes
OffsetX         : 0.000
OffsetY         : 0.000
Layers          : Drills Holes

Drill File Info:
Data Mode       : Absolute
End of Block   : CR/LF
Units          : 1/100 mm

Drill Tolerances:
-- Requested -- -- Used Drill --
Size      used  Code  Size
32.000mil 164  T02  0.810mm
36.000mil  58  T03  0.910mm
40.000mil 102  T04  1.020mm
52.000mil  4   T06  1.320mm
44.000mil 10   T05  1.120mm
130.000mil  2   T07  3.300mm

Drills used:
Code  size      used
T01   0.500mm   41
T02   0.810mm  164
T03   0.910mm   58
T04   1.020mm  102
T05   1.120mm   10
T06   1.320mm    4
T07   3.300mm    2

Total number of drills: 381

```

```

%%1000
X1677Y2076T01
X1677Y2266
X3582Y2901
X3582Y3155
X4598Y3155
X4598Y2901
X5360Y3028
X5550Y3219
X5550Y3409
X6630Y2647
X6630Y2393
X6630Y2012
X8938Y7473T02
X8189Y7473
X7440Y7473
X6439Y7473
X11176Y6330T03
X11176Y5187
M30

```

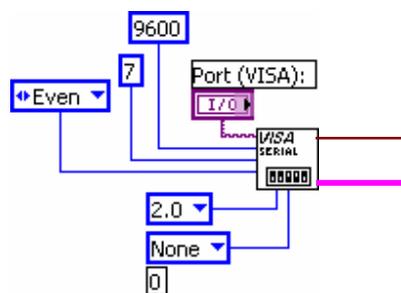
LabVIEW LabVIEW (Laboratory Virtual Instrument Engineering Workbench) es un lenguaje de programación gráfico para el diseño de sistemas de adquisición de datos, instrumentación y control. Labview permite diseñar interfaces de usuario mediante una consola interactiva basada en software. Es posible diseñar especificando un sistema funcional, un diagrama de bloques o una notación de diseño de ingeniería. Labview es a la vez compatible con herramientas de desarrollo similares y puede trabajar con programas de otra área de aplicación, como por ejemplo Matlab. Tiene la ventaja de permitir una fácil integración con hardware, específicamente con tarjetas de medición, adquisición y procesamiento de datos (incluyendo adquisición de imágenes).

Una de las principales características de Labview es su modularidad, es decir, la capacidad de utilizar bloques funcionales para la definición de la especificación.

Labview permite conectarse a otras aplicaciones mediante un intercambio de datos como Active X, librerías dinámicas, bases de datos, Excel y/o a protocolos de comunicación como DataSocket, TCP/IP, UDP, RS-232, entre otras.

Programación gráfica – Modularidad. Cuando se diseñan programas con Labview se está trabajando siempre bajo algo denominado VI, es decir, un instrumento virtual, se pueden crear VI a partir de especificaciones funcionales que el usuario diseña. Este VI puede utilizarse en cualquier otra aplicación como una subfunción dentro de un programa general (figura 9). Los VI's se caracterizan por ser un cuadrado con su respectivo símbolo relacionado con su funcionalidad, tener una interfaz con el usuario, tener entradas con su color de identificación de dato, tener una o varias salidas y por su puesto ser reutilizables.

Figura 9. VI para Configuración del Puerto Serial



Interfaz de usuario a partir de su código. En el ambiente de trabajo de Labview existen dos paneles, el panel frontal y el panel de programación ó diagrama de bloques; en el panel frontal se diseña la interfaz con el usuario (figura 10) y en el panel de programación se relacionan los elementos utilizados en la interfaz mediante operaciones que determinan en sí como funciona el programa o el sistema, exactamente es la parte donde se realizan las especificaciones funcionales.

En el panel de programación se puede diseñar de manera gráfica y como si fuera un diagrama de bloques el funcionamiento del sistema. La programación gráfica se basa en la realización de operaciones mediante la asignación de iconos que representen los datos numéricos e iconos que representan los procedimientos que deben realizar los (VI's), con estos iconos y mediante una conexión simple como lo es una línea recta, se enlazan para determinar una operación y/o una función (figura11).

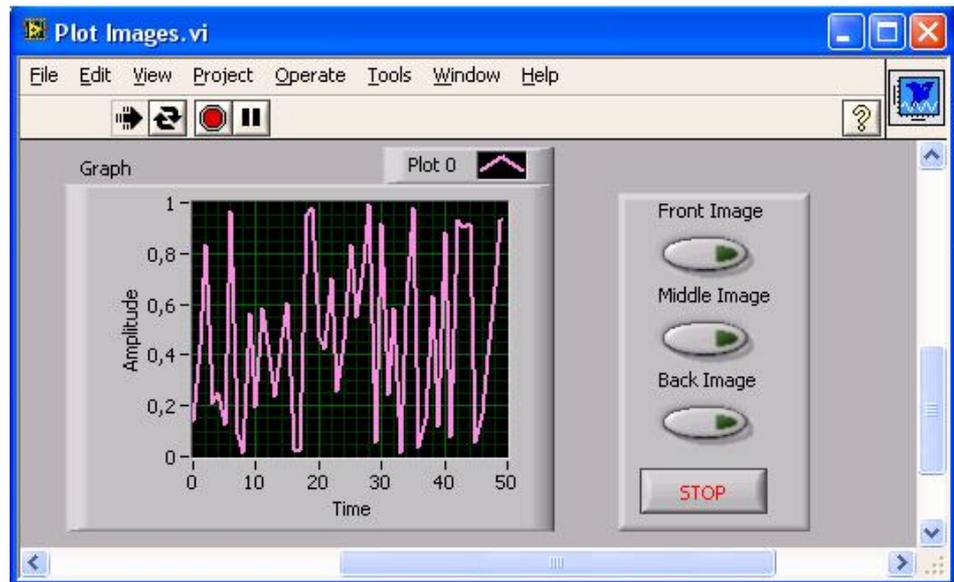
Al diseñar el programa de forma gráfica, se hace visible una programación orientada al flujo de datos, donde se tiene una interpretación de los datos también de forma gráfica, por ejemplo un dato booleano se caracteriza por ser una conexión verde, cada tipo de dato se identifica con un color diferente dentro de Labview; también es necesario tener en cuenta que cuando se realiza una conexión a un VI esta conexión se identifica por un tipo de dato específico, que debe coincidir con el tipo de dato de la entrada del VI (aunque esto no

necesariamente es cierto ya que puede haber varios tipos de datos conectados de VI a VI, además de que un arreglo de datos ``cluster`` puede albergar varios tipo de variables) permitiendo una concordancia en el flujo de datos; no siempre el tipo de dato de la entrada del VI es el mismo que el de la salida, pero sin embargo para la mayoría de los casos si se cumple.

El flujo de datos va de izquierda a derecha en el panel de programación y esta determinado por las operaciones o funciones que procesan los datos. Es fácil observar en el panel de programación como se computan los datos en cada parte del programa cuando se realiza una ejecución del programa paso a paso (Figura 12).

En Labview las variables se representan mediante una figura tanto en el panel frontal como en el panel de programación, de esta forma se puede observar su respuesta en la interfaz del usuario y en el flujo de datos del código del programa. Otros objetos como gráficas y accesos directos a páginas web cumplen estas mismas condiciones.

Figura 10. Panel Frontal - Interfaz de Usuario



Entorno LabVIEW. Controles. La programación G (gráfica) de Labview consta de un panel frontal y un panel de código como se menciono antes. En el panel frontal es donde se diseña la interfase de usuario y se ubican los controles e indicadores. En el panel de código se encuentran las funciones. Cada control que se utiliza en

la interfaz tiene una representación en el panel de código, igualmente; los indicadores necesarios para entregar la información procesada al usuario, tienen un icono que los identifica en el panel de código o de programación. Los controles pueden ser booleanos, numéricos, strings, un arreglo matricial de estos o una combinación de los anteriores; y los indicadores pueden ser como los controles pero pudiéndolos visualizar como tablas, gráficos en 2D o 3D, entre otros.

- *Panel de Programación o Código.* Los controles e indicadores que se colocaron previamente en el Panel Frontal, se materializan en el diagrama de bloques mediante los terminales.

El diagrama de bloques constituye el código fuente del VI. En el diagrama de bloques es donde se realiza la implementación del programa del VI para controlar o realizar cualquier proceso de las entradas y salidas que se crearon en el panel frontal.

El diagrama de bloques incluye funciones y estructuras integradas en las librerías que incorpora LabVIEW. En el lenguaje G las funciones y las estructuras son nodos elementales. Son análogas a los operadores o librerías de funciones de los lenguajes convencionales.

El diagrama de bloques se construye conectando los distintos objetos entre sí, como si de un circuito se tratara. Los cables unen terminales de entrada y salida con los objetos correspondientes, y por ellos fluyen los datos.

LabVIEW posee una extensa biblioteca de funciones, entre ellas, aritméticas, comparaciones, conversiones, funciones de entrada/salida, de análisis, etc.

- *Las estructuras*, similares a las declaraciones causales y a los bucles en lenguajes convencionales, ejecutan el código que contienen de forma condicional o repetitiva (bucle *for*, *while*, *case*,...). Los cables son las trayectorias que siguen los datos desde su origen hasta su destino, ya sea una función, una estructura, un terminal, etc. Cada cable tiene un color o un estilo diferente, lo que diferencia unos tipos de datos de otros.

Figura 11. Panel de Programación con Estructuras: For, While, Evento, y Caso)

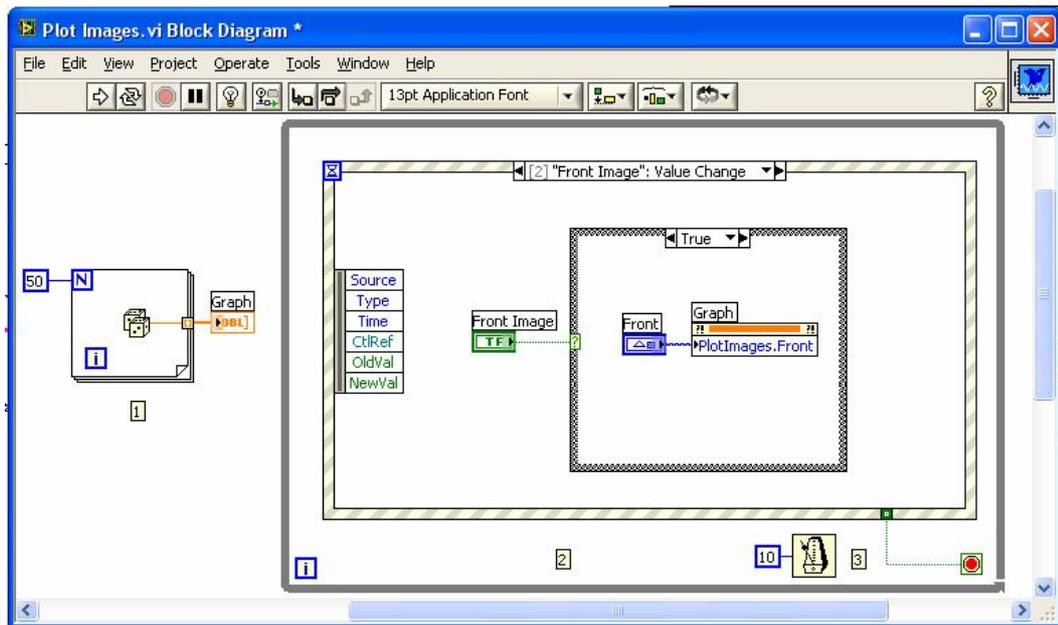
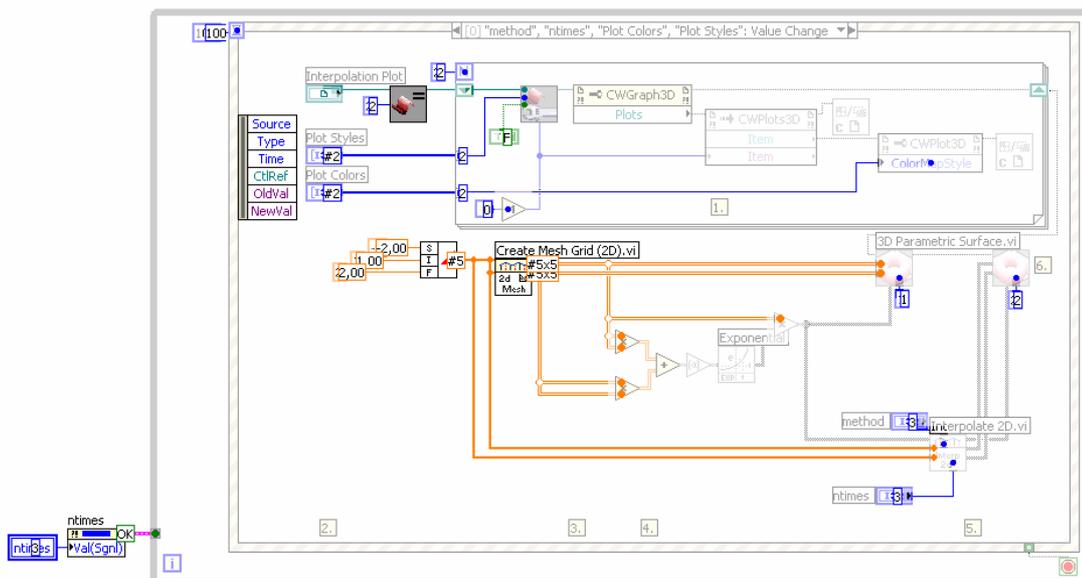


Figura 12. Ejecución del Programa Paso a Paso: Visualización del Flujo de Datos.



1.2. INTRODUCCIÓN AL CONTROL NUMÉRICO COMPUTARIZADO⁴

El CNC tuvo su origen a principios de los años cincuenta en el Instituto de Tecnología de Massachusetts (MIT), en donde se automatizó por primera vez una gran fresadora. En esta época las computadoras estaban en sus inicios y eran tan grandes que el espacio ocupado por la computadora era mayor que el de la máquina. Hoy día las computadoras son cada vez más pequeñas y económicas, con lo que el uso del CNC se ha extendido a todo tipo de maquinaria: tornos, rectificadoras, electroerosionadoras, máquinas de coser, etc.

En una máquina CNC, a diferencia de una máquina convencional o manual, una computadora controla la posición y velocidad de los motores que accionan los ejes de la máquina. Gracias a esto, puede hacer movimientos que no se pueden lograr manualmente como círculos, líneas diagonales y figuras complejas tridimensionales. Las máquinas CNC son capaces de mover la herramienta al mismo tiempo en los tres ejes para ejecutar trayectorias tridimensionales como las que se requieren para el maquinado de complejos moldes y troqueles.

En una máquina CNC una computadora controla el movimiento de la mesa, el carro y el husillo. Una vez programada la máquina, ésta ejecuta todas las operaciones por sí sola, sin necesidad de que el operador esté manejándola. Esto permite aprovechar mejor el tiempo del personal para que sea más productivo.

El término "control numérico" se debe a que las órdenes dadas a la máquina son indicadas mediante códigos numéricos. Por ejemplo, para indicarle a la máquina que mueva la herramienta describiendo un cuadrado de 10 mm por lado se le darían los siguientes códigos:

```
G90G71  
G00X0.0Y0.0  
G01X10.0  
G01Y10.0  
G01X0.0  
G01 Y0.0
```

Un conjunto de órdenes que siguen una secuencia lógica constituyen un programa de maquinado. Dándole las órdenes o instrucciones adecuadas a la máquina, ésta es capaz de maquinar una simple ranura, una cavidad irregular, la cara de una persona en altorrelieve o bajorrelieve, un grabado artístico un molde de inyección de una cuchara o una botella... lo que se quiera.

⁴ <http://www.cursos.femz.es/CNC-Basico>

Al principio hacer un programa de maquinado era muy difícil y tedioso, pues había que planear e indicarle manualmente a la máquina cada uno de los movimientos que tenía que hacer. Era un proceso que podía durar horas, días, semanas. Aún así era un ahorro de tiempo comparado con los métodos convencionales. Actualmente muchas de las máquinas modernas trabajan con lo que se conoce como "lenguaje conversacional" en el que el programador escoge la operación que desea y la máquina le pregunta los datos que se requieren. Cada instrucción de este lenguaje conversacional puede representar decenas de códigos numéricos. Por ejemplo, el maquinado de una cavidad completa se puede hacer con una sola instrucción que especifica el largo, alto, profundidad, posición, radios de las esquinas, etc. Algunos controles incluso cuentan con graficación en pantalla y funciones de ayuda geométrica. Todo esto hace la programación mucho más rápida y sencilla.

También se emplean sistemas CAD/CAM que generan el programa de maquinado de forma automática. En el sistema CAD (diseño asistido por computadora) la pieza que se desea maquinar se diseña en la computadora con herramientas de dibujo y modelado sólido. Posteriormente el sistema CAM (manufactura asistida por computadora) toma la información del diseño y genera la ruta de corte que tiene que seguir la herramienta para fabricar la pieza deseada; a partir de esta ruta de corte se crea automáticamente el programa de maquinado, el cual puede ser introducido a la máquina mediante un disco o enviado electrónicamente.

Hoy día los equipos CNC con la ayuda de los lenguajes conversacionales y los sistemas CAD/CAM, permiten a las empresas producir con mucha mayor rapidez y calidad sin necesidad de tener personal altamente especializado.

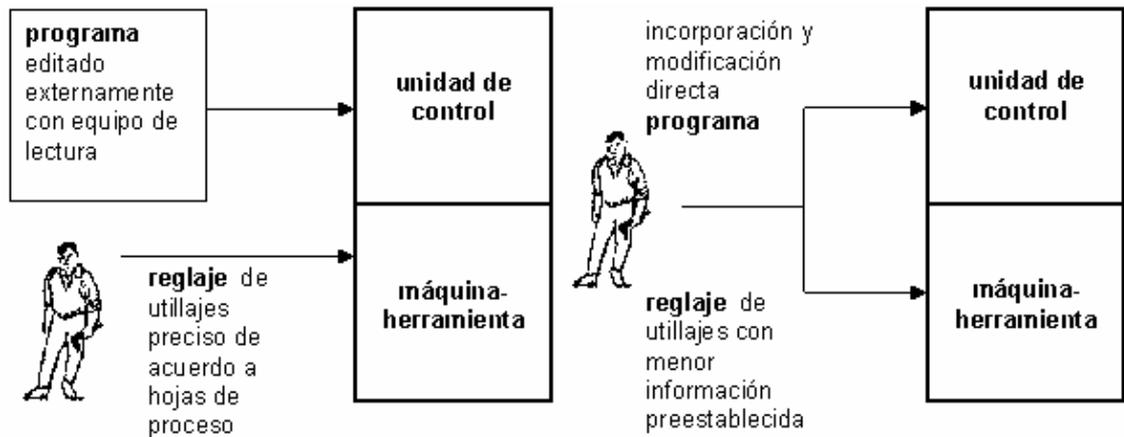
1.2.1. Sistemas CNC: (controlados numéricamente por computador)

- Presentan un computador como Unidad de Control que permite al operador comenzar (o terminar) el programa y además realizar modificaciones (editar) sobre el mismo a pie de máquina manipulando los datos con periféricos de entrada y salida.
- Las dimensiones de herramientas y utillajes se definen durante el reglaje o inicialización de las mismas, de forma independiente al programa. Estos datos se incorporan automáticamente a la programación durante la ejecución para que sean llevadas a cabo las correcciones pertinentes. Por esta razón el operador puede editar los programas con menos información de partida, limitándose a seleccionar las herramientas o utillajes en esa fase.

No existen diferencias entre CN y CNC con relación a:

- Lenguaje de programación
- Tecnología de la máquina-herramienta

Figura 13: Esquema de un sistema CN y CNC



1.2.2. Máquinas Herramientas de Control Numérico MHCNC: las Máquinas Herramientas de Control Numérico (MHCNC), constituyen una modalidad de automatización flexible mas utilizada; son maquinas herramientas programadas para fabricar lotes de pequeño y medio tamaño de piezas de formas complicadas; los programas de software sustituyen a los especialistas que controlaban convencionalmente los cambios de las maquinas y constituciones que incluye las tareas y sus velocidades. Estas máquinas pueden encontrarse en forma asilada, en cuyo caso se habla de un modulo, o bien interconectadas entre si por medio de algún tipo de mecanismo automático para la carga y descarga del trabajo en curso, en cuyo caso se hablaría de una célula de fabricación. En ocasiones las maquinas están dispuestas en forma semicircular para que un robot pueda encargarse de manejar los materiales, mientras que en otros la configuración es lineal. Cuando una maquina de control numérico actúa de forma independiente, necesita contar con la presencia de un operario, quien se ocupa de la carga y descarga de las piezas a procesar, los programas y las herramientas.

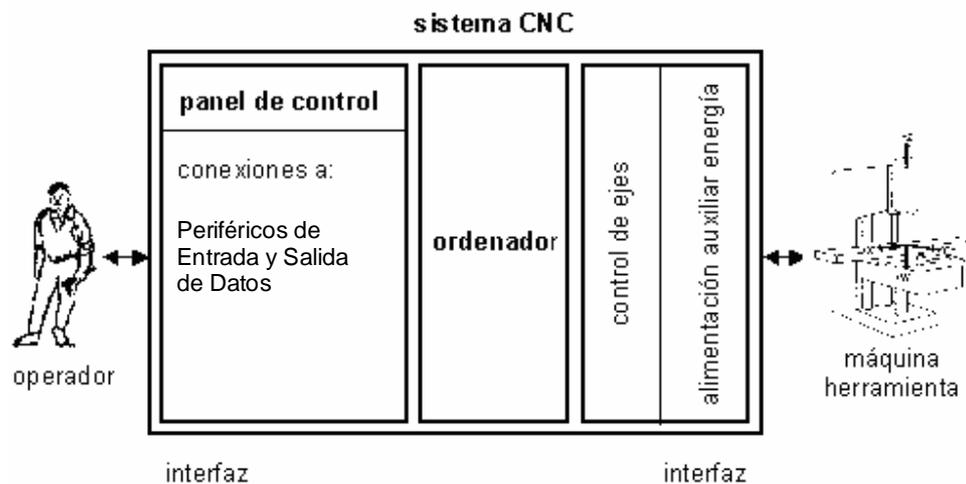
Algunas máquinas CNC incluyen “cartucheras” rotatorias con diferentes herramientas. El programa de ordenador puede seleccionar la herramienta a utilizar, de este modo, una máquina puede encargarse de realizar distintas operaciones que antes habían de hacerse en varias máquinas. No solo se reduce

el tiempo de lanzamiento, sino que también se simplifica el flujo de items en curso por el taller. En otros casos, frente a las máquinas se ubica un carrusel de herramientas, materiales, etc. y aquellas, sin necesidad de intervención humana, seleccionan con un “brazo” el instrumento o material que necesitan para desarrollar una determinada tarea.

Se cree que, en un futuro, las máquinas de Control Numérico harán el trabajo de precisión, mientras que los robots se limitaran a la carga, descarga y ensamblaje. En los casos de producción de gran volumen, la automatización rígida, más sencilla y barata, sería suficiente porque, aunque puede haber excepciones, las máquinas CNC y los robots son lentos. Para determinar la conveniencia de estas máquinas en términos de costo habrá que considerar la mano de obra, la disponibilidad de operarios especializados, tipo y grado de precisión requerida, fiabilidad de las máquinas, etc. Algunas empresas que producen una gama de productos estrecha se han dirigido, no obstante, a las máquinas CNC porque, aunque el costo de la programación sea alto, una vez hecha esta, puede ser utilizada posteriormente sin necesidad de volver a programar.

1.2.3. Componentes de un sistema CNC: un sistema CNC está constituido por numerosos componentes.

Figura 14. Componentes de un sistema CNC



El corazón de un sistema CNC es un computador que se encarga de realizar todos los cálculos necesarios y de las conexiones lógicas. Tendiendo a que el sistema CNC sea el puente de unión entre el operador y la máquina-herramienta se necesitan dos interfaces (traductores):

- El interfaz del operador formado por el panel de control y varios a él conectados relacionados generalmente con dispositivos de periféricos almacenamiento o impresión de la información.
- El interfaz de control de la máquina-herramienta que esta subdividido en múltiples conexiones de control y que afectan los actuadores de ejes, etc. hasta llegar al sistema auxiliar de alimentación de energía.

1.2.4. Programación De CN: a continuación se describen los distintos pasos de que constan las operaciones de mecanizado mediante máquinas de CN, sin considerar un lenguaje de programación. Para seguir los pasos necesarios, la máquina CN requiere principalmente, información tecnológica y geométrica.

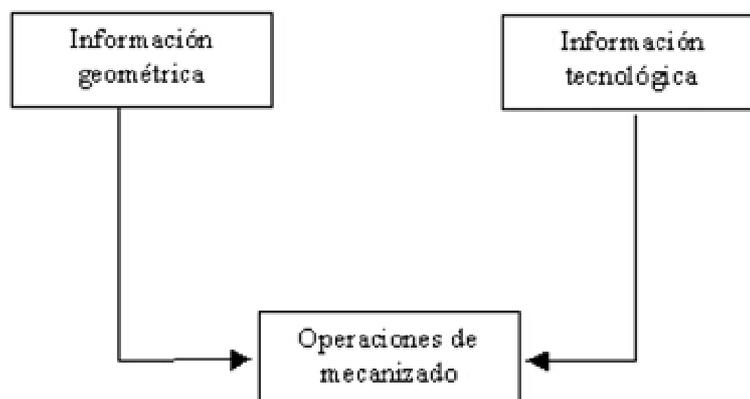
La información geométrica consiste en:

- Datos dimensionales de la posición final.
- Descripción de los movimientos de la herramienta y
- Posicionamiento en el área de trabajo del cero y puntos de referencia necesarios.

La información tecnológica consiste en:

- Datos necesarios sobre la herramienta a usar,
- Datos de corte (velocidad, avance, etc.) y
- Funciones de la máquina a ser controladas (cambio de herramienta, etc.).

Figura 15. Relación Entre Información Geométrica Y Tecnológica Para Los Pasos De Mecanizado



2. DISEÑO Y DESCRIPCIÓN DEL MECANISMO

“Perforar circuitos impresos” bajo esta premisa se basa el diseño mecánico del prototipo MHCNC. Limitado por sensores, actuadores, mecanismos y piezas que se pueden obtener en el comercio y están condicionados por su costo. Se busca entonces, un equilibrio entre el factor económico y el alcance tecnológico del equipo, sin embargo este último debe tener un índice mínimo aceptable.

En el proceso de selección de componentes, como en el desarrollo de diseño del equipo, muchos mecanismos, piezas, y actuadores fueron sobredimensionados en resistencia y funcionalidad, para que en la mayoría de los casos, puedan ser ajustados a su posición o potencia adecuada, a través de procedimientos electrónicos o mecánicos.

Se sobredimensiona para tener la capacidad de adaptar el mecanismo a otros procesos (perforar otro tipo de materiales, contornear con fresa de corte transversal etc.). El diseño parte de un valor que se encuentra por encima del mínimo necesario para el desarrollo del propósito inicial del mecanismo, (perforar circuitos impresos). Este valor mínimo es obtenido por referencias externas, como especificaciones técnicas y en los casos donde la información esta ausente el parámetro se obtiene de forma experimental.

Limitar el diseño a los parámetros necesarios en un equipo es solo posible y obligatorio cuando se desea masificar el producto, y esto no solo se logra en la optimización con un solo prototipo, si no que puede darse el caso que al final de muchos prototipos, el producto tenga un resultado inaceptable.

Los parámetros de peso y volumen en el momento de comenzar una línea de producción son definitivos, ya que esto implica costos en materiales, maquinaria, embalaje y transporte. En la mayoría de las ocasiones el proceso de reducir peso y volumen, concluye en la reducción de la resistencia de los componentes a las fuerzas máximas necesarias para el propósito para el cual fue diseñado el equipo. En los casos donde es imperativo reducir el peso o volumen manteniendo la resistencia, se obliga a utilizar materiales especiales que elevan el costo de fabricación.

El hecho de sobredimensionar da como consecuencia el aumento de los costos, por tanto, en el diseño del prototipo MHCNC las características de resistencia, potencia, durabilidad entre otros, son aumentadas a un nivel que se encuentra condicionado, por los materiales y componentes.

2.1. TORQUE Y EQUILIBRIO DE CUERPO RÍGIDO⁵

En general un cuerpo puede tener tres tipos distintos de movimiento simultáneamente, de traslación a lo largo de una trayectoria, de rotación mientras se está trasladando, en este caso la rotación puede ser sobre un eje que pase por el cuerpo, y si a la vez este eje esta girando en torno a un eje vertical, a la rotación del eje del cuerpo rotante se le llama movimiento de precesión (por ejemplo un trompo), y de vibración de cada parte del cuerpo mientras se traslada y gira. Por lo tanto el estudio del movimiento puede ser en general muy complejo, por esta razón se estudia cada movimiento en forma independiente.

Cuando un cuerpo está en rotación, cada punto tiene un movimiento distinto de otro punto del mismo cuerpo, aunque como un todo se esté moviendo de manera similar, por lo que ya no se puede representar por una partícula. Pero se puede representar como un objeto extendido formado por un gran número de partículas, cada una con su propia velocidad y aceleración. Al tratar la rotación del cuerpo, el análisis se simplifica si se considera como un objeto rígido y se debe tener en cuenta las dimensiones del cuerpo.

2.2. TORQUE DE UNA FUERZA

Cuando se aplica una fuerza en algún punto de un cuerpo rígido, el cuerpo tiende a realizar un movimiento de rotación en torno a algún eje. La propiedad de la fuerza para hacer girar al cuerpo se mide con una magnitud física que llamada **torque o momento** de la fuerza. Se prefiere usar el nombre torque y no momento, porque este último se emplea para referirnos al momento lineal, al momento angular o al momento de inercia, que son todas magnitudes físicas diferentes para las cuales se usa el mismo término.

Se define el **torque** τ de una fuerza F que actúa sobre algún punto del cuerpo rígido, en una posición r respecto a un eje de rotación, al producto vectorial entre la posición r (distancia perpendicular entre su línea de aplicación y el eje de rotación, también conocida como palanca) y la fuerza aplicada F , por la siguiente expresión: $\tau = F \times r$ (Producto Vectorial)⁶

⁵ <http://130.64.87.32/ldaps/htdocs/Spanish/Physics/torque.html>

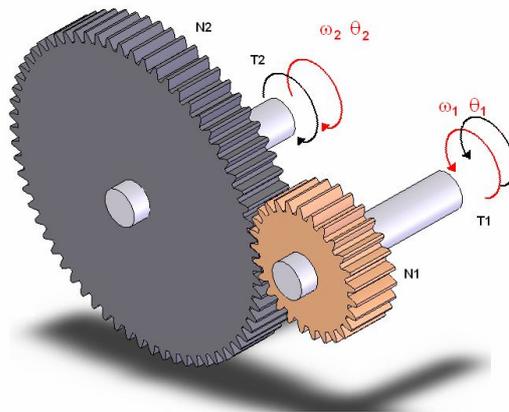
⁶ El producto vectorial de dos vectores A y B, es un vector perpendicular al plano formado por los dos vectores y cuya magnitud es igual al producto de sus magnitudes multiplicado por el seno del ángulo que forman entre sí los dos vectores y cuyo sentido se determina por la regla de la mano derecha (con el dedo índice se indica la dirección del primer vector y con el dedo del corazón la dirección del segundo vector, el pulgar da la dirección del producto vectorial).

El torque es una magnitud vectorial, si α es el ángulo entre r y F , su valor numérico, por definición del producto vectorial, es:

$$\tau = r(F \text{sen } \alpha)$$

2.3. SISTEMAS DE ENGRANAJE Y CORREAS

Figura 16. Par De Engranajes Dentados



2.3.1. Engranajes Dentados Los sistemas de reducción de engranaje se utilizan normalmente con motores, en un intento de optimizar la potencia que se dispone a velocidades altas. Aunque existen muchos tipos de configuraciones diferentes de engranajes, las relaciones fundamentales se describen utilizando un par de ruedas dentadas tal como se ilustra la figura 16. Como las fuerzas aplicadas y de carga sobre los dientes del engranaje que están en contacto son iguales, la magnitud de los pares respectivos debe estar en proporción directa al radio. La relación de los radios es, por supuesto, igual a la relación de las circunferencias. Por lo tanto, la relación del par aplicado al par transmitido es igual a la relación del número de dientes del engranaje tales que:

$$\frac{T_2}{T_1} = \frac{N_2}{N_1}$$

Donde en N es el número de dientes. Con la rotación de los engranajes, las distancias recorridas a lo largo de las circunferencias deben ser iguales y esta igualdad se puede plantear en términos del producto del cambio en ángulo y el radio tal que $\theta_1 R_1 = \theta_2 R_2$. Derivando con respecto al tiempo se obtiene $\omega_1 R_1 = \omega_2 R_2$. Para evitar el signo menos, la elección del movimiento que se supone positiva se

invierte para el segundo engranaje. Si la relación de los radios se sustituye por las relaciones de los dientes, entonces:

$$\frac{\theta_2}{\theta_1} = \frac{\omega_2}{\omega_1} = \frac{N_1}{N_2}$$

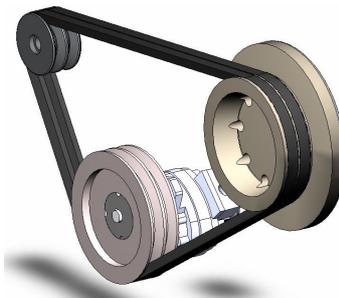
Así, el par cambia en proporción directa la relación de dientes y la velocidad angular en proporción inversa a dicha relación.

La reducción de la velocidad implica automáticamente un fenómeno muy favorable: el aumento de la fuerza o torque en el eje que recibe el movimiento. Esto corresponde a que la potencia o energía original del motor se debe conservar y como esta es directamente proporcional a la velocidad de rotación (ω) y al torque (τ); cuando la primera disminuye, el segundo debe aumentar o viceversa.

$$P_m = \tau \times \omega$$

Donde **Pm** es la potencia mecánica. En la práctica hay una pérdida muy pequeña de potencia debido a imperfecciones físicas en la forma y construcción de los piñones, sin embargo es tan pequeña que en la mayoría de los casos se puede despreciar.

Figura 17. Sistema de transmisión por correa



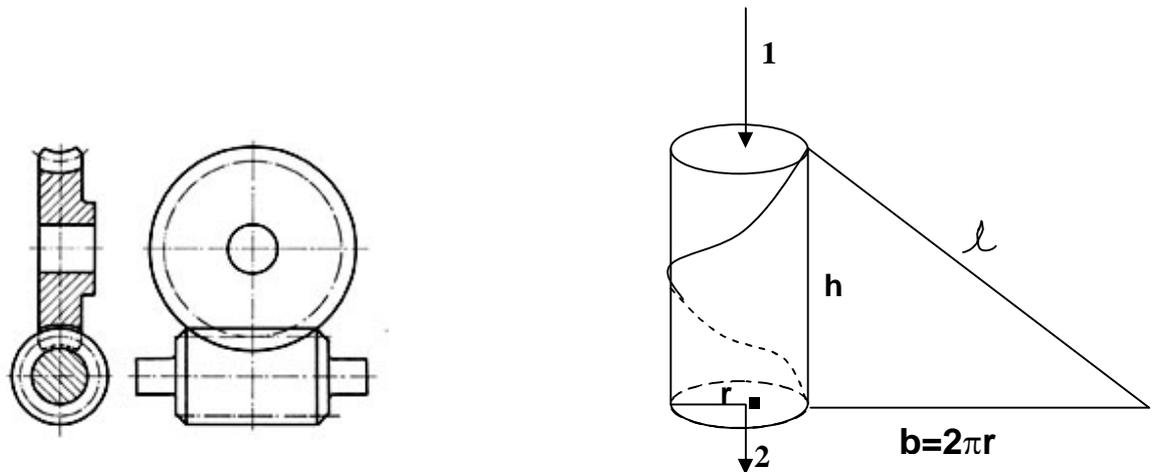
2.3.2. Correas Las ruedas conectadas por poleas o cadenas funcionan del mismo modo que los engranajes. La única diferencia radica en la dirección en que estas giran. Por ejemplo, una rueda con un diámetro dos (2) veces mayor que otra rueda. La haría girar con una fuerza dos (2) veces mayor a la mitad de velocidad, pero en la misma dirección.

2.3.3. Tornillo sin fin Este mecanismo se compone de un tornillo cilíndrico o hiperbólico y de una rueda (corona) de diente helicoidal cilíndrica o acanalada. Es muy eficiente como reductor de velocidad, dado que una vuelta del tornillo provoca un pequeño giro de la corona. Es un mecanismo que tiene muchas pérdidas por roce entre dientes, esto obliga a utilizar metales de bajo coeficiente de roce y una lubricación abundante, se suele fabricar el tornillo (gusano) de acero y la corona de bronce.

En la figura 18 se aprecia una gata de tornillo accionada por un mecanismo tipo tornillo sin fin y rueda helicoidal.

Es posible afirmar que el tornillo sin fin es una modificación del plano inclinado⁷, ya que consiste en que el plano inclinado envuelva un cilindro. Si se recorta un triángulo rectángulo de papel y se envuelve en un cilindro de radio r , de forma tal que $b=2\pi r$, la hipotenusa dibujará sobre el cilindro una "Hélice" ó "Rosca", de tal manera que h será el "Paso de Rosca".

Figura 18: Rueda helicoidal accionada por tornillo sin fin, y la representación geométrica del tornillo sin fin.



<http://www2.ing.puc.cl/Tornillo sin fin y rueda helicoidal.htm>

El trabajo aplicado al extremo superior del tornillo (durante una vuelta) es igual al trabajo desarrollado por su extremo inferior (durante una vuelta).

⁷ Plano inclinado, es el que forma un ángulo α con el horizontal. Mediante el una fuerza puede subir un cuerpo con ventaja mecánica, es decir, ejerciendo una fuerza menor que el peso del cuerpo

Puesto que al dar una vuelta al tornillo éste avanza longitudinalmente un “paso”:

$$P * 2\pi r = Q * h$$

$$Q = P * \frac{2\pi r}{h}$$

2.4. RODAMIENTOS⁸

Los rodamientos son piezas de acero aleado con cromo, manganeso y molibdeno, para facilitar la ejecución de rigurosos tratamientos térmicos y obtener piezas de gran resistencia al desgaste y a la fatiga. En la selección de los materiales, deben tomarse en consideración las temperaturas de operación y una adecuada resistencia a la corrosión.

El material para las jaulas ha evolucionado en forma importante actualmente se utilizan aceros, metales de bajo roce y poliamida.

Figura 19. Fotografía de rodamientos de bolas SKF utilizados en la MHCNC



⁸ <http://www2.ing.puc.cl/~icm2312/apuntes/roda/roda1.html>

Otra característica de los rodamientos es la exactitud de sus dimensiones cada parte debe tener tolerancias muy estrechas para un satisfactorio funcionamiento del conjunto.

Existen rodamientos de muy variados tipos para adecuarse a las diversas aplicaciones, es muy importante escoger el rodamiento preciso, tomando la decisión en base a criterios tales como: costo, facilidad de montaje, vida útil, dimensiones generales, simpleza del conjunto, disponibilidad de repuestos y tipo de lubricación.

2.4.1. Clasificación de los Rodamientos Básicamente hay tres formas de clasificar los rodamientos:

- **Según la dirección de la carga que mejor soportan**
 - *Rodamientos Axiales:*
 - *Rodamientos Radiales*
 - *Rodamientos de contacto angular*
 - *Rodamientos rígidos*
 - *Rodamientos rotulados*
- **Según la rigidez del rodamiento**
- **Según el elemento rodante**

En el prototipo MHCNC se utilizan 19 rodamientos radiales para minimizar el rozamiento y evitar el desgaste. Los rodamientos radiales son diseñados para resistir cargas en dirección perpendicular al eje. Constan en forma general de tres piezas: Un aro exterior, un aro interior y un elemento rodante con algún tipo de canastillo o jaula. Por ejemplo, las ruedas de un carro se apoyan en el suelo y reciben la carga en el eje, de esta forma los rodamientos de las ruedas trabajan bajo carga radial.

Figura 20. Rodamientos Radiales



Al catalogar un rodamiento es útil entregar una información completa, indicando los tres conceptos anteriores, por ejemplo: Rodamiento radial rígido de bolas, rodamiento radial rotulado de barriletes, rodamiento axial rígido de conos.

Afortunadamente los fabricantes de rodamientos han mantenido una numeración estándar en todas las marcas, permitiendo una identificación sencilla de los rodamientos en base a un número y en ocasiones acompañado de unas letras.

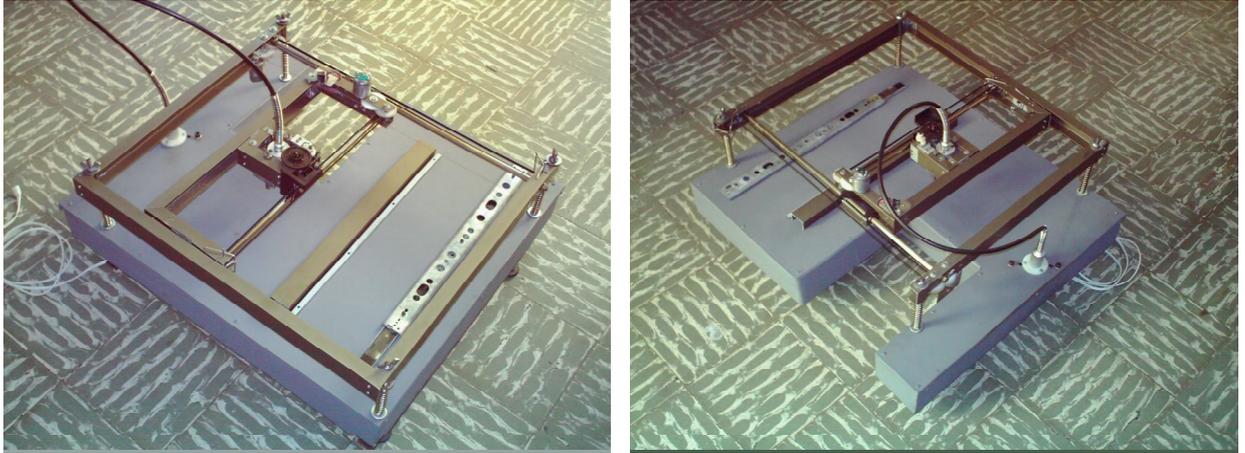
2.5. DESCRIPCIÓN GENERAL DE LA MHCNC

Para la comprensión de la descripción del diseño y fabricación del prototipo MHCNC se han nombrado los siguientes mecanismos independientemente funcionales:

- **Núcleo** lo constituyen el conjunto de mecanismos y piezas que permiten asegurar la herramienta, de igual manera acondiciona el rotor principal para que el actuador transfiera el movimiento rotacional para la acción de perforar.
- **Cabeza** lo conforman el conjunto de mecanismos y piezas que permiten desplazar y medir el desplazamiento del *Núcleo* en un movimiento translacional en la dirección +/- Z.
- **Coordenada X** lo componen el conjunto de mecanismo y piezas que permiten desplazar y medir el desplazamiento de *la Cabeza* en un movimiento translacional en la dirección +/-X.
- **Coordenada Y** lo constituyen el conjunto de mecanismos y piezas que permiten desplazar la *Coordenada X* en un movimiento translacional en la dirección +/-Y.
- **Soporte impreso** lo conforman el conjunto de mecanismos y piezas que permiten al operario ajustar y asegurar la placa de baquelita a ser perforada.
- **Base** lo componen el conjunto de mecanismos y pieza que soportan la *Coordenada Y*, el soporte de impreso, a su vez; en el se encuentran instalados los circuito electrónicos encargados del control del prototipo MHCNC y el motor de corte.

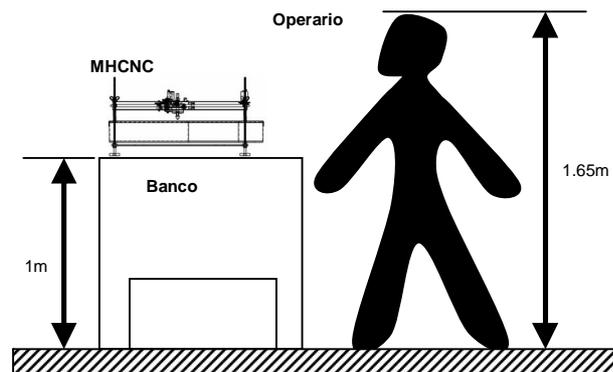
En la figura 21 se observa el prototipo MHCNC

Figura 21. Fotografías isométrica frontal y posterior de la MHCNC



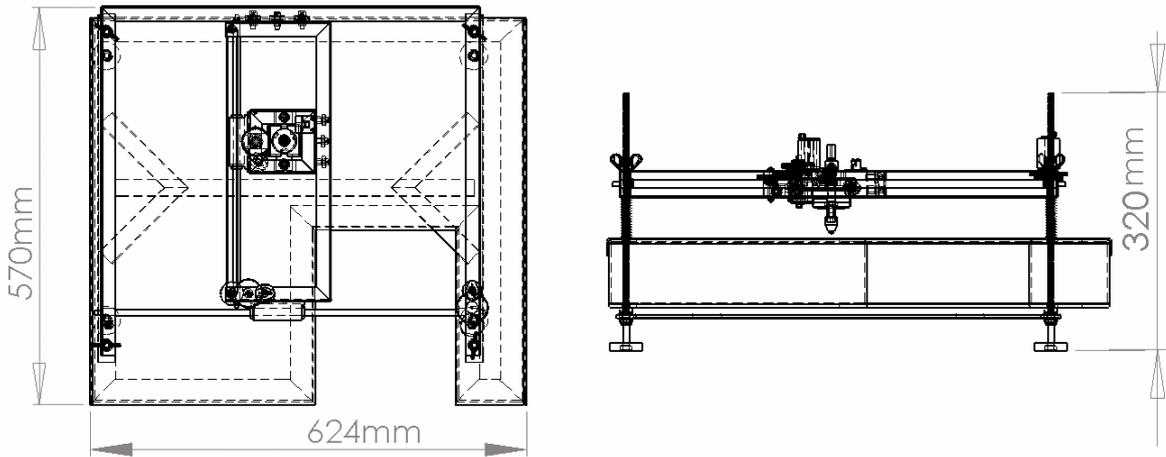
El operador asegura la placa de circuito impreso, ubica la herramienta del trabajo actual y debe iniciar el proceso de perforación desde el Botón “INICIO/PARADA” ubicado en la MHCNC. También debe monitorizar problemas como el quiebre de la herramienta o un atascamiento. Para obtener un campo de visión que permita la observación de estos inconvenientes y facilitar el cambio de herramienta la fig.22 muestra la ubicación propuesta para satisfacer esta necesidad.

Figura 22. Ubicación MHCNC



2.5.1. Dimensiones

Figura 23. Dimensiones MHCNC



2.5.2. Herramientas

Figura 24. Fotografía de herramientas: brocas, collets y portaherramienta



- **Broca** Es el componente mecánico utilizado en conjunción a un taladro o máquina afín, para la creación de un hoyo o agujero durante la acción de taladrar. La gran diversidad de éstas, como la gran cantidad de industrias que emplean este tipo de herramienta, hace que cierta broca pueda ser muy común y corriente o altamente especializada, rara o cara.

Las brocas tienen diferente geometría dependiendo de la finalidad con que éstas hayan sido fabricadas. Diseñadas específicamente para quitar material y formar, por lo general, un orificio o una cavidad cilíndrica, la intención en su diseño incluye la velocidad con que el material ha de ser removido y la dureza del material y demás cualidades características del mismo ha de ser modificado.

La broca común es generalmente recta, de una misma sección, con diferente filo y ángulo de punta, dependiendo el uso se usa para madera, plástico, metales ferrosos, metales no ferrosos, concreto.

Figura 25. Broca



Las brocas se desgastan debido a la fricción, y se quiebran debido al desgaste, aunque también por fuerzas no perpendiculares, o una mala incidencia en la pieza a perforar. Las brocas son piezas consumibles y por tal razón deben ser de uso común, fácil acceso y bajo costo.

La tabla 1, muestra el diámetro de las brocas de acero rápido comercialmente disponibles.

Tabla 1. Diámetros de Brocas Comerciales

mm	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	5.5	6.0	6.5	7.0	7.5	8.0
inch	1/64	1/13	5/64	3/32	7/64	1/8	9/64	5/32	11/64	3/16	13/64	7/32	15/64	¼

El mandril seleccionado para ser usado en el prototipo MHCNC permite montar en él, herramientas que van desde 1/64" hasta 1/8". Este rango es ideal para el propósito del equipo, ya que los componentes electrónicos vienen con conexiones eléctricas cuyos diámetros no supieran 3/32" y la tortillería para sujeción del circuito es normalmente de 3mm.

2.5.3. Actuator para Corte Escoger el motor adecuado para mover la herramienta para corte (sobre su propio eje), requiere que se estime la resistencia que ofrece la placa de baquelita al momento de ser perforada. El motor debe tener un torque tal que supere esta resistencia.

En el intento de encontrar un valor mínimo necesario para el torque que necesita el motor, se encontró que en muchas ocasiones la broca se frenaba, en unos casos debido a su desgaste y en otros, a que al realizar perforaciones pasantes (traspasar la placa), llega un momento en que la delgadez de la capa que queda por traspasar se rompe por presión y no por el corte de la broca, causando que el rotor se frene debido a los astillamientos de la baquelita. Una posible solución a este inconveniente es escoger un motor con un torque sobre dimensionado, sin embargo, esto obliga a que su peso y volumen se incrementen, junto con el tamaño de sus mecanismos de movimiento, y piezas de soporte; de tal forma, que ubicarlo dentro del núcleo es inapropiado. Es por tal razón, que en el diseño final del equipo, el motor que mueve la herramienta para la acción de corte, no se encuentra montado sobre el mecanismo de posicionamiento. En lugar de ello, es montado sobre la base y el movimiento es transmitido a través de una guaya flexible especial utilizada para este tipo de herramienta. Esta permite que el mecanismo de posicionamiento (eje), no tenga que soportar el peso del motor.

Se seleccionó un motor AC de escobillas con un potencia de $\frac{1}{4}$ de caballo (HP) que garantiza un torque de arranque de aproximadamente 750g-cm y velocidad máxima de 2200 rpm, características similares a los motores utilizados en los electrodomésticos como procesadores de alimento o licuadoras.

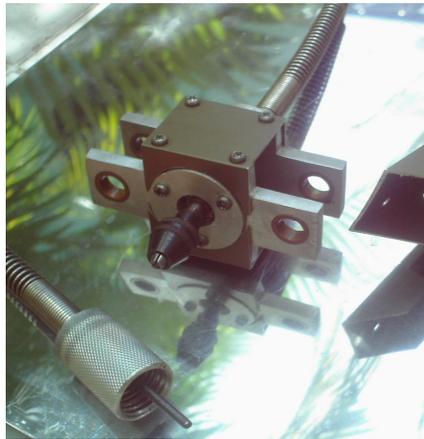
Figura 26. Fotografía de la vista inferior del MHCNC



Motores DC de imán permanente que proporcionarían un a fuerza par cercana, requerirían voltajes DC muy altos y un control mas complejo, sin contar con el costo del mismo.

2.5.3. Núcleo

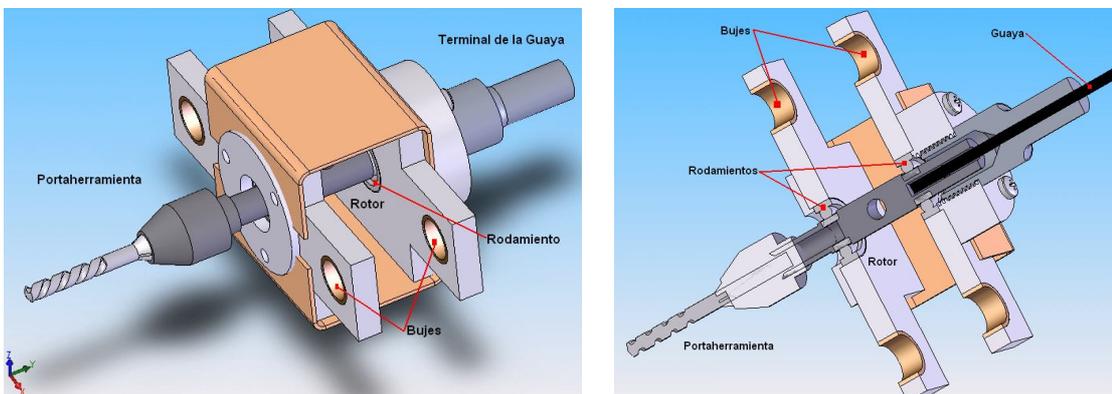
Figura 27. Fotografía del Núcleo MHCNC



Este mecanismo fue diseñado para cumplir las siguientes funciones:

- Adecuar al rotor el movimiento transmitido por la guaya.
- Permitir el movimiento rotacional del rotor sin rozamientos.
- Admitir el montaje del portaherramienta.
- Asegurar condicionamientos especiales para que este pueda desplazarse en la coordenada z, perpendicularmente a la placa.

Figura 28. Ensamble del Núcleo (SolidWorks)



Se escogieron los rodamientos radiales rígidos de bolas para el ensamble del núcleo, estos aseguran que el rotor gire libremente a causa del movimiento transmitido por la guaya, que se encuentra asegurada por medio de un tornillo prisionero. A su vez el rotor, permite asegurar el portaherramientas en el extremo opuesto.

En la figura 28, se observa los bujes para el deslizamiento en la coordenada Z y un corte transversal de núcleo montado sobre rodamientos SKF que garantizan su funcionamiento a más de 30.000 rpm

Portaherramienta El portabrocas de cierre rápido DREMEL (4486) permite un cambio sencillo e inmediato de los accesorios de su multiherramienta DREMEL sin necesidad de utilizar boquillas. Admite accesorios con vástago de 0,4 mm a 3,4 mm.

Figura 29. Portabrocas DREMEL



www.dremel.com

2.5.5. Coordenada Z La coordenada Z lo componen los mecanismo y piezas de soporte que permiten al núcleo desplazarse perpendicularmente con respecto a al pieza a perforar (+/-Z), garantizando la fuerza necesaria para mantener la velocidad de avance adecuada.

Existen mecanismos como pistones neumáticos o hidráulicos que dan un buen nivel de precisión y fuerza, pero que llevan consigo el uso de compresores y pistones especializados que elevarían el costo equipo. Para lograr este movimiento translacional (lineal), la mejor opción para el actuador es un motor DC, que por medio de engranajes convierte el movimiento rotacional en lineal, y al mismo tiempo, reduce la velocidad para ganar fuerza.

Las revoluciones del taladro deben ser menores cuanto mayor sea el diámetro de la broca y cuanto mayor sea la dureza del material a trabajar.

Ahora, se deben relacionar tres variables, para realizar el proceso de perforación de una lámina de circuito impreso.

- *Velocidad de corte*
- *Velocidad de avance*
- *Diámetro de la broca*

Las dos primeras son inversamente proporcionales al diámetro de la broca, y la velocidad de avance puede ser aun mayor al aumentar la velocidad de corte. Si observamos el Anexo A, se puede obtener el avance de la broca en metros por minuto en función del diámetro de la broca y la dureza del material. Aunque no se encuentra especificada la materia de baquelita (felónica o fibra de vidrio), podemos asegurar que su dureza es inferior al acero de 50kgs/mm². Utilizando una broca de diámetro 1/8" (≈3.2mm) se determina una velocidad máxima de avance, para este material, de 10mm por segundo.

La tabla proporciona una idea de la velocidad de avance al perforar, si la velocidad de avance se incrementa constantemente por encima del valor estimado (Anexo A), puede darse el caso, que a determinada velocidad la herramienta se quiebre o se atasque, contando con que el actuador pueda mantener la fuerza que mueve a la herramienta en forma perpendicular a la operación de taladrar.

Ante la imposibilidad de encontrar información técnica de la fuerza necesaria (en función del diámetro y la velocidad de corte) para mantener la velocidad de avance al perforar un placa de de baquelita de 5mm, recubierta con una capa de 1mm de cobre, se encontró un valor de referencia en forma experimental. Para ello se tomó un mototool con un peso de 200 gramos, 5 velocidades diferentes ($V_{\text{máx}}=30.000\text{rpm}$), una herramienta de 1/8" (máxima). El Mototool fue descargado perpendicularmente sobre una placa de baquelita, tratando de que la única fuerza que actuara sobre él, fuese la gravedad.

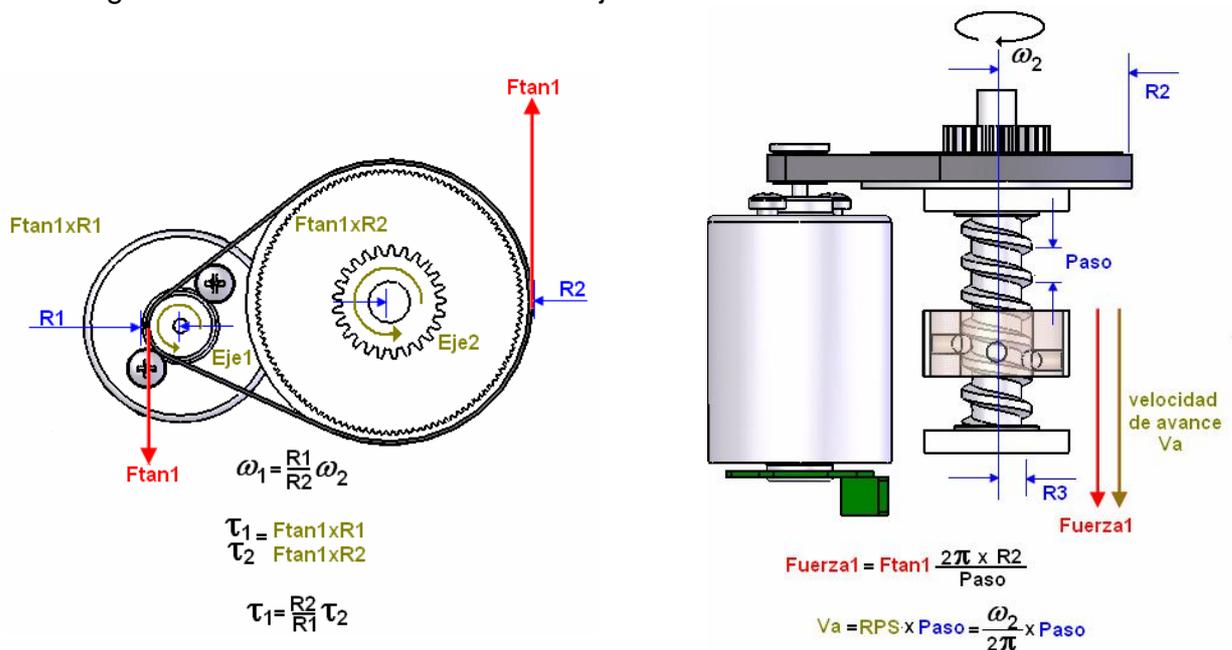
Como resultado se obtuvo que en velocidades en el rango de 1000rpm hacia delante, el tiempo en traspasar la placa no supero el segundo, además; este disminuyo al incrementar la velocidad.

Este sencillo experimento permite concluir, que una fuerza igual a la masa del mototool por la aceleración de la gravedad, es un punto de partida para decidir la fuerza necesaria para estimar la fuerza del actuador.

A través del control electrónico es posible garantizar la velocidad de avance y la velocidad de corte adecuada, para la herramienta seleccionada. Además, se selecciona una fuerza dos veces mayor que el peso del Mototool, sobre la baquelita, que es un buen parámetro para iniciar el desarrollo del mecanismo.

En la figura 30, se muestra el mecanismo utilizado para transmitir el movimiento al núcleo. Consta del motor actuador, que transmite el movimiento a un tornillo por medio de una correa. Asociado al tornillo, una tuerca convierte el movimiento rotacional a lineal, a través de balines (ubicados en su interior), que se desplazan entre los filetes del tornillo. Finalmente la tuerca es asegurada al núcleo, para transmitir el movimiento.

Figura 30. Transmisión Movimiento Eje Z



El motor genera un par igual a $F_{tan1} \times R_1$, debido a su material y diseño suponiendo que la correa no presenta deformaciones (en el rango de operación normal). La fuerza (F_{tan1}) es transmitida a la segunda polea para cuyo eje se tiene un par igual a $F_{tan1} \times R_2$.

Se necesita entonces, que la **fuerza1** sea por lo menos dos veces la estimada en el experimento igual a 400g (4N con $g=10m/s^2$). De la ecuación de la **fuerza1** se obtiene la relación entre la fuerza línea final con respecto a la tangencial inicial, en la polea del motor.

La ecuación:

$$Fuerza1 = F_{tan1} \frac{2\pi \times R_2}{Paso}$$

Despejando F_{tan1} se tiene:

$$F_{tan1} = \frac{\text{Fuerza1} \times \text{Paso}}{2\pi R_2}$$

Paso=5mm

Fuerza1=400g (4N)

Radio₁=9mm

Radio₂=20mm

$$F_{tan1} = \frac{400g \times 5mm}{2\pi 20mm} = 15,91g$$

Por tanto el τ_{Motor} mínimo para garantizar una fuerza de 400 gramos es:

$$\tau_{Motor} = F_{tan} \times R_1 = 15,91g \times 9mm$$

Normalizando a cm.

$$\tau_{Motor} = 14,32g \times cm.$$

Para simplificar los cálculos, no se ha incluido el peso del mecanismo núcleo, sin embargo este afectaría la ecuación si se tienen en cuenta los coeficientes de fricción entre el tornillo y la tuerca, ya que el peso del núcleo descansa sobre los filetes del tornillo, afectando el torque necesario al momento de elevar la pieza a su posición de inicio. No obstante insertar el peso del núcleo (80g), como una adición de la fuerza1 (que sería el caso más drástico) y evaluarlo en la ecuación del torque mínimo necesario para el motor da como resultado.

$$\tau_{Motor} = 17,18g \times cm.$$

En subida

$$\tau_{Motor} = 11,46g \times cm.$$

En bajada

El motor DC de referencia KC380SP-14250⁹ cuyo fabricante Kindly Industrial Co. Ltd es utilizado para el prototipo MHCNC, sus principales características son:

Rango de operación: 9.0 hasta 30.0V; Nominal: 12.0V

Velocidad libre de carga: 4800rpm; amperaje: 0.08A

Velocidad con eficiencia máxima: velocidad: 3798 rpm; amperaje: 0.30A;

Torque: 50.09g-cm.

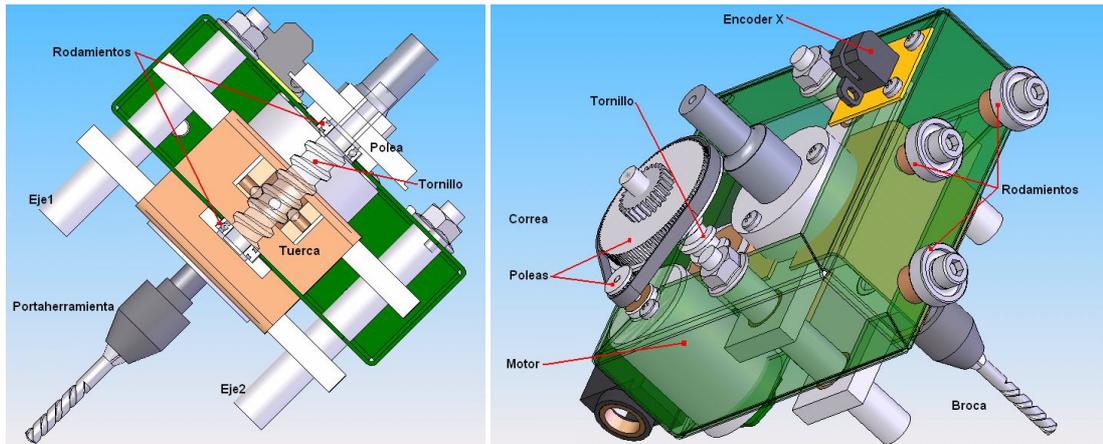
Torque de atascamiento: 240 g-cm.

El torque que puede proporcionar el motor es suficiente y mas que necesario para superar la fuerza mínima estimada, además de compensar el rozamiento y el efecto del peso ejercido por la guaya, que ha sido despreciado en los cálculos.

⁹ <http://kindly.manufacturer.globalsources.com/si/6008800993902/Homepage.htm>

Mecanismo Cabeza MHCNC

Figura 31. Ensamble Cabeza MHCNC (SolidWorks)

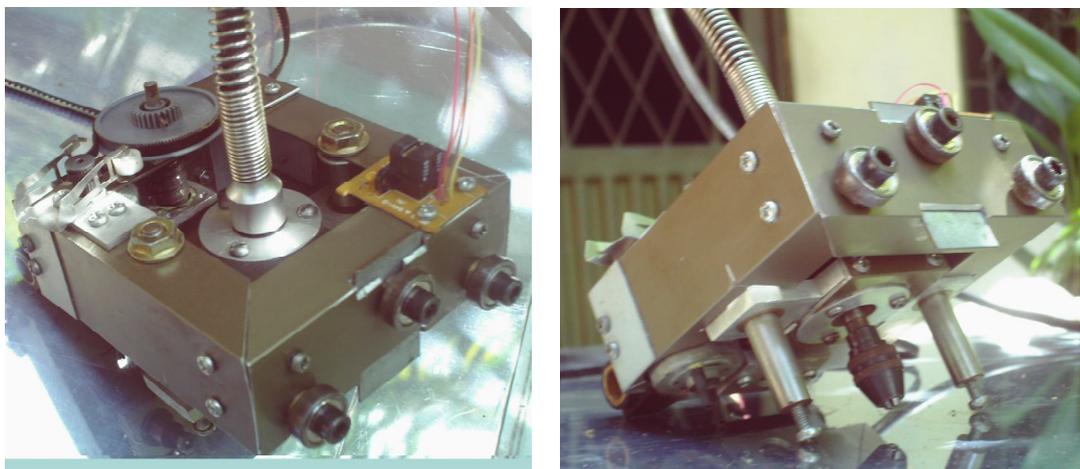


Una vez estimado el torque mínimo requerido para el motor, es preciso finalizar el diseño del mecanismo que hará girar perpendicularmente al núcleo, para realizar el proceso de perforación.

En la figura 31 y 32, se muestra el diseño y una vista real de la cabeza en el prototipo MHCNC ya finalizada.

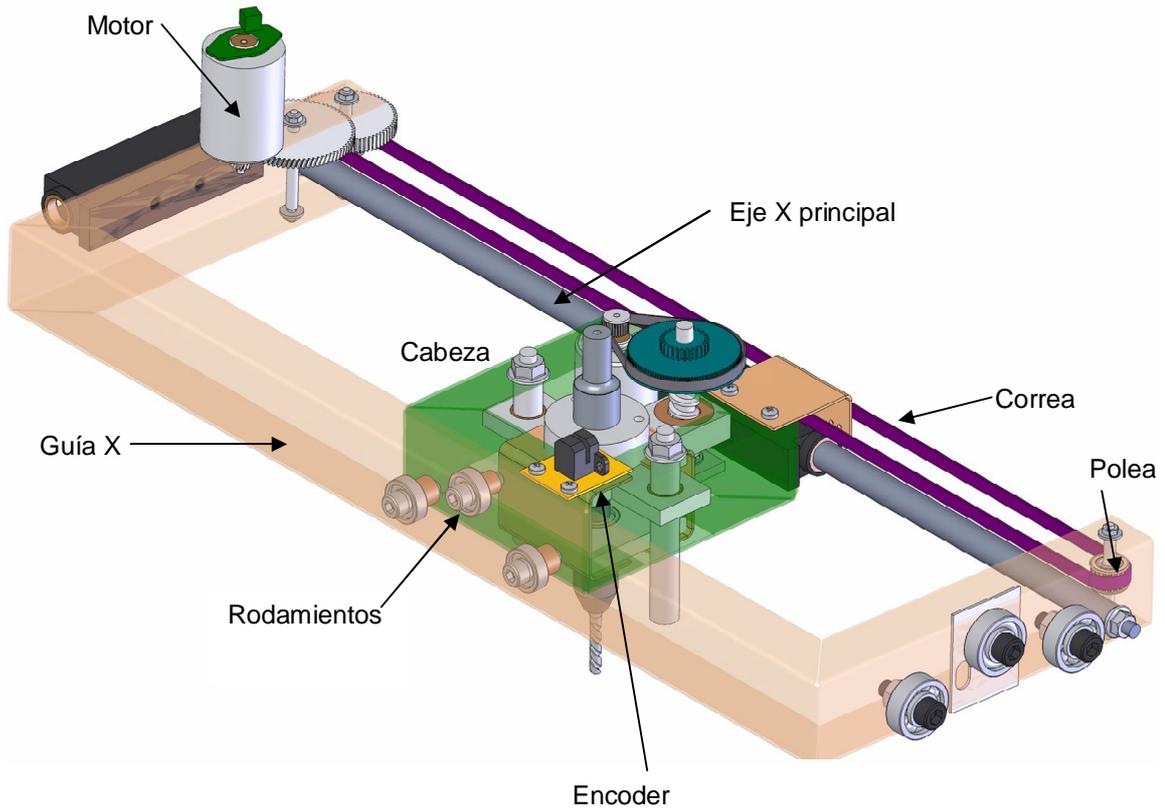
Consta de dos ejes en los que se desplaza el núcleo guiado por bujes, estos son completamente paralelos, al igual que los bujes. También se observa un corte transversal de la cabeza, que muestra el mecanismo usado para desplazar el núcleo a través de la coordenada Z (algunos componentes están ocultos para mejorar la visualización), y el motor actuador transmitiendo el movimiento por medio de la correa.

Figura 32. Fotografía cabeza MHCNC



2.5.6. Coordenada X

Figura 33. Ensamble Coordenada X (SolidWorks)



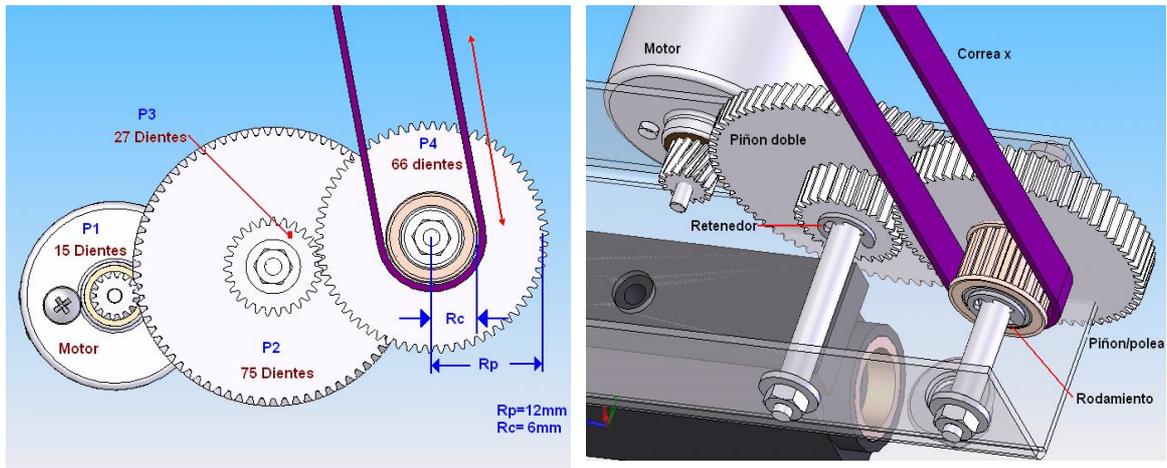
Los mecanismos que intervienen en la coordenada X permiten el movimiento lineal de la cabeza, por medio de un actuador (motor). En un espacio de 27 cm. La cabeza se desliza sobre un eje de acero paralelo a la guía X, por donde, en esta última, los rodamientos de la coordenada X se deslizan y consiguiéndose un movimiento paralelo con un mínimo de rozamiento.

El objetivo del mecanismo es posicionar la herramienta en la coordenada X requerida para la perforación.

La fuerza requerida para desplazar la cabeza es mínima, ya que el sistema posee una fricción inapreciable gracias a la utilización de rodamientos y una buena lubricación del eje. Cuando los efectos de la gravedad sobre las piezas que componen un mecanismo se minimizan a tal punto que pueden ser despreciados, se dice que el sistema es emboladizo.

En la figura 33, se observan los rodamientos necesarios para el desplazamiento en la coordenada X. Todo este mecanismo es montado en un esquema similar al de la cabeza, sobre el mecanismo de la coordenada Y.

Figura 34. Ensamble del Sistema para la transmisión del movimiento en la coordenada X.



En la figura 34, se muestra el ensamble del mecanismo reductor de velocidad que mueve la cabeza en la coordenada X.

La reducción de velocidad efectuada por este juego de piñones:

$$\omega_{P4} = \omega_{P3} (N_{P3}/N_{P4}) = \omega_{P3} (27/66) = 0.41 \omega_{P3}$$

$$\omega_{P3} = \omega_{P2}$$

$$\omega_{P2} = \omega_{P1} (N_{P1}/N_{P2}) = \omega_{P1} (15/75) = 0.2 \omega_{P1}$$

$$\omega_{P4} = \omega_{P1} (N_{P1}/N_{P2}) (N_{P3}/N_{P4}) = 0.082 \omega_{P1}$$

En término de revoluciones por segundo

$$\text{rps}_{P4} = 0.082 \text{rps}_{P1}$$

Que indica que el motor girando a su velocidad máxima 3600rpm (60rps) hará girar el piñón 4 a una velocidad de 295 rpm (4,92rps). Esta relación también proporciona el incremento del torque en el eje del piñón 4:

$$\tau_{P4} = (1/0.082) \tau_{\text{motor}} = 12.2 \tau_{\text{motor}}$$

Es así, que utilizando un motor con las mismas características, al usado como actuador para la coordenada Z, suministra un torque en el piñón 4 que puede estar por encima de:

$$\tau_{P4} = 12.2 \times 50.09 \text{g-cm.} = 611,09 \text{g-cm.}$$

Como el radio de la polea es igual a 6mm, se puede proporcionar una fuerza tangencial en la polea F_{Tx} (tensión de la correa como fuerza lineal) de un valor igual a:

$$F_{Tx} = \tau_{P4} \text{ normalizado a } 6 \text{mm} = 1018,5 \text{g} = 1,0185 \text{ Kg.}$$

Ahora se relaciona la velocidad angular del piñón 4 con el desplazamiento de la correa X de la siguiente manera

$$\omega_{P4} = \omega_{Polea}$$

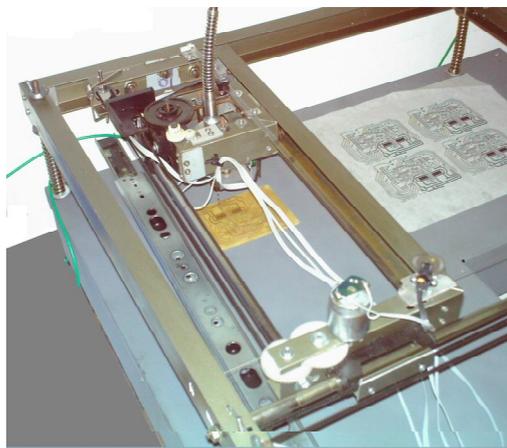
Por cada revolución del piñón 4 la correa se desplaza en la dirección de giro una distancia igual al perímetro de la polea ($2\pi r$) con $r_c = 6 \text{ mm}$.

$$\text{Perímetro de la polea} = 2\pi r_c = 37,7 \text{mm}$$

De manera que en el caso más drástico donde el motor gira a su velocidad máxima (3600rpm@ 24v) el desplazamiento del mecanismo sujeto a la correa será a una velocidad:

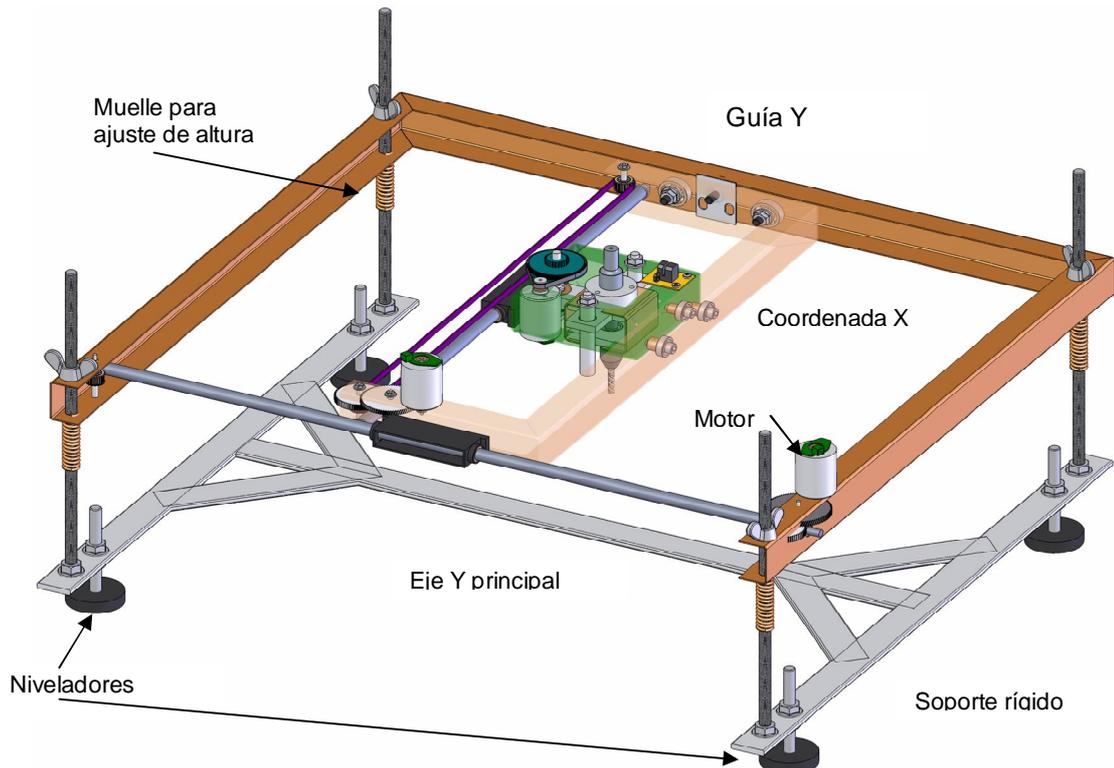
$$V_{x_{max}} = \text{rpm}_{P4} \times \text{perímetro de la polea} = 185,5 \text{ mm./seg.} = 18,55 \text{cm. /seg.}$$

Figura 35. Fotografía coordenada X



2.5.7. Coordenada Y

Figura 36. Ensamble Coordenada Y (SolidWorks)



Los mecanismo de la coordenada Y se encuentra asegurados al soporte rígido del prototipo MHCNC. Como se mencionó anteriormente, el principio mecánico de la coordenada X y la coordenada Y es el mismo. El movimiento del motor es reducido por la acción de un juego de piñones (idéntico al de la coordenada X) y es transmitido por medio de una correa al mecanismo (coordenada X) para deslizarlo, guiado por un eje principal. La guía Y, y el eje principal soportan el peso del mecanismo, con una buena lubricación y el uso de rodamientos tensionados hacia las paredes internas de la guía, el sistema mecánico es un sistema emboladizo. Aunque los mecanismos de las coordenadas X e Y son similares, sus diferencias radican en los momentos de inercia para cada una de ellas. Esto afecta los parámetros a la hora de sintonizar el controlador PID de cada mecanismo.

Realizado el mismo procedimiento para obtener la velocidad máxima $V_{X_{max}}$, el torque en la polea (piñón 4) τ_{P8} y fuerza de tensión máxima proporcionada sobre la correa F_{Tx} en la coordenada X, se consigue los valores para la coordenada Y:

$$V_{Y_{max}}=18,55\text{cm. /seg.}$$

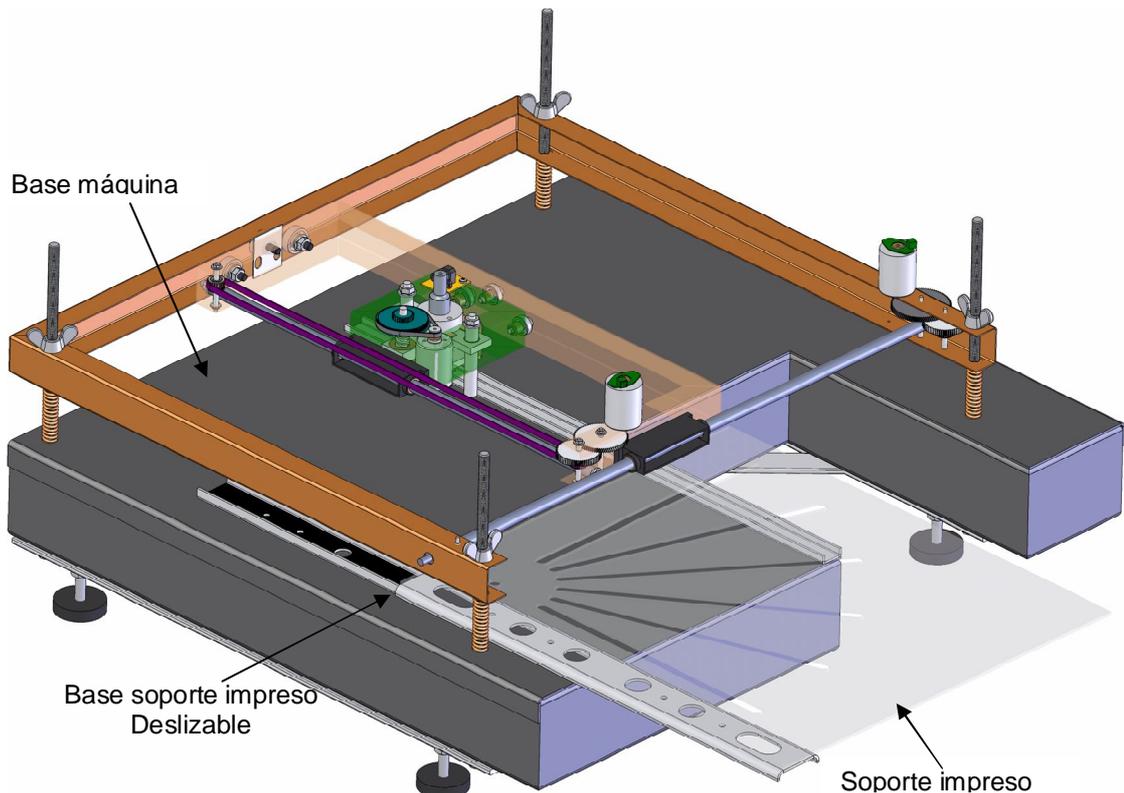
$$\tau_{P8} = 12.2 \times 50.09 \text{g-cm.} = 611,09 \text{g-cm.}$$

$$F_{Ty} = \tau_{P8} \text{ normalizado a } 6 \text{mm} = 1018,5 \text{g} = 1,0185 \text{ Kg.}$$

2.5.8. Base y Soporte de impreso La base del mecanismo, es un rectángulo de acero, hueco en su parte posterior, utilizada para albergar los circuitos electrónicos, los mecanismos para la bandeja donde se sujeta la placa de baquelita, el motor principal AC con los dispositivos que aseguran la guaya. Además posee una hendidura, realizada con el propósito de que la cabeza principal de la máquina utilice este espacio para posicionarse en el momento que el operario necesite tener a su alcance el núcleo de la máquina, para realizar un cambio de herramienta, o cualquier otro proceso.

El soporte del impreso permite asegurar placas de baquelita de diferentes tamaños por medio de varios seguros que se desplazan en seis ranuras adecuadamente realizadas para este propósito. También es posible desplazarlo hacia el interior y exterior de la base de la máquina, ya que cuenta con dos bases laterales, una de los cuales es deslizable.

Figura 37. Ensamble Base y Soporte de impreso de la MHCNC



2.5.9. Modelado Los mecanismos cabeza, coordenada X y coordenada Y son cinemáticamente independientes para el propósito de la MHCN. Ya que la perforación es un trabajo punto a punto, las coordenadas de posicionamiento pueden desplazarse independientemente, es decir, no es necesario la sincronización a cada momento.

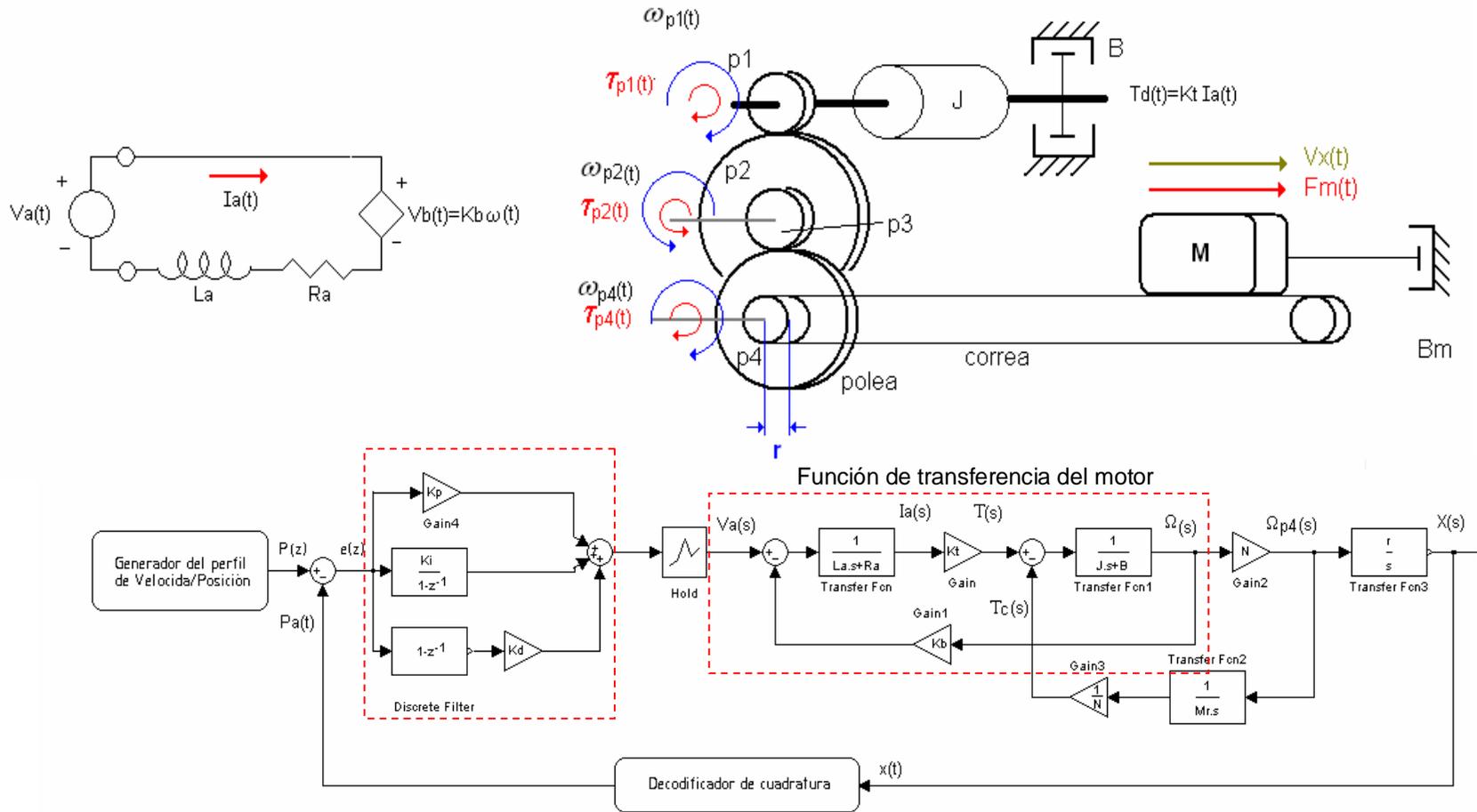
En un primer intento de encontrar la función de transferencia del mecanismo se esboza el diagrama de bloques de la coordenada X para obtener el modelo matemático de la planta (figura 38), en donde la masa **M** corresponde al conjunto de las piezas que constituyen el mecanismo de la cabeza, los torques τ_{p1} , τ_{p2} , τ_{p4} y las velocidades angulares ω_{p1} , ω_{p2} , ω_{p4} corresponden a los ejes de los piñones p1, p2 y p4 respectivamente, **J** a la inercia del rotor del motor, **B** a la fricción viscosa en los cojinetes del motor, **Bm** a la fricción viscosa en los puntos de deslizamiento del mecanismo cabeza, **Va**, **La**, **Ra** son el voltaje, la inductancia y la resistencia de armadura del motor, **Vb** el voltaje producido por la fuerza contraelectromotriz que es proporcional a la corriente de armadura **la** por un factor **Kb**.

En la figura 38, se muestra también el diagrama de bloques para esta coordenada, incluyendo el controlador digital y la realimentación. La correspondencia de engranajes maximiza la relación de la carga al par aplicado por el motor en un factor **N** (0.082 para la coordenada X e Y), y si la velocidad de desplazamiento del mecanismo no es exageradamente alta, el factor de amortiguamiento viscoso puede ser despreciado gracias a que el mecanismo ha sido diseñado para que sea un sistema emboladizo.

El decodificador de cuadratura es un módulo electrónico que utiliza un encoder óptico asociado con un disco ranurado o cinta codificada para almacenar en un registro de 32 bits la posición, en todo instante de tiempo. Los encoders ópticos han sido fabricados para medir velocidades de hasta 30.000 rpm con discos de una resolución de 150Lpi (líneas por pulgada). Estos no agregan tiempos de retardo, al sistema, ya que al ser utilizados para medir posicionamiento, se encuentran en un rango muy por debajo de sus especificaciones máximas. Por tanto en el lazo de la realimentación, el bloque del decodificador de cuadratura solo se muestra para indicar que es utilizado para adecuar el valor de la posición actual para los cálculos en el PID digital (**Pa(t)**).

El generador del perfil de velocidad es también un módulo electrónico que determina la trayectoria programada y es proporcionada al sistema como un valor discreto (**P(z)**). El periodo de generación de los valores de posición **T**, es el mismo utilizado para los cálculos por parte del PID para minimizar el error de posición (**e(z)**).

Figura 38. Modelado Sistema de Regulación Automática Coordinada X



$P(z)$ = Posición deseada
 P_a = Posición actual
 V_a = Voltaje de armadura
 i_a = Corriente de armadura
 T = Torque en el rotor
 T_c = Torque debido a la carga

$\Omega_{p4}(s)$ = velocidad angular en el piñón 4 (Dominio continuo)
 $\Omega(s)$ = velocidad angular (Dominio continuo)
 N = relación del juego de piñones (N_1/N_2)(N_3/N_4)
 $X(s)$ = posición Dominio continuo
 r = radio de la polea

L_a = inductancia de armadura
 R_a = Resistencia de armadura
 J = Inercia del rotor
 B = amortiguamiento viscoso
 M = masa del mecanismo
 K_b = ganancia de fuerza contraelectromotriz

El período de muestreo T , es un parámetro de diseño importante que suele seleccionarse en función de un compromiso entre el deterioro de la calidad del control que puede darse por la selección de un valor alto de T y la potencia de cálculo necesaria para ejecutar el algoritmo de control con valores de T pequeños. El valor del periodo de muestreo para el control de los sistemas en el prototipo MHCNC desarrollado, es de 512 μ seg, por tanto la señal correctiva del error a la salida del control PID se acerca a una señal de tiempo continua ya que las componentes de más altas frecuencias tienen un periodo mayor que el muestreo. En el capítulo cuatro se explica con más detalle su procedencia.

Ante la imposibilidad de obtener los valores de las constantes de las plantas de los motores se recurre a otros medios para la adquisición de la función de transferencia del sistema.

El método de la tangente de Ziegler-Nichols⁸ es una técnica de identificación experimental basada en la respuesta del sistema a una entrada escalón (*curva de reacción del proceso*). Se trata de un método de lazo abierto, cuyo procedimiento es básicamente gráfico. Este método se utiliza para identificar modelos simples, normalmente modelos de primer o segundo orden más retardo, en caso que exista. Sin embargo, la aplicación de esta técnica al sistema de posicionamiento en lazo abierto no es viable, ya que los motores DC son diseñados para que a partir de un voltaje V_a , se obtenga una velocidad angular en el rotor. Aun si la planta en lazo abierto se tomara con una realimentación de la posición (adecuada proporcionalmente al tipo señal de entrada), la velocidad del cambio angular del rotor sería muy elevada dando como resultado una planta inestable, y solo con una reducción muy alta de la velocidad angular por medios mecánicos, como engranajes, se conseguiría algo de estabilidad.

El principal objetivo de obtener la función de transferencia del sistema mecánico es aplicar las técnicas matemáticas y obtener los coeficientes del filtro de compensación PID, para realizar la cancelación y desplazamientos de polos y ceros con el fin de contrarrestar los efectos desestabilizantes del sistema, sin embargo debido a las tolerancias, a los efectos no lineales y otras variables no asumidas, los resultados de una simulación pueden ser engañosos y no realistas con la realización práctica. Por tal razón en los sistemas de control de procesos donde la dinámica del sistema no es bien conocida o no está bien definida, se sintonizan de forma experimental los controladores PID⁹. Es así que en el desarrollo del prototipo MHCNC, se enfoca al desarrollo de medios para aplicar métodos y obtener una correcta sintonización del filtro de compensación PID, descritos en el capítulo 4.

⁸ Ingeniería De Control. Modelado Análisis Y Control De Sistemas Luis Moreno, Santiago Garrido Carlos Balaguer (Ariel) 1ª edición octubre 2003 sección 7.4.2 Métodos de Ziegler - Nichols Método de la respuesta a un escalón

⁹ Sistemas de control en tiempo discreto, segunda Edición, Katshito Ogata. Sección 3-5 Función de Transferencia Pulso, Pág. 121

3. INTERFAZ ELECTRÓNICA

3.1. ARQUITECTURA DE LA INTERFAZ ELECTRÓNICA

En la figura 39, se observa siete bloques que componen el circuito electrónico. Cada bloque es independiente de los otros pero están sometidos al procesador principal (microcontrolador) a través de un bus de datos de 8 bits. En el primer bloque se encuentra el procesador anfitrión, el control de flujo del bus y la selección de chip, el segundo bloque corresponde al bloque de almacenamiento, compuesto por una memoria RAM y un latch para su direccionamiento. Los tres bloques siguientes corresponden al control de las tres coordenadas (X,Y,Z), derivado de la programación de su correspondiente controlador de movimiento (LM629), cuyos coeficientes, velocidades, espacios y aceleraciones; varían según la coordenada, Sin embargo; el circuito para cada eje conserva la similitud.

Se observa también, un pequeño bloque determinado como RS232, encargado de convertir los niveles de voltaje TTL a niveles RS232 para recibir y transmitir información al o desde el PC, centrado en el circuito integrado MAX232 (Multichannel RS232 Driver/Receivers) fabricado por MAXIM.

Por último el bloque fuente de poder y control AC, se encarga de generar el voltaje de alimentación para los circuitos lógicos (5 V) y +24V para controlar los motores. También permite el aislamiento y el control de la velocidad del motor AC encargado de mover la herramienta seleccionada.

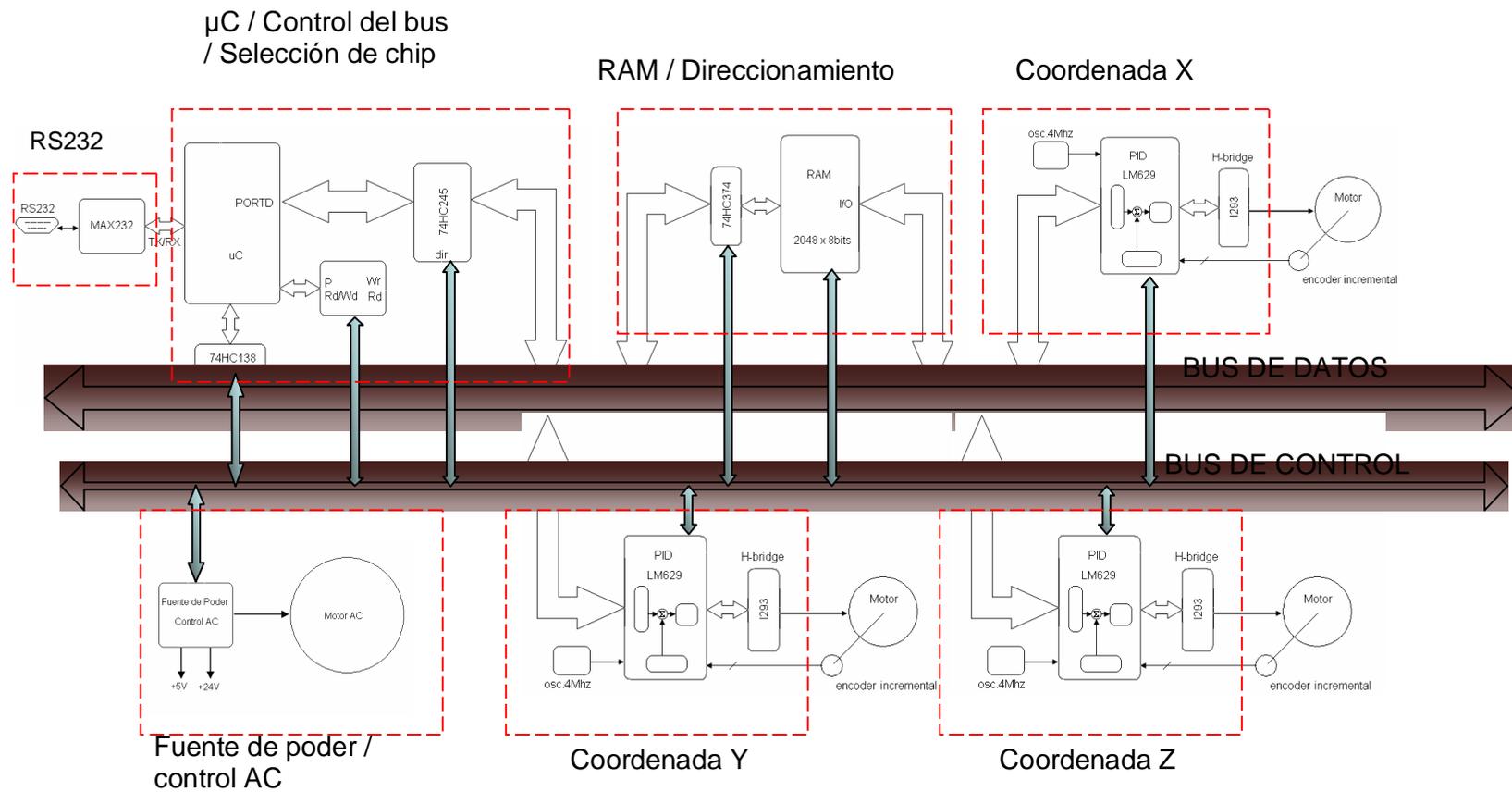
Se observan siete bloques bien definidos que son gobernados por el microcontrolador PIC16F874A, a continuación se describen los circuitos que componen cada uno de los bloques.

3.1.1. Bloque 1: Microcontrolador / Control Del Bus / Selección De Chip Y Rs232

Microcontrolador. El microcontrolador implementado es el PIC16F874, fabricado por MICROCHIP con tecnologías CMOS. Trabajando a una frecuencia de 20Mhz se obtienen ciclos de instrucción de 200 nanosegundos.

$$\text{CiclosdeInstrucción} = \frac{4}{F_{osc}}$$

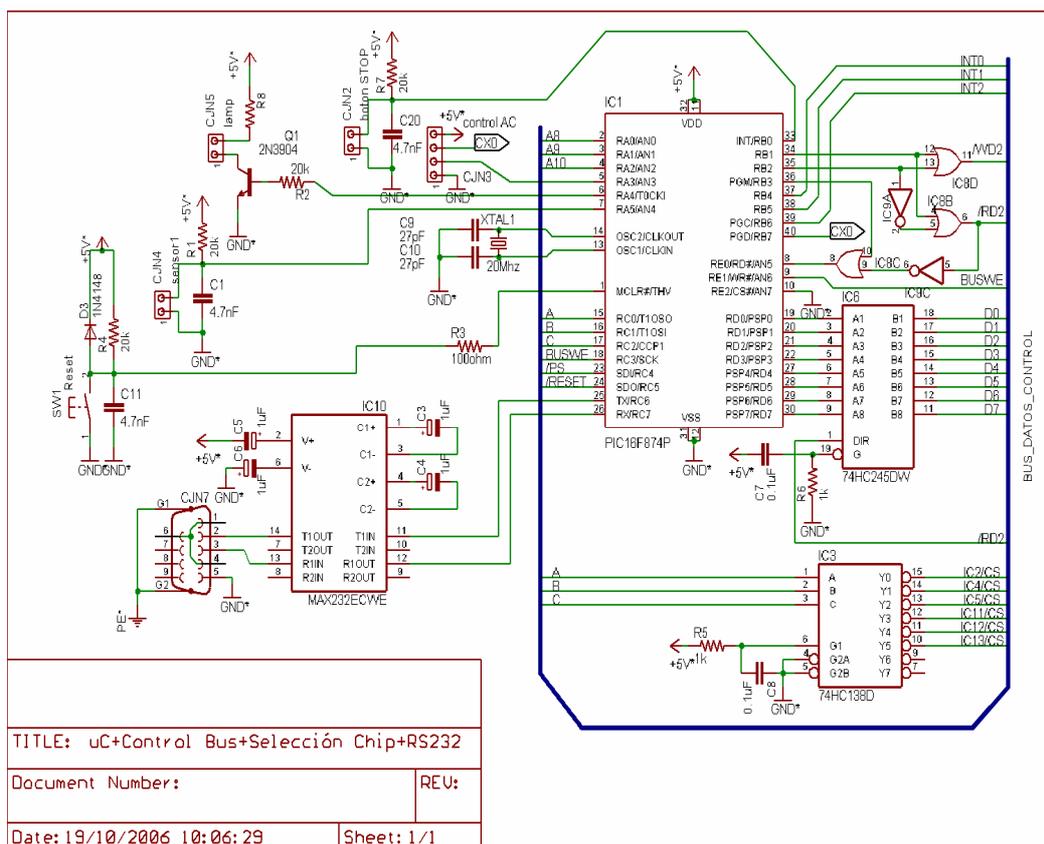
Figura 39. Arquitectura de la Interfaz Electrónica.



Se seleccionó este dispositivo debido a su facilidad de programación y a los periféricos incluidos en el chip, también por la gran cantidad de información disponible sobre él. Su gran desventaja es su incapacidad de realizar operaciones matemáticas medianamente complejas. Sin embargo, todo el proceso computacional de tiempo real es relevado a los circuitos integrados LM629, que son los procesadores de control de precisión del movimiento, y cuya descripción se encuentra en el Anexo B. De igual manera la cantidad de memoria RAM incluida en el chip, insuficiente para el desarrollo de las acciones que controlan las funciones en la máquina, forzó la adición de una memoria RAM estática externa de 2048 palabras x 8 bits.

En el diagrama esquemático de la figura 40, se observan los circuitos integrados utilizados en este bloque. El microcontrolador encargado de gobernar todas las acciones, el transmisor-receptor MAX232 con sus componentes discretos para generar los niveles de voltaje RS232, las señales TX y RX que se dirigen hacia el PC son accedidas a través de un conector DB9.

Figura 40. Diagrama esquemático del circuito de control



El circuito integrado 74HC138 es un demultiplexor de una a ocho líneas con salidas activas en bajo. Este dispositivo permite al procesador seleccionar el dispositivo en el cual se desea escribir o leer a través del bus de ocho bits.

El microcontrolador también posee una memoria EEPROM, utilizada para almacenar datos muy importantes como lo son los coeficientes de cada filtro, compensador para cada eje, espacios límites, velocidad y aceleración máximas, velocidad de perforación y otra serie de datos que son cargados al inicializar el equipo. De igual manera, otras propiedades ventajosas del microcontrolador son su bajo consumo de potencia, su costo, y otras características especiales que le permiten desarrollarse con un mínimo de componentes externos.

Bus Maestro

Configuración Del Puerto D. Como ya se menciona anteriormente, el circuito integrado 16F874 trabaja generalmente, junto con otros dispositivos como Procesadores que utilizan su puerto D como PSP (Puerto paralelo Esclavo) para establecer una comunicación bidireccional, controlada por el Microprocesador. Sin embargo para la realización de la interfaz electrónica de la Máquina Herramienta, el Microcontrolador 16F874 es el dispositivo principal y procesador anfitrión, por lo tanto; es necesario, para establecer una comunicación bidireccional con los demás dispositivos, un **puerto paralelo maestro**.

Un de las formas de lograrlo, es realizar una subrutina de programación para el microcontrolador que le permita reconfigurar el puerto D, cada vez que sea necesario realizar una escritura o lectura de un dato. Sin embargo, esta opción maximiza el riesgo a que se produzca una colisión, ocasionándose el daño a algún dispositivo, o al mismo microcontrolador.

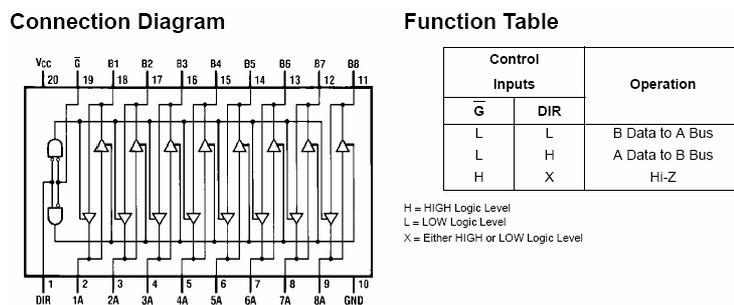
La segunda opción es utilizar un microcontrolador con un bus bidireccional que le permite leer o escribir en cualquier dispositivo, sin necesidad de utilizar circuitería adicional. Sin embargo estos dispositivos, como los Microcontroladores de Hitachi de 8 Bits de la serie H8, requieren sistemas de desarrollo especializados de muy costosa adquisición.

Por último, y la opción implementada, es un arreglo de compuertas lógicas y unas pocas líneas de programación que garantizan que nunca pueda existir una colisión de datos que destruyan un dispositivo. También generando las señales de lectura y escritura para todos los periféricos, controlando la dirección o flujo de datos desde o hacia el uC.

El arreglo de compuertas lógicas (IC8 e IC9, figura.40), es el circuito de control del bus, que en conjunto con el circuito integrado 74HC245, trabajan para acondicionar el bus octal D como maestro y para evitar una colisión de datos. Este

serio problema se puede dar usualmente en el proceso de programación al realizar las pruebas, en donde es posible perder el control del programa, ocasionando (involuntariamente) que el uC coloque un dato en el bus y al mismo tiempo habilite la lectura de un chip periférico, según sean los datos; estos pueden colocar en corto circuito las salidas de cualquiera de los dos dispositivos, destruyéndolo(s).

Figura 41. Diagrama de conexiones y tabla de función del circuito Integrado 74HC245



www.fairchildsemi.com DM74ALS245A Octal 3-STATE Bus Transceiver data sheet

El circuito integrado 74HC245 (figura 41), es un transceptor bus octal con salida de alta impedancia. Éste dispositivo ofrece dos caminos de comunicación entre buses controlados con la señal entrada dir (pin 1). El dato es transmitido desde el bus A al bus B o viceversa. El dispositivo es deshabilitado cuando la señal \bar{G} es colocada en un nivel lógico alto.

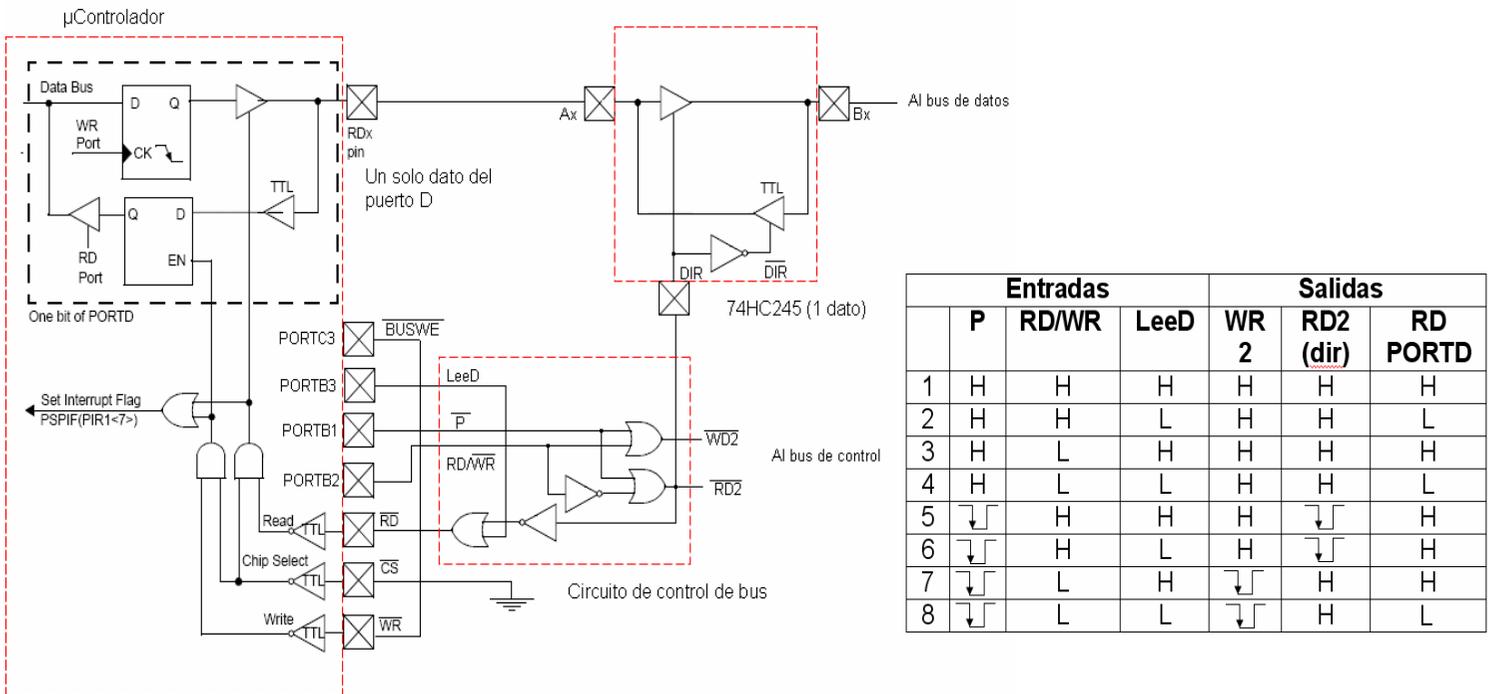
En el diagrama de la figura 42, se muestra el circuito controlador de bus con su tabla en lógica inversa. También se describen los circuitos internos en el microcontrolador que permiten, originalmente, trabajar al puerto D como PSP, pero que en este caso junto con las compuertas lógicas, habilitarán al puerto D para trabajar como maestro. Finalmente, para simplificar la descripción del control de flujo de datos, en el diagrama de la figura 42, se exhibe el control de flujo de un (1) bit al puerto D (puerto de 8bits), realizado por el transceptor 74HC245.

Para evitar que exista una colisión de datos, la señal $\overline{RD2}$ ha sido conectada al pin de dirección del 74HC245, de manera que al encontrarse en un cero lógico, habilitando la lectura \overline{RD} en el periférico seleccionado, la palabra presente en el bus de datos se transferirá del pin Bx al Ax del transceptor y estará disponible en el puerto D para la lectura. Hasta ahora solo se ha garantizado que no exista colisión entre el periférico seleccionado al leerse y el transceptor del bus (74HC245). Si se observa en la tabla de verdad del circuito controlador de bus (figura 42), se advierte que cual fuese la condición de entrada que obligue a la salida a $\overline{RD2}$ encontrarse en un nivel bajo (pulso en la línea 5 y 6 de la tabla, figura 42), la señal \overline{RD} del PORTD nunca será habilitada, de manera que jamás existirá un dato presente al mismo tiempo, por parte del uC o del transceptor del bus, sobre el puerto D.

Proceso De Lectura. Simplificando el proceso de lectura de un periférico, se inicia con todas las señales en un nivel lógico alto (figura.42, línea 1 de la tabla), acto seguido, el firmware del uC selecciona el dispositivo a través del demultiplexor, luego este reafirma la señal RD/WR(PORTB2) en un nivel alto indicando lectura, de igual forma la señal LeeD se mantiene en un nivel alto e instantes después la señal P toma un nivel bajo durante 10us (figura 42, línea 5 de la tabla), tiempo en el cual el periférico coloca el dato a leer en el bus que se encarga de permitir su transmisión al puerto D a través del transceptor del bus. Durante los 10us que el dato esta presente \overline{BUSWE} en el puerto D, la señal (PORTC3) es pulsada (por el firmware) para que finalmente este sea almacenado en el latch de entrada interno del uC.

Proceso De Escritura. La operación de escritura inicia, con todas las señales en un nivel lógico alto. El firmware del uC selecciona el dispositivo a través del demultiplexor, coloca la señales RD/\overline{WR} (PORTB2) y la señal LeeD en un nivel bajo indicando operación de escritura, (figura 42, línea 4 de la tabla). En este momento el dato almacenado en el latch de salida del puerto D esta presente en las entradas Ax del transceptor y por consiguiente en el bus (ya que la señal esta en un nivel alto). Posteriormente, la señal P toma un nivel bajo durante 10us (figura.42, línea 8 de la tabla), tiempo en el cual el periférico coloca el dato que se encuentra presente en el bus y lo almacena con el flanco de subida de la señal. Finalmente la señales LeeD (PORTB3) y RD/\overline{WR} (PORTB2) retornan a niveles altos consecutivamente.

Figura 42. Circuito de Control de Bus utilizando el PSP del uC



El timer 2 del microcontrolador es utilizado ampliamente para desarrollar una base de tiempo real, gracias a sus características se programo para obtener una interrupción cada 10 μ segundos, con el fin de basar en este, todos los tiempos de escritura y lectura sobre los demás dispositivos.

Comunicación. Uno de los periféricos más utilizados del microcontrolador es el transmisor receptor universal asincrónico (USART), usado para establecer la comunicación con él PC. Existen dos formas de establecer la comunicación serial: la síncrona y la asíncrona.

Cuadro 1: velocidad de baudios para modo asíncrono: muestra el valor que debe ser cargado en el registro SPBRG para determinada velocidad de transmisión.

BRGH = 0				BRGH = 1			
BAUD RATE (K)	Fosc = 20 MHz			BAUD RATE (K)	Fosc = 20 MHz		
	KBAUD	% ERROR	SPBRG value (decimal)		KBAUD	% ERROR	SPBRG value (decimal)
0.3	-	-	-	0.3	-	-	-
1.2	1.221	1.75	255	1.2	-	-	-
2.4	2.404	0.17	129	2.4	-	-	-
9.6	9.766	1.73	31	9.6	9.615	0.16	129
19.2	19.531	1.72	15	19.2	19.231	0.16	64
28.8	31.250	8.51	9	28.8	29.070	0.94	42
33.6	34.722	3.34	8	33.6	33.784	0.55	36
57.6	62.500	8.51	4	57.6	59.524	3.34	20
HIGH	1.221	-	255	HIGH	4.883	-	255
LOW	312.500	-	0	LOW	1250.000	-	0

El USART es también conocido como la interfase de comunicación seriales (SCI), esta puede configurarse como un sistema asincrónico full duplex. Permite también la comunicación multiprocesador usando la detección de direcciones de 9 bits. El BRG es una función interna del uC dedicada para sincronizar los modos de transmisión de la USART. El registro SPBRG controla el período de un timer de ocho bits que se encuentra corriendo libremente. Colocando en un nivel alto la bandera BRGH (bit 6 del registro TXSTA), se indica al procesador si el generador trabajará a alta velocidad. En el cuadro 1 se muestra el valor entero que debe ser cargado en el registro SPBRG para una tasa baudios determinada, de igual forma se muestran el porcentaje de error. Para el caso del prototipo MHCNC la velocidad de baudios es de 9600 colocando en un nivel alto la bandera BRGH y cargando el valor decimal 129 en el registro SPBRG de esta manera obtenemos un error del 0.16 %.

3.1.2. Bloque 2: Almacenamiento. El CI HM6116 (figura 43), es una memoria de acceso aleatorio (RAM) de 16 Kbytes distribuidas en 2048 palabras de 8 bits. Entre sus características mas importantes se destaca su rápido tiempo de acceso (200ns máximo), con entradas y salidas TTL disponible en empaque DIP (dual in Line) o 24 SOIC. La parte alta de las direcciones (páginas) es manejada directamente desde el uC, mientras que los ocho bits inferiores lo realizan a través del CI 74HC374 (figura 44), el cual es un registro de almacenamiento octal tipo D cuyas entradas se encuentran conectadas al bus de datos.

Figura 43. Diagrama esquemático del bloque de almacenamiento.

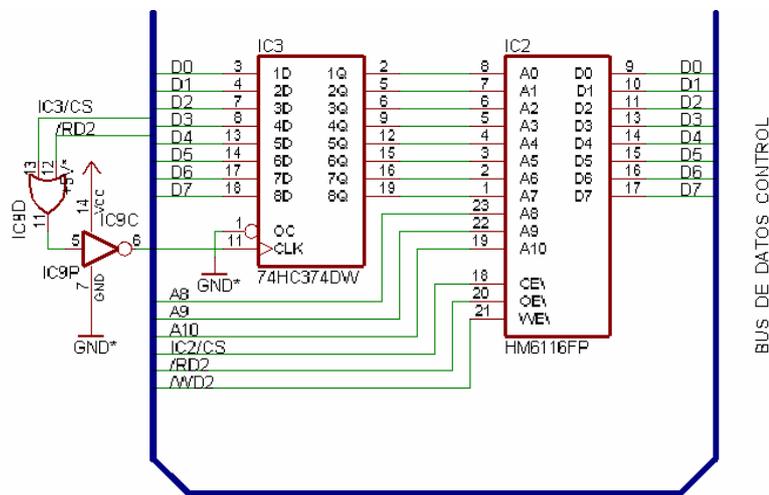
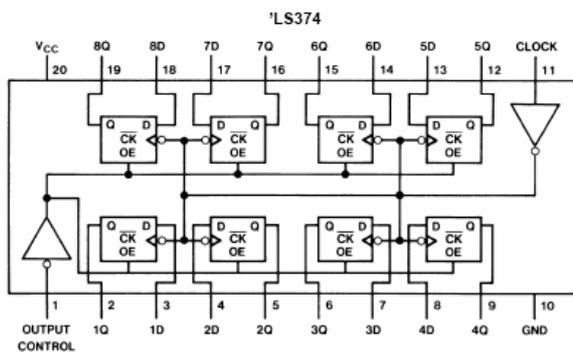


Figura 44. Diagrama de conexiones del 74HC374.

Connection Diagrams

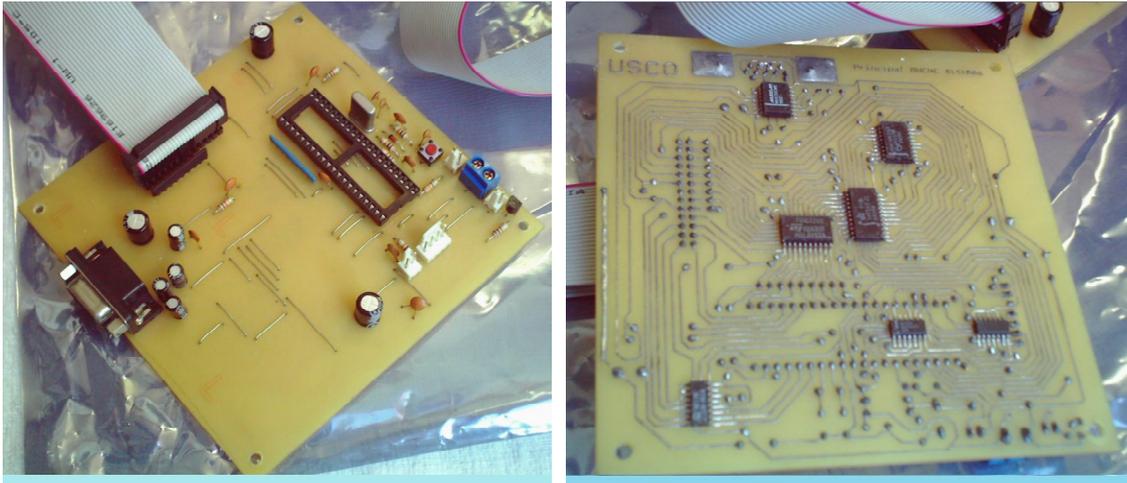


DM54/74LS374

Output Control	Clock	D	Output
L	↑	H	H
L	↑	L	L
L	L	X	Q ₀
H	X	X	Z

www.fairchildsemi.com DM74LS373/DM74LS374 3-STATE Octal D-Type Transparent Latches and Edge-Triggered Flip-Flops data sheet

Figura 45. Fotografía tarjeta principal



3.1.3. Bloque 3,4 Y 5: Coordenadas .Se centra básicamente en el circuito integrado LM629 (figura 46), procesador de control de movimiento. Este dispositivo puede operar a una señal de reloj en el rango de 1MHz a 6 MHz (LM629-6). Posee todas las señales (\overline{CS} , \overline{WR} , \overline{RD}) para establecer una comunicación bidireccional con el uC (procesador anfitrión), a través del bus de ocho bits. Otras señales que son gobernadas directamente por el uC son la señales de RESET, PS (Port Select) y la señal de interrupción que es emitida por el LM629 e indica al procesador anfitrión algún evento que debe ser atendido.

El circuito conformado por las compuertas IC13D e IC13E (figura 46), asociado con el cristal (xtal2) y otros componentes discretos, es un oscilador de relajación que genera una señal de reloj de 4Mhz necesaria para que el LM629 realice todas las operaciones en su interior. La compuerta IC13F es un buffer que entrega la señal de reloj para no cargar el oscilador.

El circuito integrado LM629 proporciona una interfase para el encoder óptico de posición, decodificando las dos señales de cuadratura se obtienen información de la posición y la dirección, figura 47.

Figura 46. Diagrama Esquemático del bloque correspondiente a una coordenada.

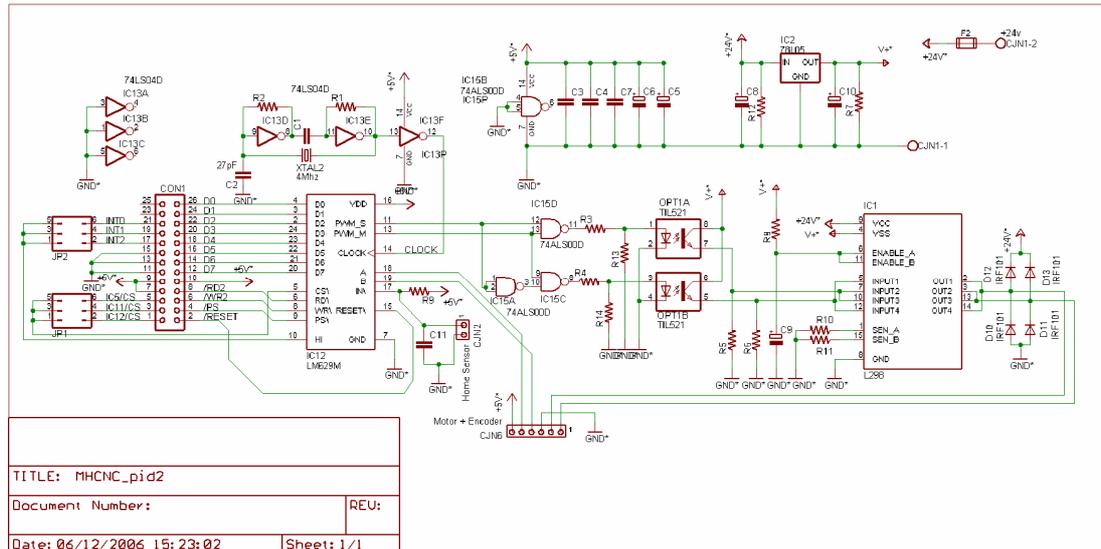
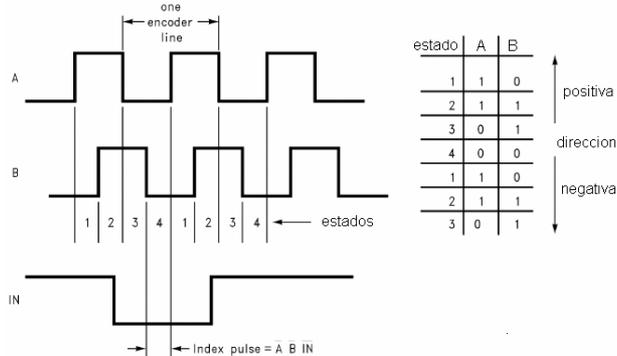


Figura 47. Señal de cuadratura de salida del encoder y su tabla de decodificación. Fotografía cinta codificada, disco codificado con su encoder óptico.



Opcionalmente una tercera señal índice de posición que puede usarse para capturar posición por una revolución. Cada uno de los cuatro estados de la señal de cuadratura de posición son decodificados por el LM2915 entregando un

incremento de 4 veces en la resolución de la posición del número de líneas del encoder. Un encoder con "N" líneas es decodificado con un número de "4N" número de posiciones por el LM629.

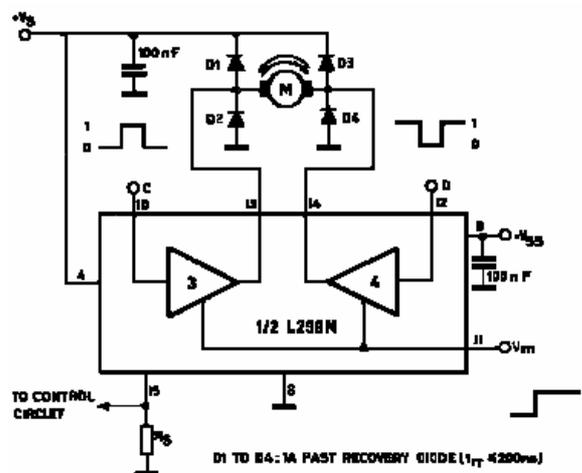
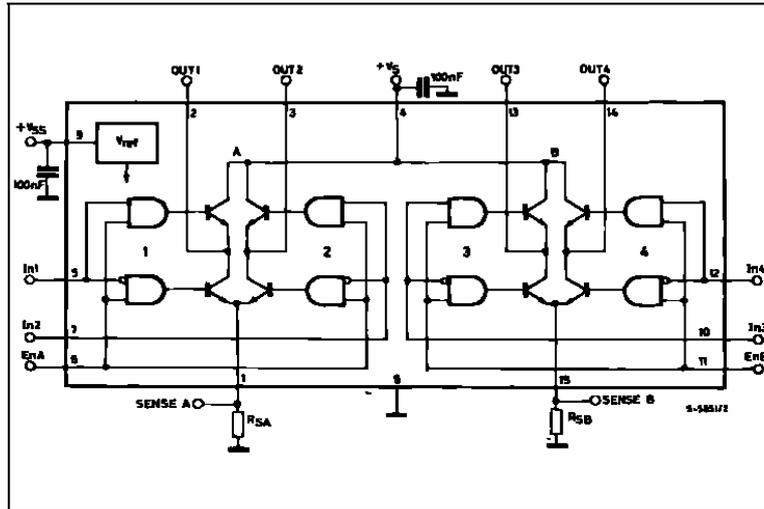
La proporción máxima de la captura del encoder es determinada por el número mínimo de ciclos del reloj que él toma para descifrar cada uno los estados del encoder, figura 47, este número mínimo es 8 ciclos de reloj, la captura del pulso del índice se logra también durante estos 8 ciclos del reloj. Esto da una proporción máxima de la captura del encoder de 750 KHz operando a una frecuencia de 4 MHz. Por ejemplo, usando el motor con un encoder de 500 líneas moviéndose a 30,000 rpm, genera una frecuencia de $[(30.000\text{rpm}/60\text{seg}) \times 500\text{lineas}]/2$ aproximadamente 125KHz.

Manejo De Motores. El dispositivo L298, contiene en un solo empaque cuatro interfases de potencia que operan con un voltaje alto y proporcionan una corriente alta. Diseñado para aceptar niveles lógicos de entrada DTL o TTL y manejar cargas inductivas (como relés, solenoides, motores DC y paso a paso). Para simplificar su uso como dos puentes-H, cada par de canales están provistos con una entrada de habilitación. Una entrada del suministro de energía para el circuito lógico, permite el funcionamiento a un bajo voltaje. La versión L298 maneja 2A por canal.

La figura 48, muestra el control bidireccional de un motor DC, en el se aprecia un sólo un puente-H que se necesita. El puente externo de diodos D1 a D4 es formado por cuatro diodos de recuperación rápida (el trr 200 nseg) que debe escogerse con un voltaje en directo VF (forward Voltaje) tan bajo como sea posible, y en el peor caso igual a la carga. La función del freno (Fast motor stop) requiere que el valor máximo absoluto de 2 A nunca se supere.

Cuando la carga necesita una corriente máxima repetitiva más alta que 2 Amperios, puede escogerse (figura 49) una configuración en paralelo. Esta configuración permite manejar hasta 3 A en operación DC.

Figura 48. Puente H (h-bridge) L298 y su tabla de verdad



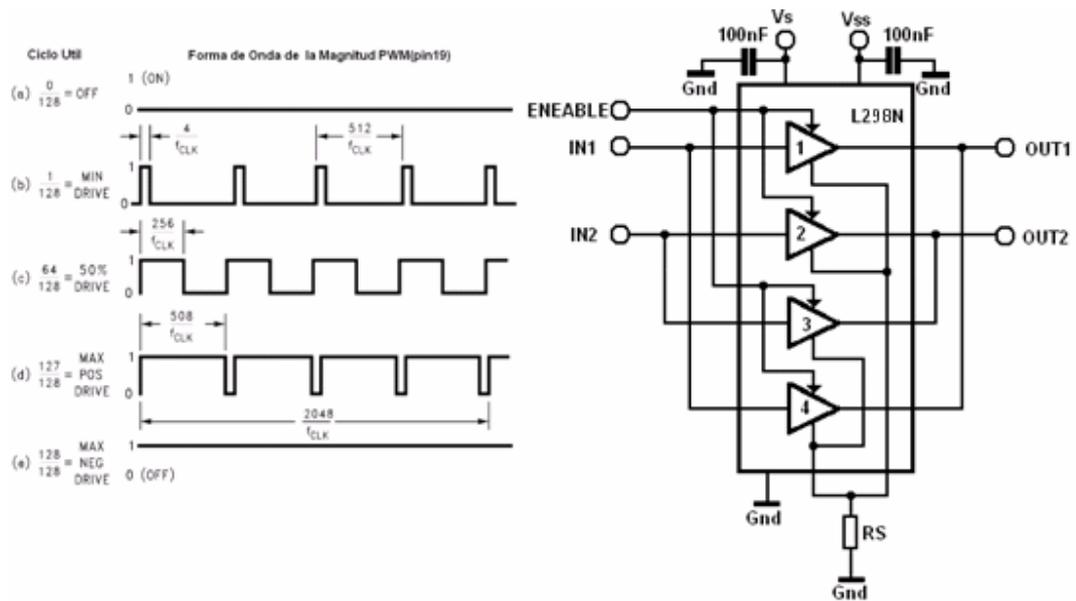
Inputs		Function
$V_{en} = H$	C = H ; D = L	Forward
	C = L ; D = H	Reverse
	C = D	Fast Motor Stop
$V_{en} = L$	C = X ; D = X	Free Running Motor Stop

L = Low H = High X = Don't care

www.st.com L298 DUAL FULL-BRIDGE DRIVER data sheet

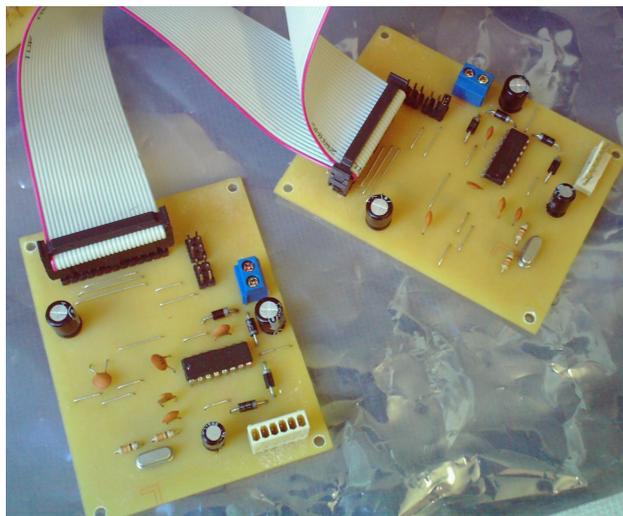
El circuito combinatorio conformado con las compuertas IC15A, IC15E e IC17B (figura 46), permite adaptar las salidas del controlador PID que corresponden al signo y magnitud.

Figura 49. Forma de ondas para la magnitud PWM y configuración en paralelo del L298 que le permite manejar corrientes más altas.



www.st.com L298 DUAL FULL-BRIDGE DRIVER data sheet

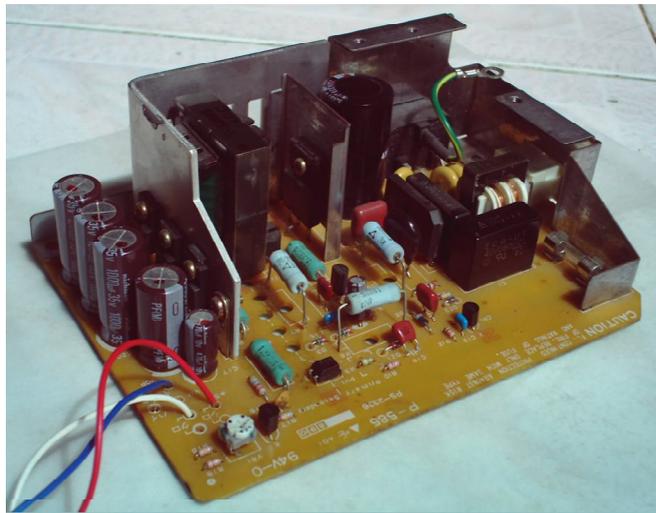
Figura 50. Fotografía de la tarjeta de control PID



3.1.4. Bloque 5: Fuente De Poder Y Control Ac

Fuente De Poder Para la generación de los voltajes de alimentación para los circuitos electrónico se selecciona un fuente genérica conmutada marca SHIJEN referencia QH13-3070 cuyas características se enumeran a continuación

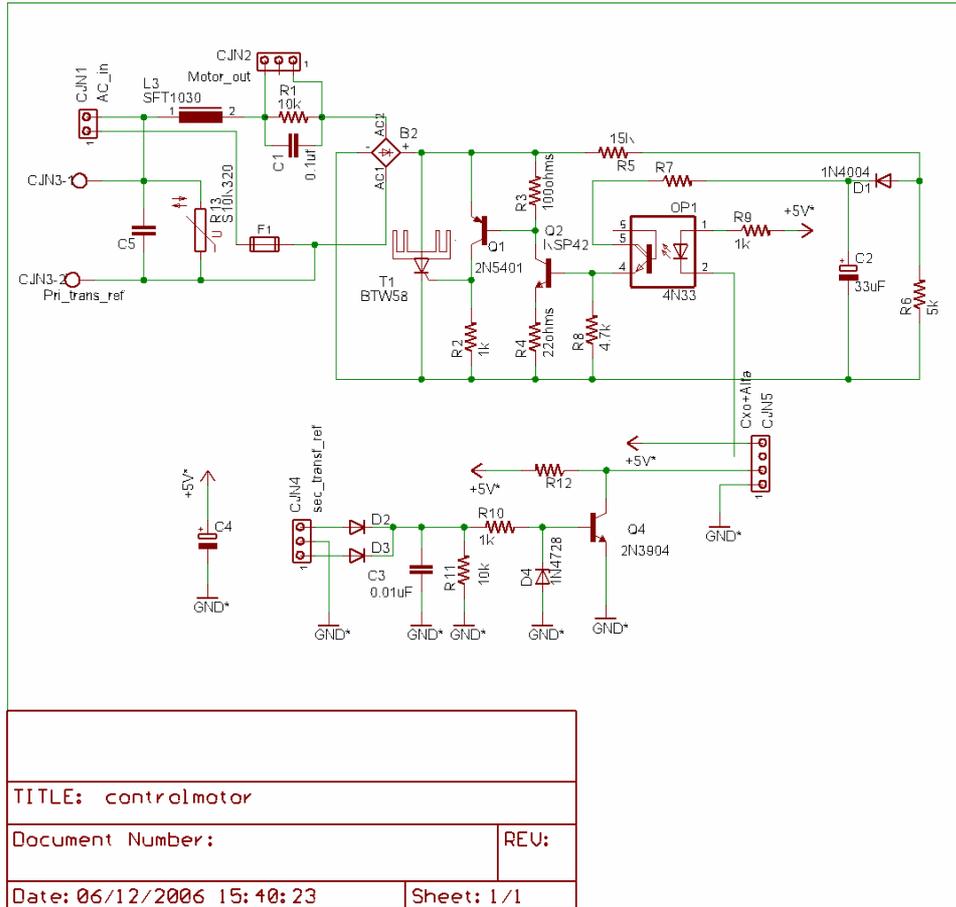
Figura 51. Fuente de Poder



Voltaje de entrada 100Vac-120Vac 50-60Hz
Voltaje de salida +5v I_{max} 1.5A, +24V I_{max} 3A, Vrizado <2%
Frecuencia de conmutación 120Khz
Protección contra cortocircuito

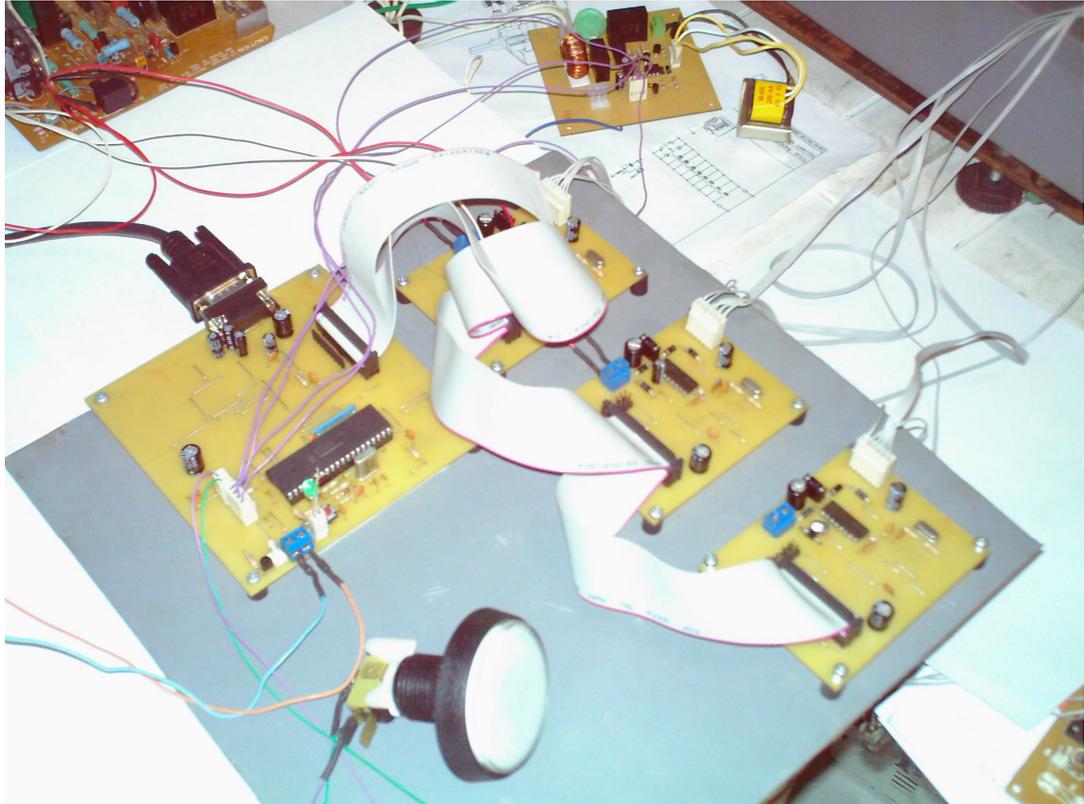
Control AC. El circuito de control del motor AC, es una variación de un semiconvertidor monofásico donde se utiliza un solo SCR para activar el motor AC en el ángulo de disparo seleccionado por el circuito de control. El microcontrolador inicia el conteo (hasta 8.33ms) una vez se ha detectado el cruce por cero que se obtiene rectificación de onda completa de una señal de 6 VAC proveniente del secundario del transformador y que posteriormente es aislada por la acción del transistor Q4.

Figura 52. Esquemático del Control del Motor AC



La acción de C3 y R11 filtran la señal para componentes más altos que la señal de 120Hz para evitar falsos cruces por cero. En el diagrama esquemático se observa que la carga se encuentra en serie con el circuito de control el cual contiene un rectificador de onda completa para proporcionar la polaridad correcta de voltaje al SCR. Los transistores Q1, Q2 junto con los elemento resistivos acoplan la señal de disparo proveniente de del optoacoplador. La red conformada por R5, R6 D1 y C2 proporcionan un voltaje para el disparo que no es superior a las especificaciones máximas de optoacoplador cuando la señal AC se encuentra en el en pico de voltaje.

Figura 53. Fotografía del montaje circuito electrónico



3.2. PROGRAMA DEL MICROCONTROLADOR

El microcontrolador utiliza dos de los cuatro kilobytes de memoria de programa disponibles para desarrollar el programa que le permite manejar todos los periféricos del dispositivo. El proceso de creación del programa parte con el objetivo de permitir una comunicación bidireccional con el computador, este último envía caracteres ASCII que se han coordinado para realizar 26 funciones (definidas y especificadas con letras del alfabeto) con la adición de los caracteres numéricos que representan valores Hexadecimales. El computador inicia un proceso enviando el *código de función* precedido de un signo pesos (\$) correspondiente a la acción que se desea realizar. Si son necesarios parámetros para dicho propósito, estos son enviados seguidamente por parte del computador. Algunas de estas funciones se enumeran a continuación:

Tabla 2. *Códigos de función (Parcial)*

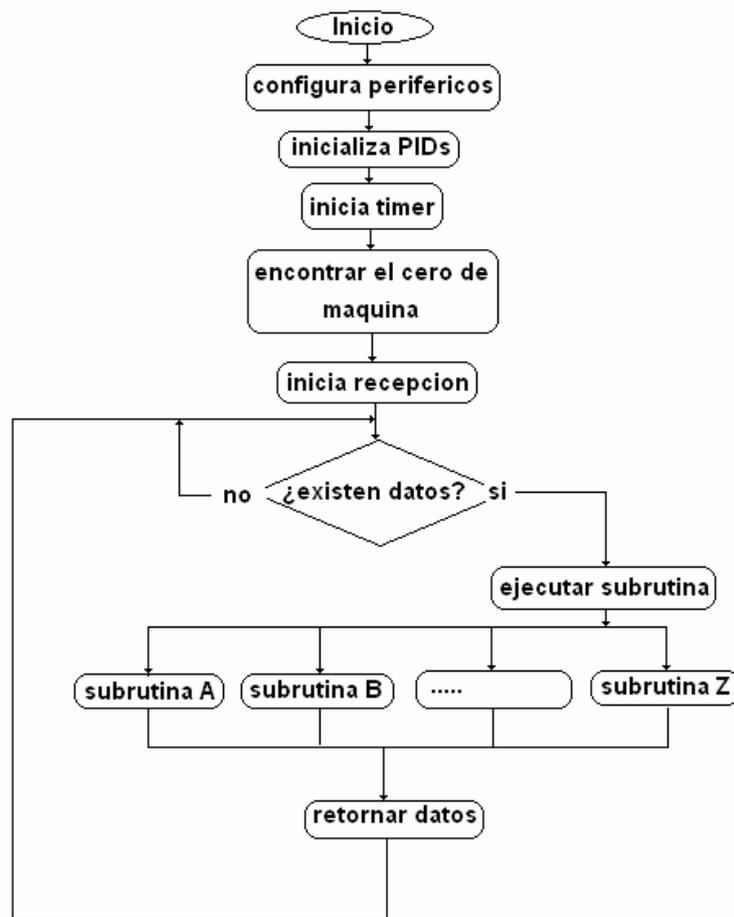
Código de función	Parámetros adicionales	Formato	Descripción
E	No	\$E	Finaliza transmisión y retorne datos.
G	No	\$G	reset
X	2 palabras (32bits)	\$X000007D0	Cargar posición deseada
F	Cuatro parámetros de 1 palabra cada uno (64Bits)	\$F0A00002F 000A025803E8	Cargar parámetros del filtro (Ki,Kp,Kd e iL)
W	1 palabra (16Bits)	\$W07D0	Cargar posición de error
U	No	\$U	Iniciar movimiento

Una vez el microcontrolador detecta que existen datos en el buffer de entrada, los almacena secuencialmente en la memoria RAM monitoreando el envío del código de finalización (\$E), para salta a la rutina especifica de la función. El *código de función* puede estar acompañado de un valor que puede tener hasta una longitud máxima de cuatro palabras (64 Bits) que la rutina requiere, por ello, otra subrutina transforma los datos ASCII que representan un número Hexadecimal a registros de ocho bits según sea la longitud del parámetro. Cuando se envía una palabra como parámetro, este, ocupa 4 bytes en la memoria RAM, que deben ser convertidos a una palabrada 16 bits y almacenado con el fin de que puedan ser procesados.

Las subrutinas especificadas por el *código de función* toman el control del bus transfiriendo y recibiendo información de los procesadores de movimientos LM629, desde o hacia la memoria RAM y que son almacenados en el buffer de transmisión como caracteres ASCII para posteriormente ser enviados hacia el computador.

Una característica importante del modelo de programación, es la realización de rutinas paralelas donde todas son ejecutadas secuencialmente. Se salta de una subrutina a otra cuando esta no ha sido habilitada o se encuentra haciendo un espera (delay), con el objetivo de no utilizar rutinas de retardo, en lugar de ellas, si el programa tiene que realizar una espera, activa uno del los 9 contadores que son gobernados por una base de tiempo exacta a través del Timer2 y marca el lugar de la subrutina donde se realiza la espera para continuar con otras labores mientras finaliza el conteo.

Figura 54. Diagrama de flujo del programa principal



Los controladores de movimiento proporcionan una señal de interrupción que obliga al microcontrolador a saltar al vector de interrupción, donde el programa determina la naturaleza de la interrupción y toma las acciones necesarias. Esta condición puede darse cuando finaliza un movimiento o existe una condición de posición de error.

Una vez el programa es editado y ensamblado por el programa MPLAB¹⁰, el archivo de extensión .HEX es enviado a la memoria Flash de microcontrolador por medio de un programador JEDEC.

¹⁰ Software de MICROCHIP. (www.microchip.com)

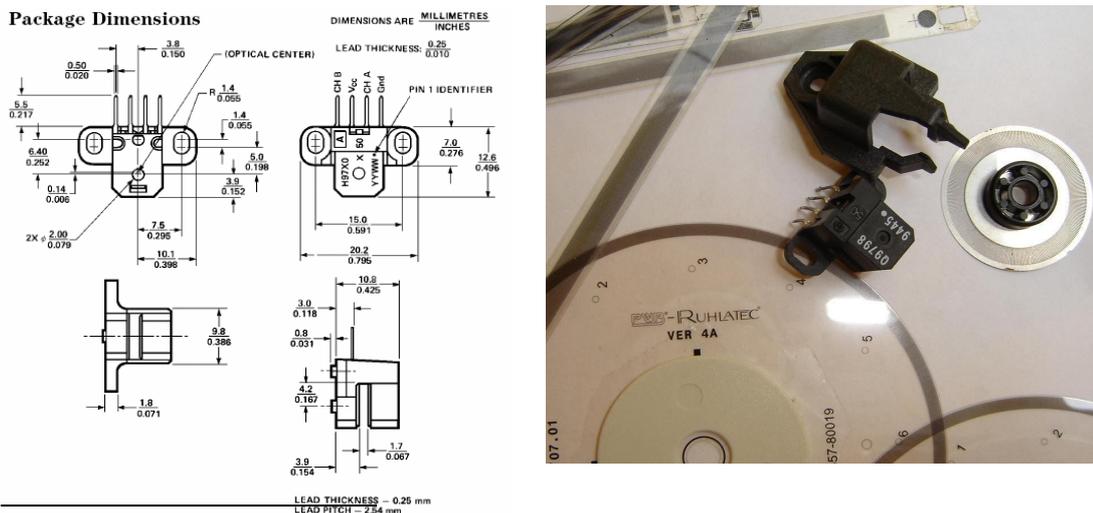
4. SISTEMAS DE CONTROL

4.1. ENCODER¹¹

4.1.1. Descripción La serie de HEDS-9700 es un módulo encoder óptico incremental de alto desempeño y bajo costo. Cuando opera en conjunto con un disco codificado "codewheel" o una correa codificada "codestrip", este módulo detecta la posición rotacional o lineal. El módulo consiste en un LED y un lente como la fuente y un detector IC adjuntó en un empaque plástico en forma de C que obliga a la fuente de luz a enfrentarse favorablemente a un arreglo único de foto-detectores. El módulo es tolerante a montarse desalineadamente.

Las dos salidas digitales de los canales y la entrada del suministro de energía de 5V son accesibles a través de cuatro pines. El HEDS-9700 estándar es diseñado para el uso con un codewheel de 11 mm de radio óptico o un codestrip lineal con una resolución de 150 líneas por pulgada (150 Lpi) y puede operar a una frecuencia máxima de 20 Khz. Existen versiones que pueden operar hasta 30 Khz a una resolución de 560 Lpi. En figura 55, se puede observar que el espacio entre patrón de líneas en el codewheel corresponde al espacio del arreglo de los fotodetectores.

Figura 55: Diagramas de empaque para el modulo encoder y fotografía que muestra al encoder HEDS9798 junto a dos tipos de discos y una correa codificada con resolución de 150 Lpi (líneas por pulgada)



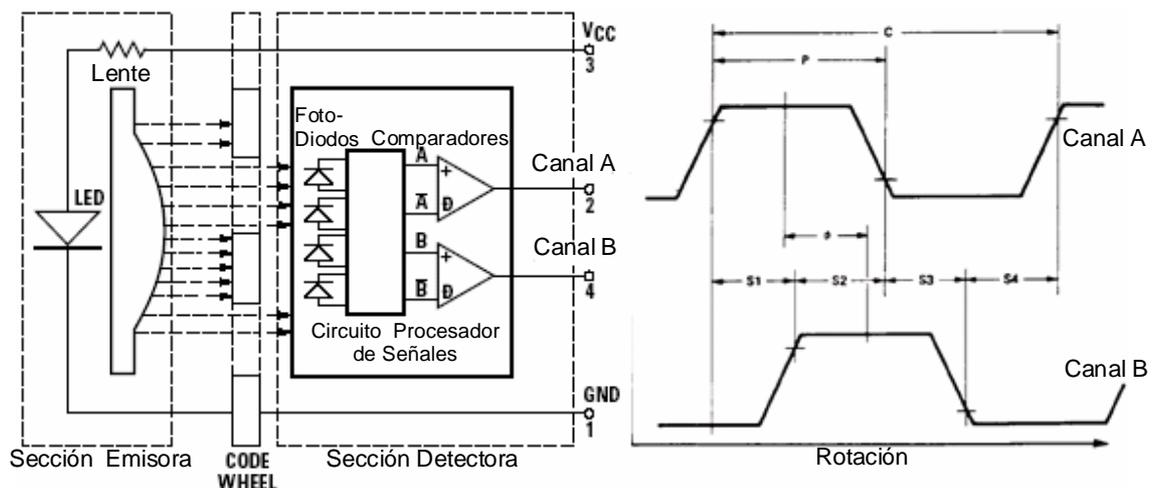
¹¹ www.avagotech.com HEDS9700 series small optical encoder modules technical data Agilent technologies, inc 2004

4.1.2. Teoría de Funcionamiento El HEDS-9700 es un módulo de emisor detector en forma de C o ranurado. Acoplado con un codewheel, traduce el movimiento rotatorio, acoplado con un el codestrip, traduce el movimiento lineal; para ambos casos en una salida digital de dos canales.

Como se ve en el diagrama de bloques (figura 56), el módulo contiene un único diodo emisor (LED) como fuente de luz. Los haces de luz son emitidos paralelamente por medio de un lente simple localizado directamente sobre el LED. En el lado opuesto del emisor se encuentra el circuito integrado detector. Este circuito integrado consiste en múltiples juegos de foto-detectores y el circuito necesario para procesar la señal y producir las formas de onda digitales. El codewheel o el codestrip se mueven entre el emisor y detector, causando una interrupción del haz de luz por la configuración de los espacios y obstrucciones presentes en ellos. Los fotodiodos que detectan estas interrupciones se colocan en un arreglo que corresponde al radio y densidad de líneas del codewheel o el codestrip. Estos detectores son también espaciados de tal forma que el periodo de la luz en un par de detectores corresponde a un período oscuro sobre un par adyacente de detectores. Las salidas de los fotodiodos son transmitidas a la circuitería procesadora de la señal.

Dos comparadores reciben estas señales y producen las salidas finales para los canales A y B. Debido a esta técnica de fases la salida digital del canal A esta en cuadratura con el canal B (90 grados fuera de fase).

Figura 56. Diagrama de bloques del modulo encoder y forma de onda de salida del encoder.



4.2. PROCESADOR DE CONTROL DE MOVIMIENTO LM629¹²

El circuito integrado LM629 es un procesador especializado en el control movimiento, diseñado para el uso con una variedad de motores DC de escobillas, servomotores DC y otros servomecanismos. Proporcionando una señal de cuadratura incremental de posición en realimentación. El dispositivo desarrolla las intensas tareas de cómputo en tiempo real para lograr un control digital de movimiento de alto desempeño. El software de control facilita la interfaz a través de un juego de comandos de alto nivel.

El LM628 tiene una salida de 8 bits que puede manejar un DAC de 8 bits o 12 bits. Los componentes requeridos para construir el sistema del servo se reduce al motor/actuador DC, un encoder incremental. El LM629 proporciona una salida PWM de 8 bits para manejar directamente un driver puente-H. Las partes se fabrican en tecnología NMOS y viene en varias presentaciones (28 pines DIP o 24 pines SOIC). Las versiones están disponibles con los sufijos -6 y -8, respectivamente, designadas para operar a 6 MHz y 8 MHz como frecuencia máxima.

4.2.1. Características

- Registros de aceleración, posición, y velocidad de 32 bits.
- Filtro PID Programable digitalmente con coeficientes de 16 bits.
- Intervalo de muestreo derivativo programable.
- Salida signo y magnitud PWM de 8 bits.
- Generador de perfil de velocidad trapezoidal Interior.
- la Velocidad, posición objetivo, y parámetros del filtro pueden ser cambiados durante el movimiento.
- Funcionamiento en modo de posición o velocidad.
- Interrupciones en tiempo real programables.
- Interfaz de ocho bits con el anfitrión.
- Interfase con encoder incrementales de cuadratura, con entrada para un pulso índice.
- Empaque de montaje de superficie (LM629).

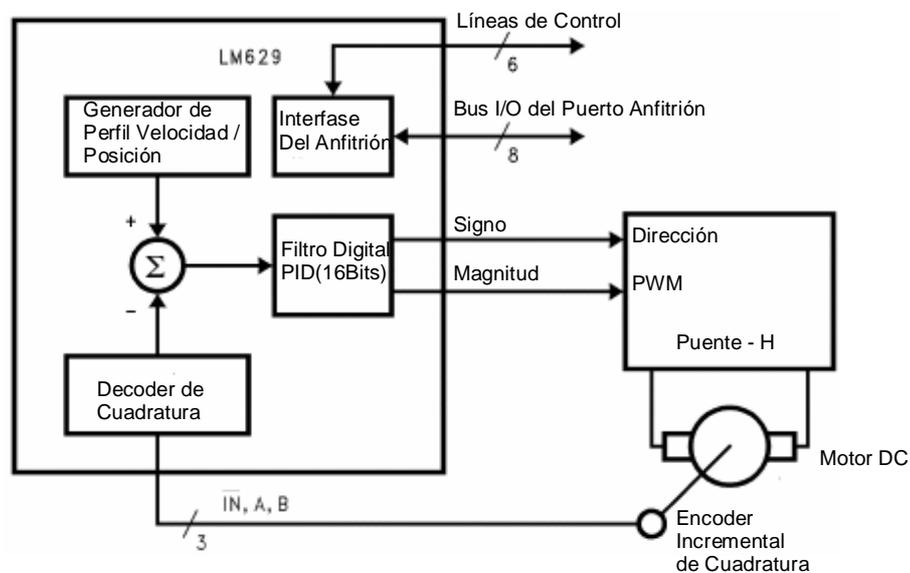
¹² www.national.com LM629/628 Presicion motion controller enero2003

4.2.2. Teoría De Operación

Introducción El diagrama de bloques del sistema típico (figura 57) ilustra el sistema del servo construido usando el LM629. El procesador anfitrión se comunica con el LM629 a través de un puerto de I/O para facilitar la programación del filtro digital de compensación y el perfil de velocidad trapezoidal. Un encoder incremental mantiene el lazo cerrando de la posición del servo.

El generador de perfil de velocidad trapezoidal calcula la trayectoria requerida según sea el modo de funcionamiento (modo de posición o velocidad). En el funcionamiento, del LM629 subtrae la posición real (la posición de realimentación) de la posición deseada (el generador de perfil de posición), y el error de la posición resultante es procesada por el filtro digital para manejar el motor hacia la posición deseada. El cuadro 2, proporciona un resumen breve de especificaciones ofrecido por el LM629.

Figura 57. Sistema de Control de Motor



www.national.com LM629/628 Precision motion controller enero2003

Interfaz De Posición De Realimentación El LM629 obtiene la señal para conducir el motor a través de un encoder incremental. Se proporcionan tres entradas: dos señales de entrada de cuadratura, y una entrada de pulso de índice.

- La salida de pulso de índice optativo proporcionado por algunos encoders asume el estado lógico-bajo, una vez por revolución. Si el LM629 se programa así por el usuario, grabará la posición absoluta del motor en un registro especializado

(el registro índice), en el momento, que todas las tres entradas del encoder se encuentre en un estado lógico bajo. Si el encoder no proporciona la salida del índice, la entrada del índice en el LM629, puede usarse para grabar la posición de HOME (inicio) del motor. En este caso, típicamente el motor cerrará un interruptor que causará un nivel lógico bajo en la entrada del índice optativo. El LM629 grabará la posición del motor en el registro índice y alertará (con una interrupción) al procesador anfitrión.

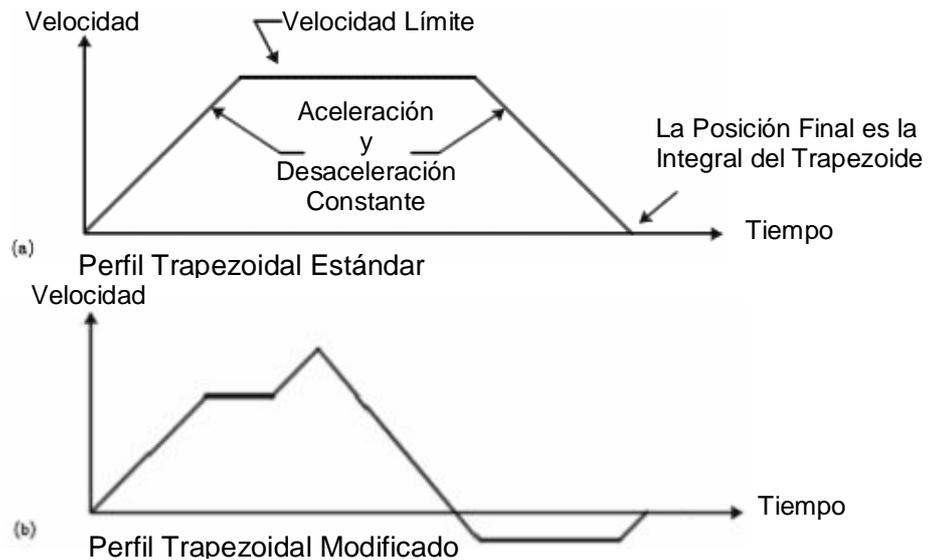
Cuadro 2. Especificaciones del LM629

Rango de Posición	-1,073,741,824 a 1,073,741,823 Cuentas
Rango de Velocidad	0 a 1,073,741,823/2 ¹⁶ Lineas/ Muestra; ej: 0 a 16,383 Lineas/ Muestra, con una Resolución de 1/2 ¹⁶ Lineas/ Muestra.
Rango de Aceleración	0 a 1, 073, 741,823/2 ¹⁶ Lineas/ Muestra / Muestra; ej, 0 a 16,383 Lineas/ Muestra / Muestra, con una Resolución de 1/2 ¹⁶ Lineas/ Muestra / Muestra.
Salida al Driver del Motor	LM629: 8-bit PWM Señales de Signo y Magnitud.
Modos de Operación	Posición y Velocidad
Dispositivo de Realimentación	Encoder Incremental (Señales de Cuadratura; soportadas por un pulso de índice)
Algoritmo de Control	Proporcional Integral Derivativo (más límite de Integración Programable)
Muestras de Intervalo	Término Derivativo: Programable desde 2048/f _{CLK} a (2048 * 256)/f _{CLK} en pasos de 2048/f _{CLK} (256 a 65,536 µs para un reloj de 8.0 MHz). Proporcional e Integral: 2048/f _{CLK}

Generador De Perfil De Velocidad (La Trayectoria)

Operación En Modo De Posición. El generador de perfil de velocidad trapezoidal calcula la posición deseada del motor contra tiempo. En funcionamiento como modo de posición, el procesador anfitrión especifica la aceleración, la velocidad máxima y posición objetivo. El LM629 usa esta información para afectar el movimiento, acelerando hasta alcanzar la velocidad máxima especificada o hasta que la desaceleración empiece a detener el motor en la posición objetivo especificada. La proporción de desaceleración es igual a la proporción de aceleración. En cualquier momento durante el movimiento la velocidad máxima y/o la posición objetivo pueden ser cambiadas, y el motor acelerará o disminuirá la velocidad de acuerdo al cambio.

Figura 58. Ilustra dos perfiles trapezoidales de velocidad típica: trapezoide simple y en la parte inferior un ejemplo de la forma de la trayectoria, cuando se cambian velocidad y posición en dos momentos diferentes durante el movimiento.



www.national.com LM629/628 Presicion motion controller

Operación En Modo De Velocidad. Al operar en el modo de velocidad, el motor acelera a la velocidad especificada a la proporción de aceleración definida, manteniendo esta velocidad especificada hasta que se ejecute el comando para detenerse. La velocidad se mantiene mientras avanza a la posición deseada a una proporción constante. Si hay perturbaciones al movimiento durante el funcionamiento de modo de velocidad, la velocidad media en un periodo de tiempo largo se mantiene constante. Si el motor es incapaz de mantener la velocidad especificada (qué podría causarse por el rotor forzado o frenado, por ejemplo), la posición deseada continuará siendo incrementada, produciendo un error de posición muy grande. Si esta condición no es detectada, y la fuerza que frena el motor se libera, el motor podría alcanzar una velocidad muy alta para ponerse al día con la posición deseada (qué esta todavía avanzando según lo especificado). Esta condición se descubre fácilmente a través de los comandos LPEI y LPES.

Todos los parámetros de trayectoria son valores de 32 bits. La posición es una cantidad con signo. La aceleración y velocidad se especifican como enteros positivos de 16 bits, con 16 bits como fracción. La porción del entero de velocidad especifica cuántas líneas por intervalo de muestreo debe cruzar el motor. La porción fraccionaria designa una cuenta fraccionaria adicional por

contar en el intervalo. Aunque la resolución de la posición del LM629 es limitada a las líneas del entero, las líneas fraccionarias proporcionan un aumento en la medida de la resolución de velocidad. La aceleración se trata de la misma manera. En cada intervalo de muestreo, el valor de aceleración ordenado se agrega a la velocidad actual deseada para obtener una nueva velocidad (a menos que el valor de velocidad máxima se haya alcanzado).

Filtro De Compensación PID¹³ El LM629 usa un filtro digital Integral Proporcional Derivativo (PID) (figura 64), para compensar el lazo de control. El motor se sostiene en la posición deseada aplicando una fuerza restauradora al motor que es proporcional al error de la posición, más el error integral, más el del error derivado. La ecuación del tiempo discreto siguiente, ilustra el control desarrollado por el LM629:

$$U(n) = Kp * e(n) + Ki \sum_{N=0}^n e(n) + Kd[e(n') - e(n'-1)]$$

Donde $u(n)$ es la señal de salida de control del motor en un tiempo de muestreo n , $e(n)$ es el error de posición en el momento de la muestra n , n' indica una muestra del periodo de muestreo derivativo, y los coeficientes k_p , k_i , y k_d son los parámetros de filtro de tiempo discreto cargados por el usuario.

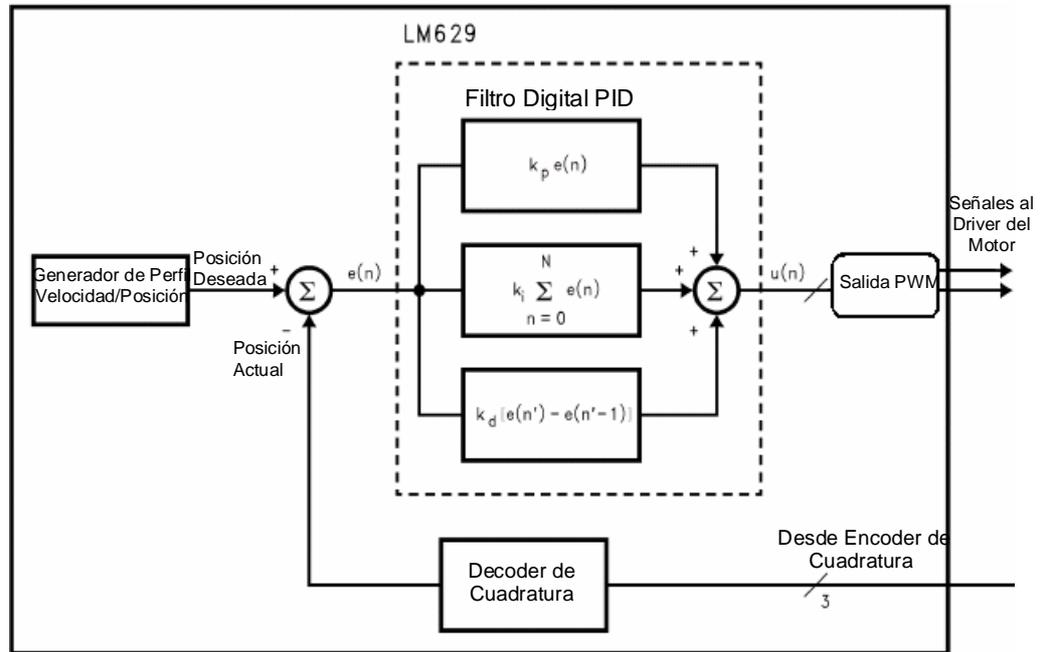
Término Proporcional K_p . El término proporcional, suministra una fuerza restauradora proporcional a la posición de error, así como hace un resorte obedeciendo la ley de Hooke.

Término de Integración K_i . El término de integración, provee una fuerza restauradora que crece con el tiempo, asegurando así que el error de posición estático sea cero. Si hay un Torque constante cargando, el motor todavía podrá lograr un error de posición cero.

Término Derivativo K_d . El término derivativo, aporta una fuerza proporcional a la razón de cambio de la posición de error. Simplemente actúa como el amortiguamiento viscoso en un sistema amortiguado de masa-resorte (como un amortiguador en un automóvil).

¹³ LM628 Programming Guide. National Semiconductor, Application Note 693; Stevent Hunt, January 1999. pág. 18.

Figura 59. Diagrama de Bloques Simplificado del Filtro PID LM629



www.national.com LM629/ LM628 Programming Guide. Nacional Semiconductor, Application Note 693; Stevent Hunt, January 1999. pág. 1

El intervalo de muestreo asociado con el término derivativo es seleccionable por el usuario; esto le permite al LM629 la capacidad de controlar un rango más ancho de cargas inerciales, proporcionando una buena aproximación al derivador continuo. En general, intervalos de muestreo de mucho tiempo son útiles para la baja velocidad.

Durante su funcionamiento (figura 59), el algoritmo del filtro recibe una señal de error de 16 bits desde el punto de suma del lazo. La señal de error es completamente representada por los 16 bits para asegurar una conducta predecible. Además de ser multiplicado por el coeficiente de filtro k_p , la señal del error se agrega a una acumulación de errores anteriores (para formar la señal integral) y, a una proporción determinada por el intervalo de muestreo derivativo escogido, el error anterior es sustraído del error actual (para formar la señal derivativa). Todas las multiplicaciones del filtro son operaciones con 16 bits únicamente y solo son usados los 16 bits LBS del producto.

La señal integral se mantiene a 24 bits, pero sólo los 16 bits más significativos son usados. Esta técnica escalar tiene como efecto un mayor rango utilizable de valores del coeficiente k_i . Los 16 bits son desplazados hacia la derecha ocho

posiciones y multiplicados por el coeficiente k_i para formar el término que contribuye al control del rendimiento del motor. La magnitud absoluta de este producto es comparado al coeficiente del límite de Integración, y la menor, magnitud apropiadamente asignada contribuye entonces a la señal de control de motor.

En cada intervalo de muestreo derivativo, la señal derivativa se multiplica por el coeficiente k_d . Este producto contribuye al rendimiento del control del motor en cada intervalo de muestreo, independientemente del intervalo de muestreo derivativo escogido por el usuario. Se suman los términos k_p , k_i , y k_d para formar una cantidad de 16 bits. Dependiendo del modo de salida, los 8 o 12 bits más significativos son entregados para la señal de control del motor.

Sintonización del PID¹⁴ Antes de iniciar la programación o puesta a punto del PID, es necesario determinar la polaridad correcta del motor de forma que maneje el sentido de giro apropiado. Si la polaridad es incorrecta, la señal que excita al motor empujará al mecanismo asociado al eje del motor lejos de su posición deseada, en lugar de hacia él. Esto resulta en “la huida de motor”, una condición que se caracteriza por el corrimiento del motor continuamente a alta velocidad.

Especificaciones de la respuesta transitoria. En muchos casos prácticos, las características de desempeño deseadas de los sistemas de control, sean en el tiempo continuo o en el tiempo discreto, se especifican en términos de cantidades en el dominio de tiempo. Esto ocurre debido a que los sistemas con almacenamiento de energía no pueden responder en forma instantánea y siempre que estén sujetos a entradas o perturbaciones mostrarán alguna respuesta transitoria.

Con frecuencia, las características de desempeño de un sistema de control están especificadas en términos de su respuesta transitoria ante una entrada escalón unitario, ya que la entrada escalón unitario es fácil de generar y es lo suficientemente drástica para proporcionar información útil tanto en las características de la respuesta transitoria como de la respuesta del estado permanente del sistema.

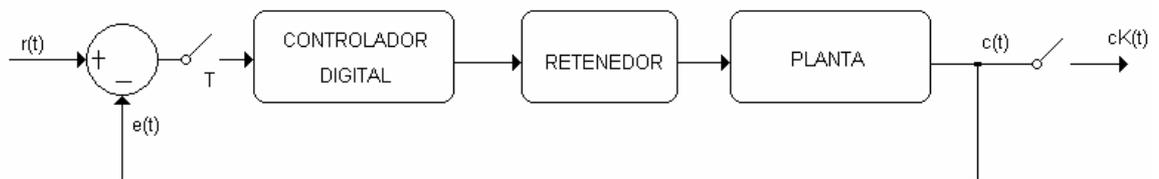
La respuesta transitoria de un sistema a una entrada escalón unitario depende de las condiciones iniciales. Por comodidad, al comparar respuesta transitoria de varios sistemas, es común utilizar las condiciones iniciales estándar: el sistema está inicialmente en reposo y la salida, y ; todas sus derivadas con respecto al

¹⁴ Sistemas de Control en Tiempo Discreto, segunda Edición, Katshito Ogata. Sección 4-4 Análisis de la Respuesta Transitoria y en Estado Permanente, pág.193.

tiempo son cero. Las características de respuesta pueden entonces compararse sin dificultad.

La respuesta transitoria de un sistema de control práctico, donde la señal de salida es en tiempo continuo, ha menudo muestra oscilaciones amortiguadas antes de llegar al estado permanente. (Esto es cierto para la mayoría de los sistemas de control en tiempo discreto o digitales, porque las plantas a controlarse en la mayor parte de los casos son en tiempo continuo y, por lo tanto, la señales de salida son en tiempo continuo.)

Figura 60. Sistema de control digital



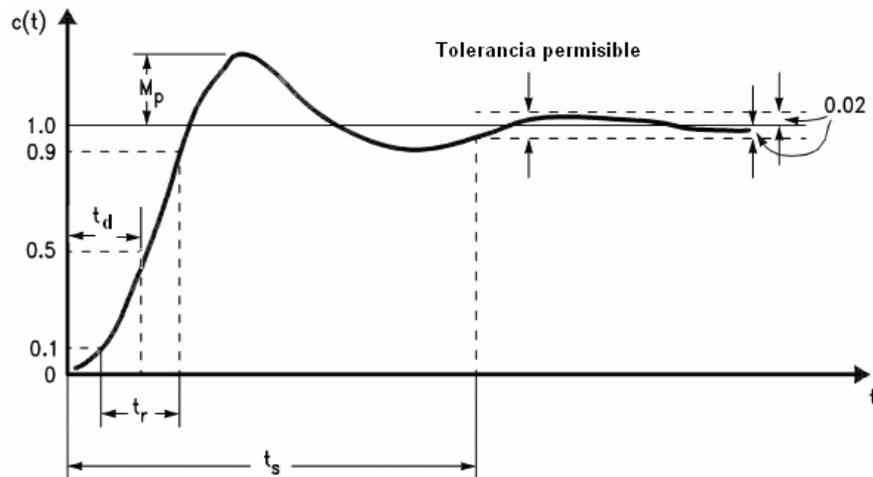
Por ejemplo, el sistema de control digital que se muestra en la figura 65. La salida de un sistema como éste ante una entrada escalón unitario puede mostrar oscilaciones amortiguadas, tal y como se muestran en la figura 60

Igual que en el caso de los sistemas de control en tiempo continuo, la respuesta transitoria de un sistema de control digital puede caracterizarse no sólo por el factor de amortiguamiento relativo y la frecuencia natural amortiguada, sino también por el tiempo de levantamiento, el sobreimpulso máximo y el tiempo de asentamiento en respuesta a una entrada escalón. De hecho, en las especificaciones de distintas características de la respuesta transitoria, es común especificar las siguientes cantidades:

- Tiempo de retardo t_d
- Tiempo de levantamiento t_r
- Tiempo pico t_p
- Sobre pasos máximos M_p
- Tiempo de asentamiento t_s

Las especificaciones de la respuesta transitoria antes mencionadas en la respuesta escalón unitario son definidas a continuación y aparecen en forma gráfica en la figura 61

Figura 61. Especificaciones de la respuesta transitoria.



Tiempo de retardo t_d . El tiempo de retardo es el tiempo requerido para que la respuesta llegue a la mitad del valor final por primera vez.

Tiempo de levantamiento t_r . El tiempo de levantamiento es el tiempo que requiere la respuesta para pasar de un 10% hasta el 90%, de 5% a 95% o de 0% al 100% de su valor final, según la situación. Para sistemas de segundo orden subamortiguados se utiliza un tiempo de levantamiento de 0% al 100%. Para sistemas sobre amortiguados y sistema con atraso de transporte, comúnmente se utiliza el tiempo de levantamiento del 10 al 90%.

Tiempo pico t_p . El tiempo pico es el tiempo requerido para que la respuesta llegué a la primera cresta del sobreimpulso.

Sobreimpulso máximo M_p . El sobreimpulso máximo es el valor máximo de la curva de respuesta medido a partir de la unidad. Si el valor final en estado permanente de la respuesta, difiere de la unidad; entonces es común utilizar el sobreimpulso porcentual máximo. Queda definido por la relación:

$$M_p(\%) = \frac{c(t_p) - c(\infty)}{c(\infty)} \times 100\%$$

La cantidad de sobreimpulso máximo (en porcentaje) indica en forma directa la estabilidad relativa del sistema.

Tiempo de asentamiento t_s . El tiempo de asentamiento es el tiempo requerido para que la curva de respuesta llegue y se quede dentro de un rango alrededor del valor final de un tamaño especificado, en función de un porcentaje absoluto del valor final, por lo general 2%. El tiempo de asentamiento está relacionado con la constante tiempo de mayor valor en el sistema de control.

Las especificaciones en el dominio del tiempo mostradas anteriormente son muy importantes, ya que en la mayor parte de los sistemas de control son sistemas en el dominio del tiempo; esto es, deben mostrar respuestas aceptables en el tiempo. (Lo que significa que el sistema de control bajo diseño deberá ser modificado hasta que la respuesta transitoria sea satisfactoria.)

No todas las especificaciones que se mencionaron anteriormente se aplican necesariamente a un caso dado. Por ejemplo en el caso de un sistema sobre amortiguado, no son aplicable los términos correspondientes al tiempo pico y el sobreimpulso máximo.

Procedimiento¹⁵ El LM629 es un controlador PID digital. El lazo del filtro de compensación del controlador de PID es normalmente sintonizado experimentalmente, sobre todo si la dinámica del sistema no es bien conocida o definida.

La respuesta del filtro de PID, está condicionada por cinco variables que incluyen la suma de los tres coeficientes de ganancia (el k_p , el k_i , y el k_d), el coeficiente de límite de integración (el i_l), y el intervalo de muestreo derivativo (d_s). Sintonizar el filtro equivale a determinar los valores para estos coeficientes de modo que el sistema sea críticamente amortiguado proporcionando un seguimiento y un tiempo de establecimiento óptimo.

La mejor manera de determinar los coeficientes del filtro se realiza de forma experimental en dos pasos.

- En el primer paso, los valores de k_p , k_i , y k_d (junto con el i_l y el d_s) son sistemáticamente ajustados hasta obtener una buena respuesta. Los métodos visuales y manuales son usados para evaluar el efecto de cada coeficiente en el comportamiento del sistema.
- En el segundo paso, el trazado de la respuesta ante una entrada escalón proporciona la información sobre el amortiguamiento del sistema, y los

¹⁵ LM628 Programming Guide. National Semiconductor, Application Note 693; Stevent Hunt, January 1999. pág. 18.

coeficientes del filtro, determinados en el paso uno, se modifican de una forma mas fina para obtener un sistema críticamente amortiguado.

Paso 1: Método Visual

Se inicializo el sistema, definiendo los parámetros del filtro con los valores de k_p , k_i , $e_{il} = 0$, $k_d = 2$, y $d_s = 1$. Seguidamente, se programaron la posición deseada y posición actual igual a cero, al ejecutar el comando STT (STarT), el lazo de control intento sostener el eje del motor en la posición actual. El eje del motor se mantuvo estacionario ya que la posición actual como la deseada, eran iguales. Cualquier desplazamiento del eje del motor constituía un error de la posición, pero con k_p y k_i puestos a cero, el lazo de control no puedo corregir este error.

Los coeficientes k_d y d_s fueron determinados con un proceso iterativo. El coeficiente se aumento sistemáticamente hasta que el eje del motor empiezo a oscilar a frecuencias altas. El coeficiente d_s se aumento de uno en uno. El proceso entero se repite hasta que el d_s alcanza un valor apropiado para el sistema. También se selecciono k_d y d_s para que el producto $k_d \times d_s$ sea el mas grande posible manteniendo aceptablemente bajas las vibraciones del motor.

Inicializando $k_d = 2$ y doblándolo para una posterior prueba, se observo que con cada aumento de k_d , la resistencia del puntero se incremento al tratar de desplazarlo de su posición. Entre mas lento se desplazaba el eje del motor mas flojo se sentía, porque el k_d suministra una fuerza proporcional a la tasa de cambio de la posición de error.

El coeficiente k_p se determino de igual forma que el coeficiente k_d , el valor de k_p se aumento, y se evaluó el amortiguamiento del sistema. Este proceso se repitió hasta que el sistema fue críticamente amortiguado. Se observo que con cada aumento del término k_p , daba como consecuencia aumentos de la rigidez del eje del motor. El eje del motor se sentirá como un resorte, si es forzado fuera de su posición deseada y es soltado, el eje del motor será "halado" hacia la posición deseada. Si k_p se seleccionaba excesivamente bajo, el sistema era sobre amortiguado y el eje del motor se recuperaba demasiado lento. Si por el contrario el k_p era excesivamente grande, el sistema era subamortiguado, y el eje del motor se recuperaba demasiado rápido. El coeficiente de ganancia proporcional k_p , se aumento al valor más grande sin que este causara un excesivo sobre impulso u oscilación, en este punto el sistema es críticamente amortiguado, y por consiguiente proporciona un seguimiento óptimo y un buen tiempo de establecimiento.

La fuerza correctiva proporcionada por el término integral aumenta linealmente con el tiempo. En modo posición esta fuerza corrige el error de posición en estado estacionario. Los valores altos de k_i proporcionan la compensación de error de posición de estado estacionario rápidamente, dando como consecuencia el sobre

impulso y las oscilaciones. En general, se selecciono el valor del término k_i del valor más pequeño que proporcione un compromiso apropiado entre tres características del sistema: sobre impulso, tiempo de asentamiento y el tiempo para cancelar los efectos posición de error de estado estacionario

Resultados: de acuerdo al los procesos descritos anteriormente para la sintonización del filtro PID, Los valores de los coeficientes obtenidos para cada coordenada son:

Paso1:

Coordenada X

$K_i = 5$

$K_p = 30$

$K_d = 4000$

$i_l = 1000$

$d_s = 10$ equivalente a $10 \times 2048 \times (1/4 \text{Mhz}) = 5,12 \text{mseg.}$

Coordenada Y

$K_i = 5$

$K_p = 20$

$K_d = 4000$

$i_l = 1000$

$d_s = 10$ equivalente a $10 \times 2048 \times (1/4 \text{Mhz}) = 5,12 \text{mseg.}$

Coordenada Z

$K_i = 10$

$K_p = 40$

$K_d = 6000$

$i_l = 1000$

$d_s = 10$ equivalente a $10 \times 2048 \times (1/4 \text{Mhz}) = 5,12 \text{mseg.}$

Paso 2: Respuesta al paso

Por medio de la visualización de la respuesta al paso se modifican los coeficientes del filtro para obtener una respuesta críticamente amortiguada de una forma mas fina. Para tal objetivo se desarrolla un programa (LabView) que permite evaluar el comportamiento de cada coordenada al ejecutar un movimiento repetitivo con la máxima aceleración y velocidad del sistema (entrada escalón de posición).

Figura 63 Respuesta al paso con parámetros de filtro para la coordenada X $k_p=100$, $k_i=10$, $k_d=4000$, $i_l=1000$

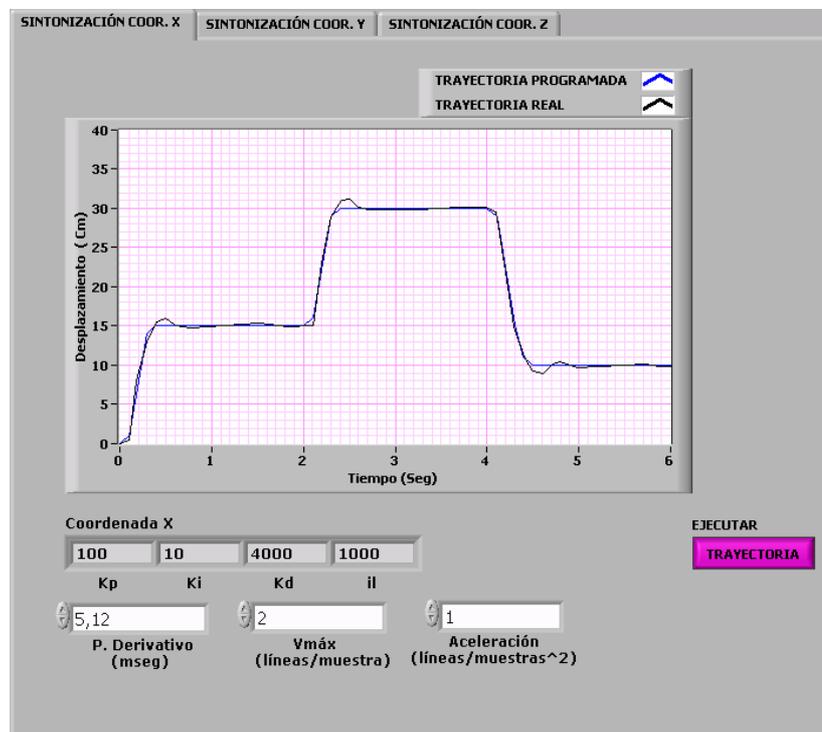


Figura 64 Respuesta al paso con parámetros de filtro para la coordenada X $k_p=35$, $k_i=5$, $k_d=4000$, $i_l=1000$

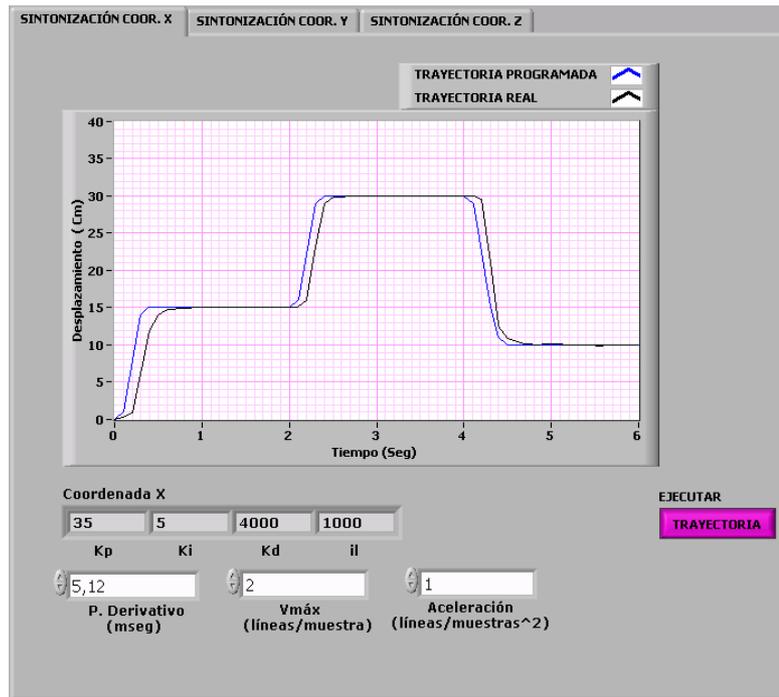


Figura 65 Respuesta al paso con parámetros de filtro para la coordenada X $k_p=100$, $k_i=50$, $k_d=4000$, $i_l=1000$

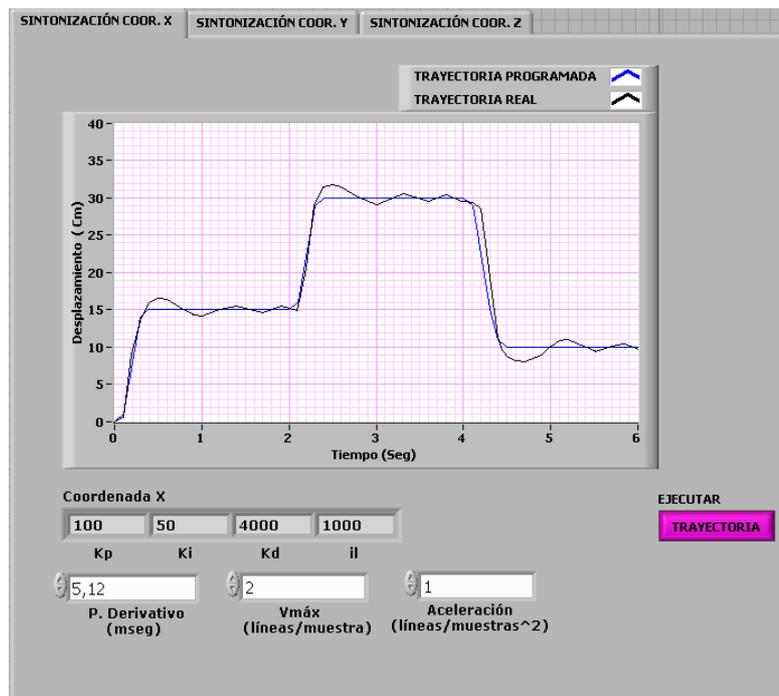
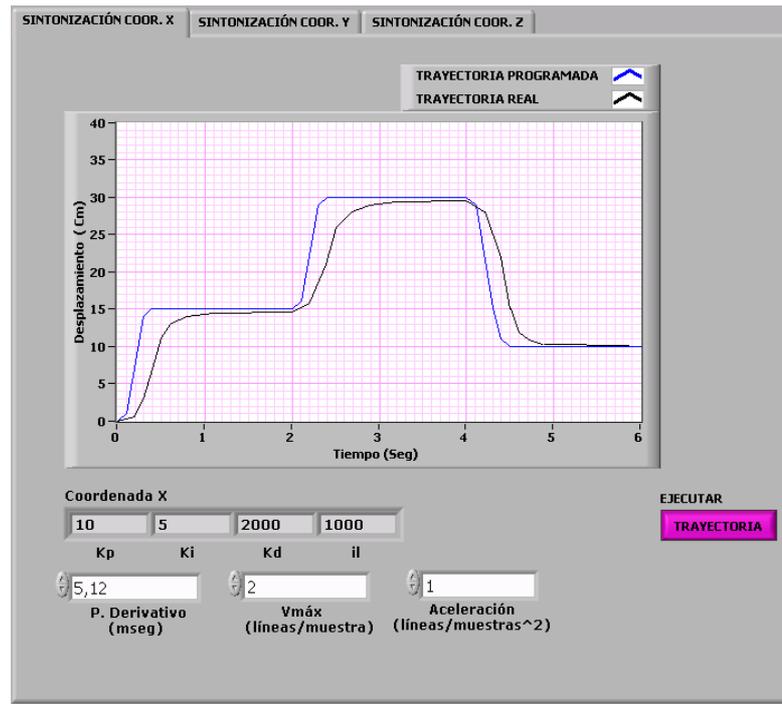


Figura 66 Respuesta al paso con parámetros de filtro para la coordenada X $k_p=10$, $k_i=5$, $k_d=2000$, $i_l=1000$



Debido a las limitaciones en la transmisión de datos por el puerto serie, el periodo de muestreo para graficar los datos (diferente al del control PID $2048 \times (1/4\text{Mhz}) = 512\mu\text{s}$) es de 100ms. Sin embargo las 10 muestras en un segundo obtenidas permiten graficar y observar la respuesta ante la entrada escalón, para realizar las correcciones necesarias a los parámetros del filtro.

En los gráficos se observa que la entrada escalón unitario no es ideal ya que es producida por el generador de trayectoria del PID. Puesto que en el filtro no es posible cargar una aceleración mayor a la velocidad máx, es decir esta debe ser máx a la mitad del valor de la velocidad, no es posible que la trayectoria del escalón unitario sea completamente lineal. Además de ello se aprecia que para valores grandes de K_i se produce una oscilación sostenida en la región de estado estacionario o permanente, que se observa funcionalmente como un movimiento continuo de la coordenada al tratar de alcanzar la posición deseada. Un valor pequeño de K_p , hace que el sistema de posicionamiento de la máquina no pueda seguir completamente la trayectoria deseada.

Figura 67 Respuesta al paso con parámetros de filtro para la coordenada Y $k_p=50$, $k_i=10$, $k_d=4000$, $i_l=1000$

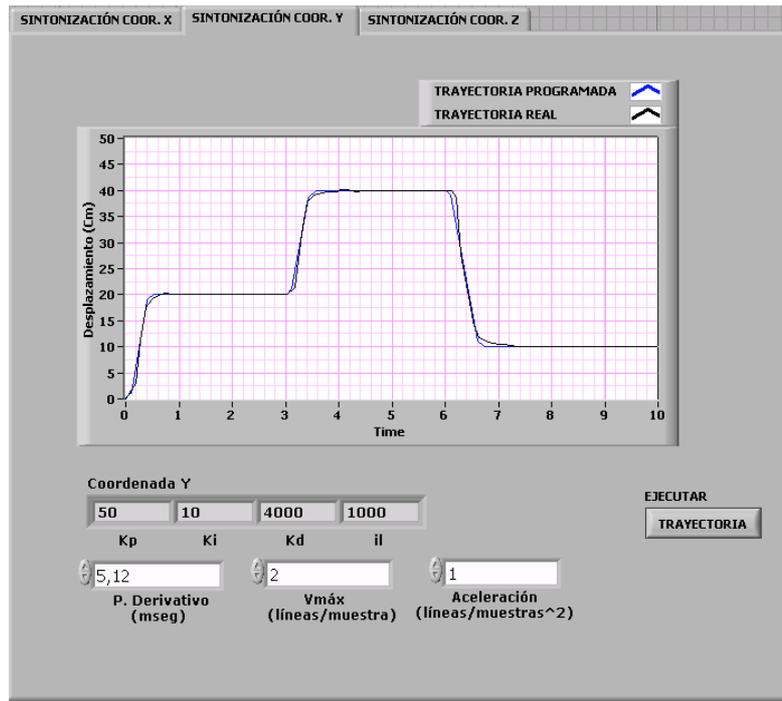
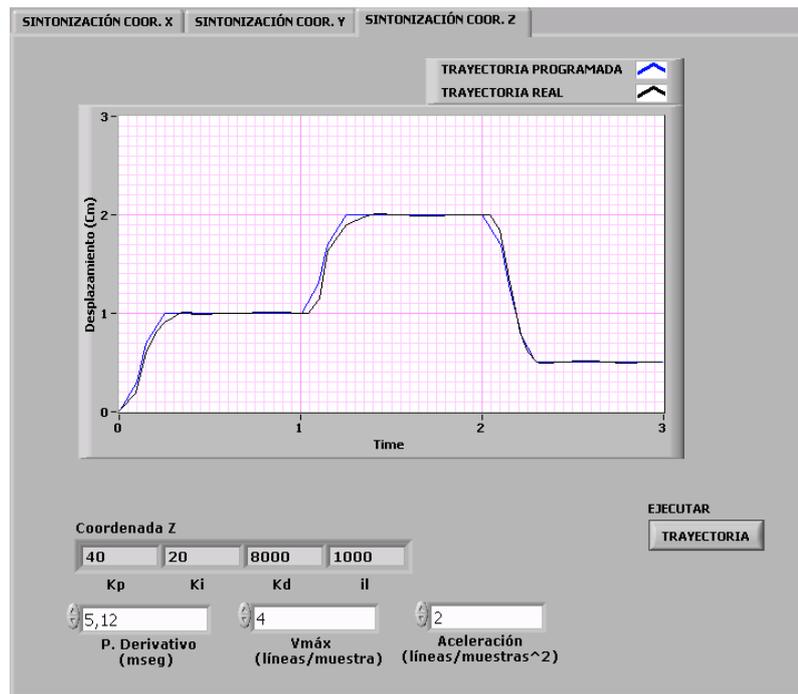


Figura 68 Respuesta al paso con parámetros de filtro para la coordenada Y $k_p=40$, $k_i=20$, $k_d=8000$, $i_l=1000$



5. INTERFAZ GRÁFICA

5.1. INTRODUCCIÓN

En un entorno industrial, la tendencia actual es que sea un software especializado quien se encargue de la comunicación con el control del sistema (Interfaz Electrónica), coordinando el funcionamiento de algunos elementos. Uno de estos programas es LabView de la multinacional National Instruments. Labview permite recoger, analizar y monitorizar los datos dentro de un entorno de programación gráfico en el que se ensamblan objetos llamados instrumentos virtuales (VI's) para formar el programa de aplicación con el que interactuará el usuario y que se denomina instrumento virtual.

La capacidad de comunicación con otros sistemas es una cualidad importante en cualquier equipo ó sistema, en este caso mediante la interfaz RS-232.

Mediante la representación en pantalla de los elementos gráficos de visualización y control que servirán de interfaz con el usuario, este observará los estados de las entradas seleccionadas en la pantalla e interactuará con las salidas directamente o mediante la ejecución de las rutinas programadas.

5.2. ARQUITECTURA DE LA INTERFAZ GRÁFICA

El esquema general de la cadena de adquisición y manejo de datos es el siguiente:

Figura 69. Diagrama de Bloques de la Interfaz Gráfica



Este esquema general integra el proceso CAD/CAM de Circuitos Impresos, mediante la adquisición, acondicionamiento y procesamiento de los datos generados por cada proceso.

5.3. ESTRUCTURA DEL SOFTWARE

El software se encarga de comunicar la interfaz de usuario del computador con el hardware de adquisición y ejecución de datos, dotando a la aplicación de la funcionalidad requerida.

5.3.1. Captación, acondicionamiento y presentación Este proceso se realiza con dos tipos de datos:

- Archivos CAM.
- Datos funcionales de la Máquina Herramienta.

Archivos CAM El software CAD Eagle implementado para el diseño de circuitos impresos. Además de posicionamiento automático de los componentes, señalizando errores, auto enrutador, etc.), permite la creación de archivos para la manufactura de circuitos impresos.

Estos archivos son generados a partir del software Eagle mediante *el Programa de Idioma de Usuario y el Procesador CAM*.

Los datos generados durante el proceso diseño del circuito impreso, Dibujo y captura del esquema, Colocación de los componentes y enrutamiento de las pistas, son implementados durante la programación del Procesador CAM del software Eagle, que genera el archivo de Códigos Numéricos necesario para la llevar a cabo el proceso de manufactura, junto con dos archivos adicionales que entregan la información complementaria a la máquina herramienta. La Interfaz Gráfica se encarga del acceso y acondicionamiento de estos archivos.

Características de los Archivo CAM. Los archivos son generados con las características de los equipos de manufactura para perforación de circuitos impresos con formato de comandos "Sieb & Meyer" SM1000 (Anexo C). Estos deben ser guardados en el mismo directorio (o carpeta). Además de compartir el nombre del archivo *.DRL, es decir, el archivo DRI y TXT (Códigos Numéricos) deben guardarse bajo el mismo nombre para procesos de adquisición y acondicionamiento posteriores. El archivo CAM *.DRL, contiene información concerniente al número y diámetro de las brocas generadas a partir del proceso de diseño del circuito impreso. Esta información podrá visualizarse en el panel frontal de la interfaz sin sufrir ninguna modificación.

Figura 70. Información contenida en un archivo *.DRL



Los archivos acondicionados a partir del Procesador CAM son:

- Archivo DRI “Drill Station Info File”. En este archivo se presenta un informe detallado de los parámetros programados en el procesador CAM Eagle, para el establecimiento del trabajo de manufactura en la estación de taladro. (Figura 71)
- Archivo TXT “Plotfile”. Contiene el conjunto de códigos numéricos que describen las perforaciones a realizarse en la placa de circuito impreso, la herramienta a utilizar, el comando de inicio y parada para la máquina herramienta. (Figura 72)

El proceso de adquisición o captación de los datos CAM se realiza de la siguiente manera:

Captación Archivos CAM

Panel de Programación. En el Diagrama de Bloques, las principales funciones y VIs implementadas en este proceso son:

- *VIs y Funciones de I/O de Archivo.* abren, leen y cierran los archivos generados en el procesador CAM. Estas funciones son transparentes al tipo de dato.

Una vez abierto y leído el archivo con extensión DRI, se verifica que corresponda realmente al generado por el procesador CAM que contiene la información, entre otras, de las brocas utilizadas. Para ello se utiliza una función llamada “Search and Replace String” con la cual se busca y reemplaza una cadena de caracteres

dentro del archivo. Esta cadena de caracteres es: "Drills used", la cual siempre está presente, una sola vez, en los archivos DRI.

Figura 71. Apariencia y Contenido del Archivo DRI (Esquemático: hexapod2)

```

Archivo Edición Formato Ver Ayuda
Drill Station Info File: D:/Documents and Settings/borisAlejo/Escritorio/hexapod2.dri
Date           : 12/11/2006 19:31:27
Plotfile      : D:/Documents and Settings/borisAlejo/Escritorio/hexapod2.txt
Drills       : D:/Documents and Settings/borisAlejo/Escritorio/hexapod.dri
Device       : Sieb & Meyer 1000 drill station

Parameter settings:
Tolerance Drill + : 1.00 %
Tolerance Drill - : 1.00 %
Rotate           : no
Mirror           : no
optimize        : yes
Auto fit        : yes
OffsetX         : 0.000
OffsetY         : 0.000
Layers          : Drills Holes

Drill File Info:
Data Mode       : Absolute
End Of Block    : CR/LF
Units          : 1/100 mm

Drill Tolerances:
-- Requested -- -- Used Drill --
Size          used Code  Size
32.000mil    164 T02  0.810mm
36.000mil     58 T03  0.910mm
40.000mil    102 T04  1.020mm
52.000mil     4  T06  1.320mm
44.000mil    10  T05  1.120mm
130.000mil    2  T07  3.300mm

Drills used:
Code  Size          used
T01  0.500mm         41
T02  0.810mm        164
T03  0.910mm         58
T04  1.020mm        102
T05  1.120mm         10
T06  1.320mm         4
T07  3.300mm         2

Total number of drills: 381

```

La función "Search and Replace String" muestra en su salida el número de reemplazos realizados en el archivo, los cuales serán **1** o **0**. Estos datos son utilizados en una Estructura Caso para seguir con los subprogramas correspondientes.

- Verificación de Archivo DRI. Si el resultado del paso anterior es uno (1), es decir, se encontró en el archivo leído la cadena de caracteres. La estructura caso ejecuta el subprograma presente en su encabezado como **1**, de lo contrario ejecutara el subprograma de su encabezado cero **0**.

Figura 72. Vista “parcial” del Archivo de Códigos Numéricos (Esquemático: hexapod2)

```

Archivo Edición Formato Ver Ayuda
%%1000
X1677Y2076T01
X1677Y2266
X3582Y2901
X3582Y3155
X4598Y3155
X4598Y2901
X5360Y3028
X5550Y3219
X5550Y3409
X6630Y2647
X6630Y2393
X3137Y1631T02
X2883Y1631
X2883Y1885
X2883Y2139
X2883Y2393
X2883Y2647
X2883Y2901
X3137Y2901
X3137Y2647
X3137Y2393
X470Y742T03
X724Y742
X724Y996
X470Y996
X470Y1250
X724Y1250
X724Y1504
X470Y1504
X470Y1758
X470Y6584
X8090Y1123T04
X7836Y1123
X6820Y5695
X7074Y5695
X7328Y5695
X7582Y5695
X7836Y5695
X8090Y5695
X5487Y7283
X5487Y7537
X5169Y6965T05
X1613Y6965
X8344Y3409
X8344Y2393
X9424Y4933
X9424Y5187
X9424Y5441
X9424Y6076
X9424Y6330
X9424Y6584
X8938Y7473T06
X8189Y7473
X7440Y7473
X6439Y7473
X11176Y6330T07
X11176Y5187
M30

```

Subprograma Caso 0. EL archivo no es mostrado en el panel frontal e inmediatamente se genera un error, que el usuario visualiza a través de una ventana de dialogo. (Figura 73).

Subprograma Caso 1. La ruta o path del archivo DRI leído, es dividida en dos con la función “Strip Path” el cual entrega la cadena de caracteres correspondiente al directorio raíz, y; nombre y extensión del archivo. La extensión del archivo es reemplazada con la función “Replace Substring” por la de un archivo de texto (.txt), para construir el path del archivo de códigos numéricos.

De esta forma, se implementan nuevamente funciones de I/O de archivos para abrir el archivo Archivo TXT “Plotfile”, archivo de códigos numéricos, con el path de la construcción anterior, perteneciente al DRI creado con el procesador CAM.

3. Ventanas de diálogo de Errores. Existen tres ventanas de errores, cada una de ellas describe el posible motivo y se ejecutan de acuerdo a las estructuras casos correspondientes.

- El error generado al intentar abrir un archivo con extensión DRI, más no correspondiente al mismo. Hace que el programa enseñe al usuario una ventana de diálogo con un mensaje que describe su posible causa. Figura 73. Este mensaje es desplegado por la estructura caso que recibe el valor booleano (Falso o Verdadero) suministrado al verificarse si se realizó el reemplazo de la cadena de caracteres por la función “Replace Substring”, cada vez que se abre el archivo .TXT.

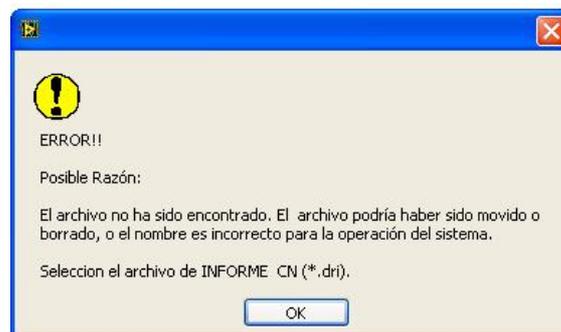
El proceso de verificación consiste en comparar un valor constante cero (0), con el valor entregado por la función “Replace Substring”, el cual también es cero (0) si el contenido del archivo no corresponde al generado por el procesador CAM.

Figura 73. Visualización Error por Ejecución Estructura Caso 0



También es posible que el programa no encuentre los archivos *.DRI, *.TXT, o *.DRL. Este error es monitoreado por la lectura del registro de Estados del programa, el cual entrega un valor booleano verdadero a una estructura caso al producirse el error. Al ejecutarse la estructura, el usuario visualizará los siguientes mensajes de diálogo, según el caso.

Figura 74. Mensaje desplegado al no encontrarse el archivo de informe de la estación de taladro, archivo de Códigos Numéricos o archivo de informe de herramientas



En el diagrama de bloques se encuentra, además una estructura Evento y una estructura While.

- *La Estructura Evento.* Espera hasta que el evento programado en su encabezado: cargar valor "Value Change" ocurra, para ejecutar el evento caso programado en su interior. Esto sucede cuando el usuario hace clic (izq) sobre el control del Path presente en el panel frontal (interfaz de Usuario) para seleccionar el informe del archivo a ejecutar, DRI. Cuando esto ocurre es posible la ejecución de las subrutinas programadas en ella (puntos 1,2 y 3 figura 75).

- *Estructura While.* Repite la ejecución de los subdiagramas (o subrutinas) programadas en su interior, hasta que se cumpla con la condición de parada.

Puesto que es necesario permitirle al usuario el acceso a un archivo en cualquier momento, la condición del terminal de parada se programa para que el While se detenga si la condición es verdadera.

Esta condición está dada por dos constantes booleanas (Falsas) ubicadas dentro y fuera de la estructura Evento y conectadas a una compuerta lógica OR. De esta forma se consigue que el valor de verdad del Terminal de parada, aun cuando la estructura evento no este en ejecución, sea falsa.

Panel Frontal: Interfaz Usuario. La interfaz gráfica ha sido diseñada sobre un control "Tab Control" que le permite al usuario navegar por varias ventanas, la primera de ellas ha sido denominada "Información Previa".

Básicamente consta de controles e indicadores para visualización e introducción de datos (Figura 75) que tienen como principales funciones:

- Acceder a los archivos CAM: *.DRI "Drill Station Info File" y *.TXT"PlotFile".
- Visualizar:
 - Archivos CAM
 - Número y diámetros comerciales de las brocas para el proceso de manufactura.
 - Perforaciones totales que la máquina herramienta debe ejecutar.

Figura 75. Apariencia de la ventana de Adquisición de los Archivos CAM.

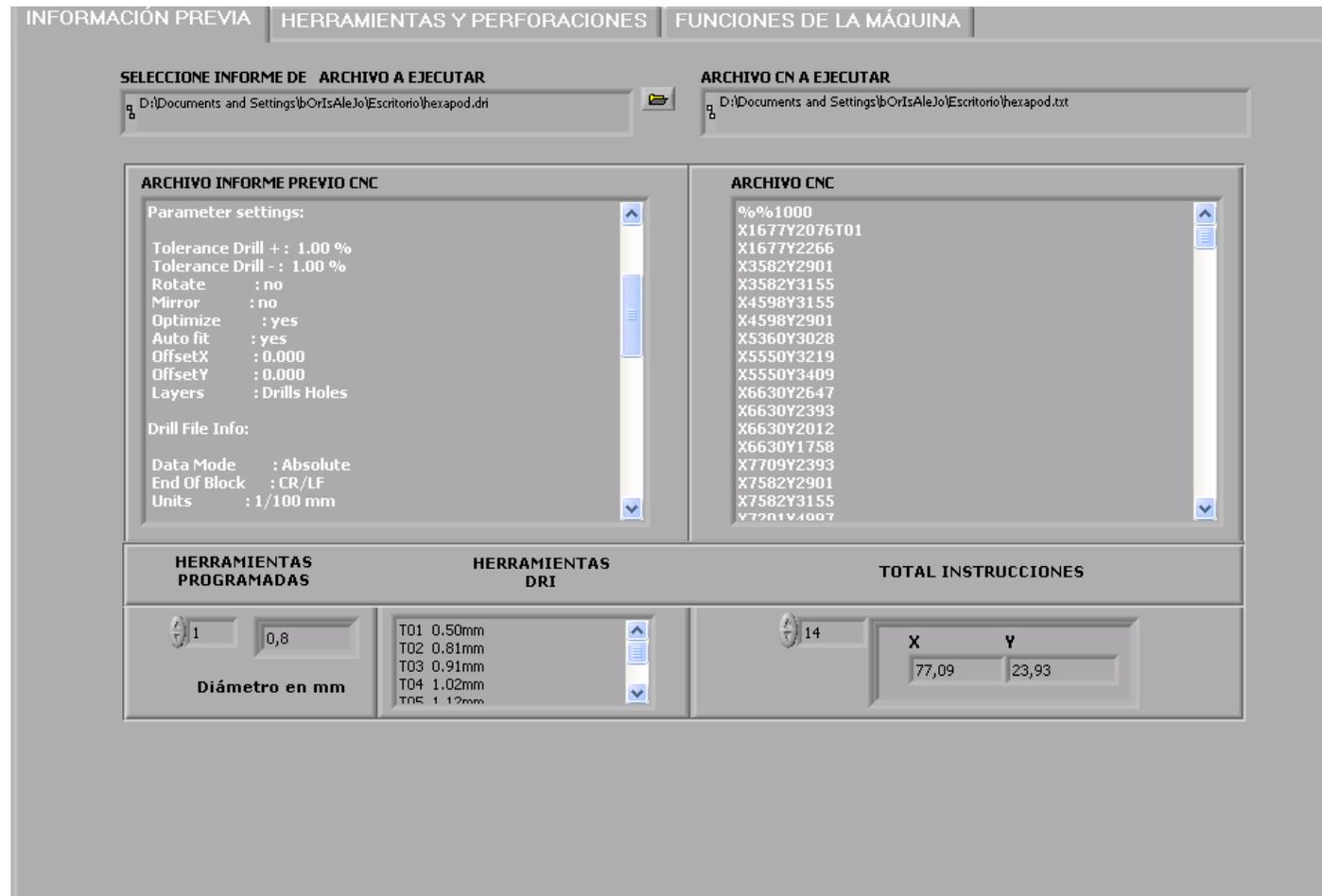
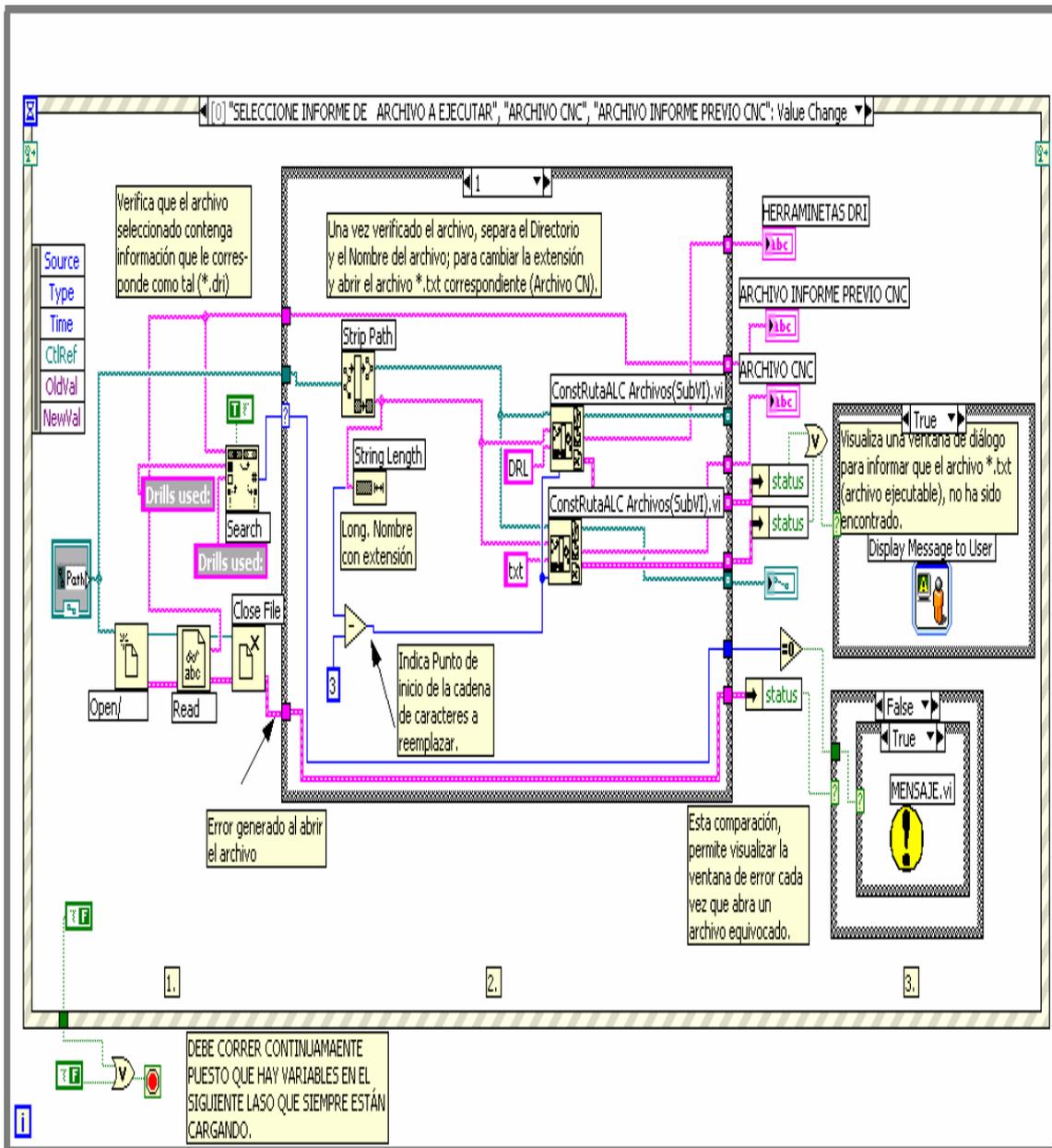


Figura 6 Diagrama de Bloques para Adquisición de los Archivos CAM



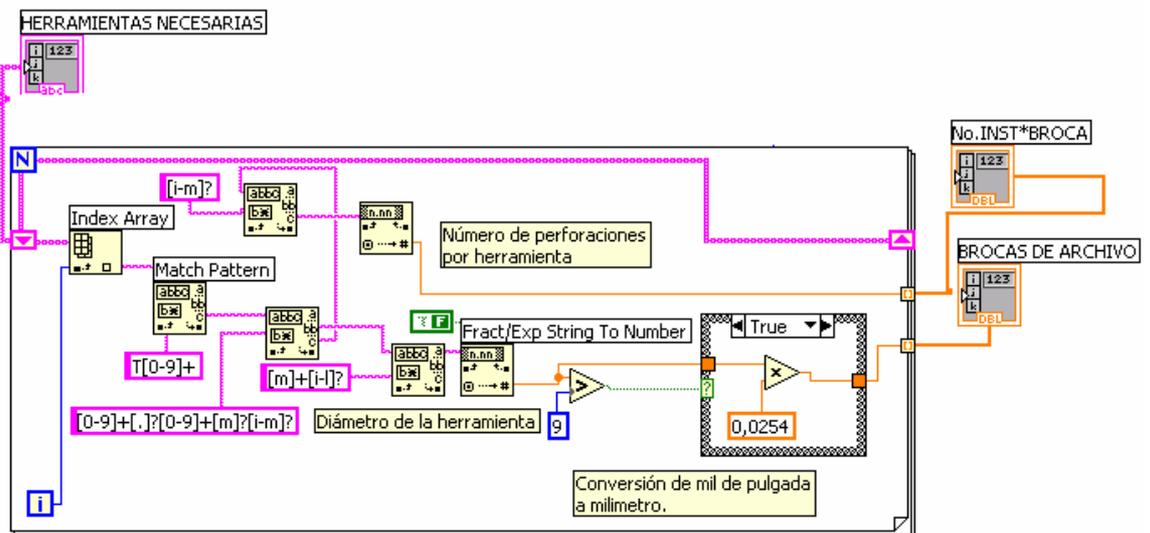
Acondicionamiento Archivos CAM. Para el desarrollo del proceso de manufactura, es esencial realizar un acondicionamiento tanto de los códigos numéricos que la máquina herramienta debe ejecutar, como de las herramientas necesarias para realizar las perforaciones.

Considerando los archivos generados por el procesador CAM (*.TXT Y *.DRI), se realiza la extracción de algunos datos, para permitirle al usuario interactuar con ellos a través de la interfaz gráfica, una vez efectuado los ajustes necesarios.

Para realizar este proceso, se creó un subVI's llamado "Manejo de Herramientas" (figura 80), el cual realiza las siguientes funciones:

- Ajuste de los diámetros de las brocas generadas en el archivo CAM *.DRL, a las comercialmente asequibles¹⁶.
- Separación de instrucciones numéricas por broca programada.

Figura 77. Subprograma Conversión de Unidades Brocas



¹⁶ Los formatos de comandos de archivos CAM generados por el software EAGLE, en este caso "Sieb & Meyer", son creados por empresas extranjeras, los cuales manejan un rango de diámetros, no comerciales en Colombia.

El archivo CAM *.DRI, proporciona información acerca del número de perforaciones a realizar por herramienta. Estos datos son extraídos y convertidos a milímetros si el archivo CAM *.DRL (contiene el diámetros de las herramientas), es generado en pulgadas.

La estructura For (Figura 77), se ejecuta el número de veces que indica la longitud del arreglo "HERRAMIENTAS NECESARIAS" que esta conectado a la entrada del registro (margen izquierda). Este arreglo contiene todas las herramientas, junto con el número de veces que debe usarse en el proceso de manufactura.

Estas herramientas son acondicionadas a través de varias funciones:

- Index Array. El cual permite tratar un dato a la vez.
- Match Patten. Busca los términos dentro del arreglo especificados en constantes String (cadenas de caracteres).
- Fract/Exp String To number. Convierte los caracteres alfa numéricos a número en diferentes formatos.

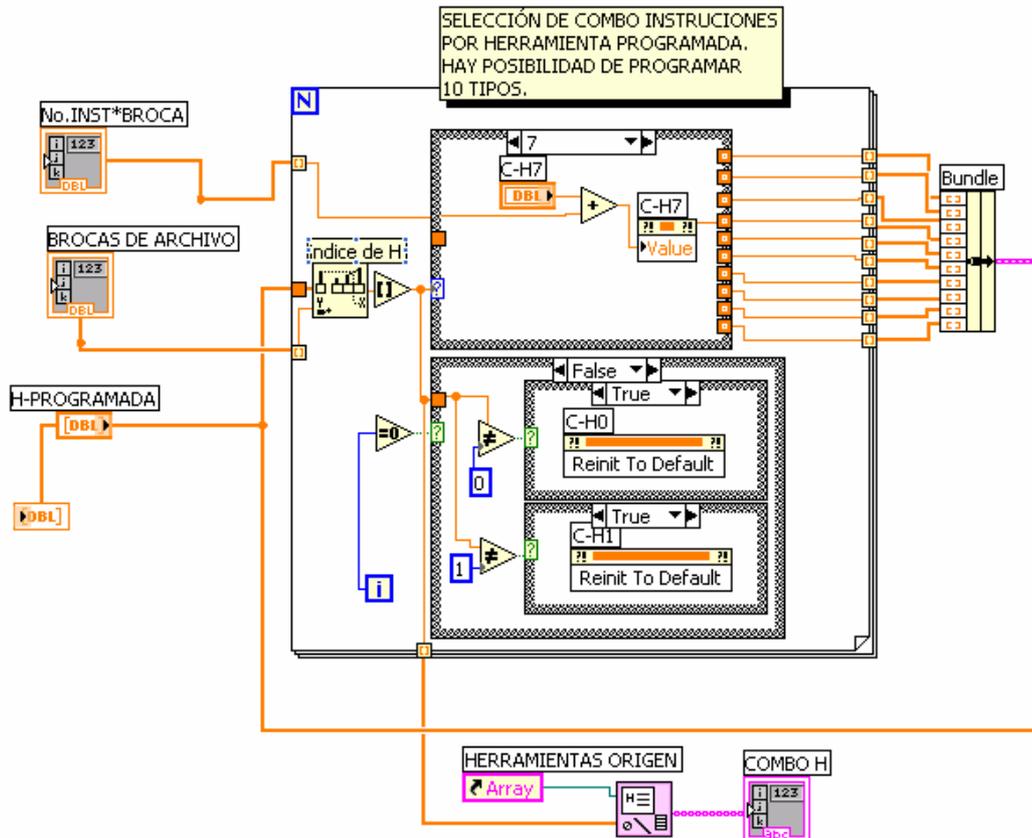
En el arreglo "BROCAS DE ARCHIVO" quedan almacenados los diámetros generados por el procesador CAM, con las unidades ajustadas. Adicionalmente, la subrutina de la estructura For (figura 69), realiza la separación del número de instrucciones que le corresponde a cada herramienta los cuales se guardan secuencialmente en un arreglo "No. INST*BROCA".

Ajuste Diámetros Herramientas. Realizado el ajuste de unidades (si es necesario) y la separación del número de perforaciones, se procede a establecer la adecuación de los diámetros de las herramientas (entregadas por el procesador CAM, archivo *.DRL), a las brocas comerciales. Para ello se toma como referencia o partida los arreglos generados en el proceso anterior: "BROCAS DE ARCHIVO y No.INST*BROCA". Además de un arreglo en el cual se han depositado los diámetros de las Brocas comerciales "H- PROGRAMADA".

Los datos de cada arreglo son implementados en una estructura For (Figura 78), en la que se emplea la subrutina de adecuación o ajuste de los diámetros, de la siguiente manera:

- Función Treshold 1D Array". Cada vez que se indexa un dato del arreglo "BROCAS DE ARCHIVO", la función verifica su proximidad con todos los diámetros comerciales del arreglo correspondiente. La función entrega un dato numérico correspondiente al índice de la herramienta al cual se acercó más. Este dato es usado para ejecutar una estructura Caso.

Figura 78. Subprograma: Ajuste de Diámetros Brocas



- La estructura Caso, permite almacenar las herramientas del archivo CAM, ajustadas a una misma broca, junto con su respectivo número de perforaciones total. Estos datos son ensamblados a través de una función Bundle.

Separación de Instrucciones Numéricas por Brocas. Con el fin de que el usuario visualice la información de manera adecuada, realizado el ajuste de los diámetros, se separa el archivo *.TXT (códigos numéricos), en sub-arreglos de acuerdo a la herramienta que los ejecutará.

También se consolida en un arreglo el diámetro y número total de herramientas necesarias en el proceso de manufactura, junto con las correspondientes perforaciones a realizar.

Figura 79. Subprograma: Separación de Instrucciones Numéricas por Brocas

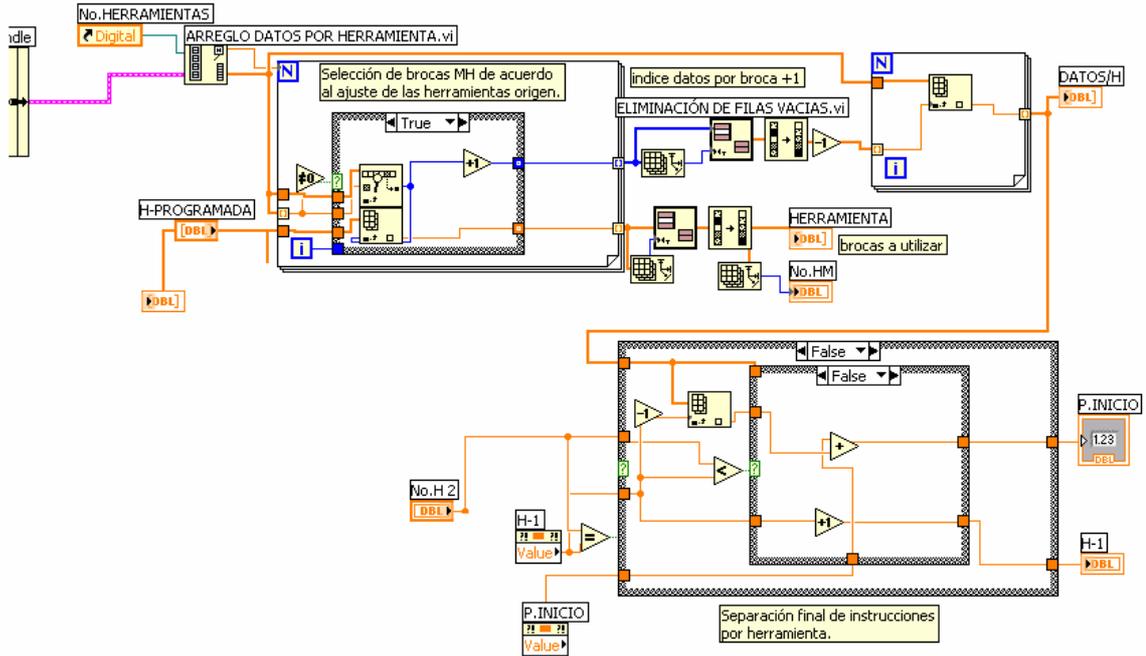
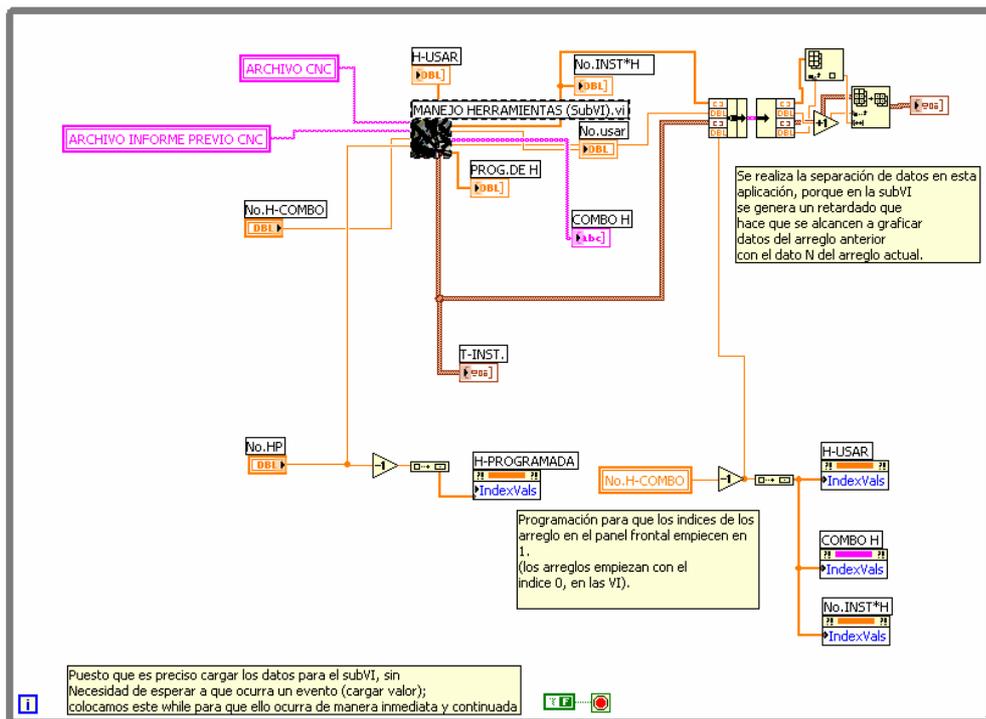


Figura 80. SubVI's Manejo de Herramientas



Finalmente, el SubVI'S "MANEJO DE HERRAMIENTAS", es implementado dentro de una estructura While que siempre se está ejecutando, con el fin de garantizar la actualización constante e inmediata de los datos.

Presentación de los Archivo CAM. En la descripción del proceso de Captación, se mostró el panel de Programación o Diagrama de Bloques y Panel Frontal correspondiente para tal proceso. Sin embargo la vista del Panel Frontal del proceso de Acondicionamiento, contiene una serie de controles que han sido adicionados con el propósito de mejorar la integración del usuario al proceso de manufactura.

Estos son:

- Pictures
- Controles Booleanos: Zoom
- Cluster

Panel Frontal: Interfaz Usuario. La segunda ventana de navegación de la interfaz gráfica ha sido denominada "Herramientas y Perforaciones". Básicamente consta de indicadores para visualización de datos (Figura 73) que tienen como principales funciones:

- Visualizar:
 - Posición de las Perforaciones, las cuales se visualizan a través del control "Picture". Adicionalmente se han empleado propiedades de este control que proporciona al usuario la posibilidad de navegar por la imagen de las perforaciones visualizando el valor del punto señalado por el raton dentro de ella (Tip Strip). También se manejan las propiedades "Área y Origen", para que el usuario pueda realizar un zoom del dibujo de las perforaciones, a través de 3 botones sobre la ventana de usuario.
 - Número y diámetros de las brocas en el proceso de manufactura.
 - Perforaciones totales que la máquina herramienta debe ejecutar.

Figura 81. Panel Frontal: Presentación de Herramientas y Perforaciones

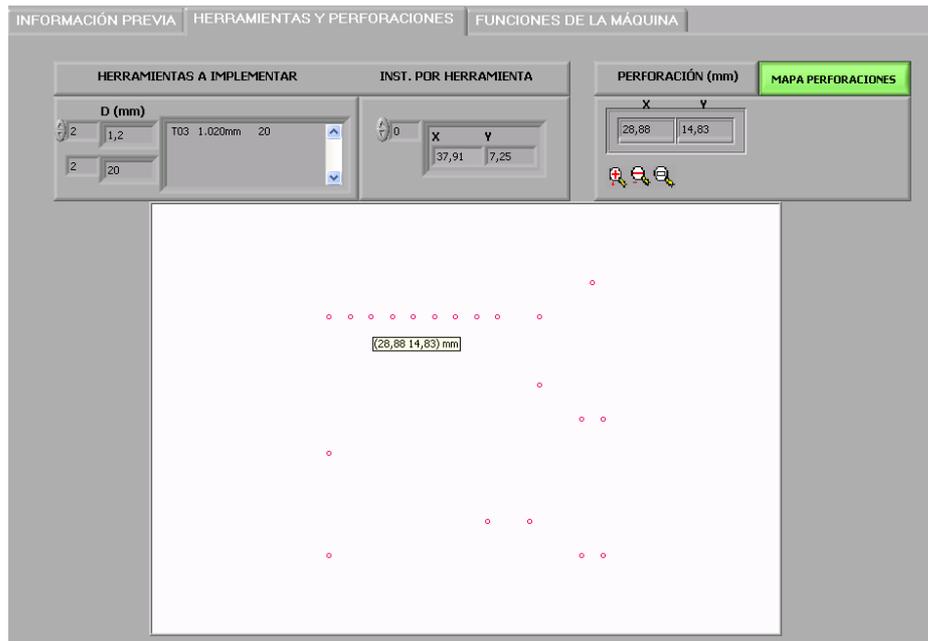
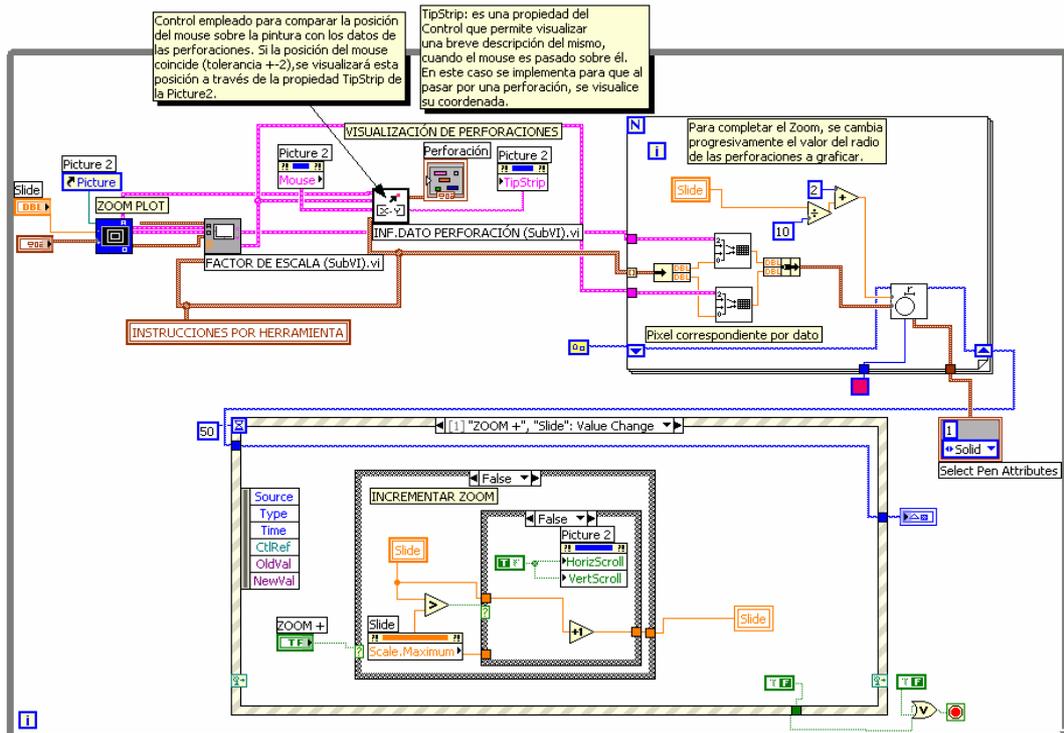


Figura 82. Subprograma Zoom Mapa de Perforaciones y Visualización Punto Señalado



Datos Funcionales de la Máquina Herramienta En la interfaz electrónica varios sensores suministran información al elemento controlador del sistema encargado de proveer todos los datos necesarios para la operación de la máquina herramienta.

El elemento controlador, que en este caso se basa en un circuito integrado LM629 “Procesador de Control de Movimiento (PID)”, recibe la información de los sensores en un proceso de comunicación establecido en la interfaz electrónica a través de un microcontrolador PIC16F874. Este proceso de comunicación también permite la transferencia de información a la interfaz gráfica.

El elemento controlador da las órdenes oportunas a los actuadores para mantener el proceso funcionando dentro de los márgenes previstos, y proporciona a la interfaz gráfica los datos de algunas funciones de la MHCNC, para su presentación o procesamiento, tales como:

- Posición Actual
- Posición Deseada
- Parada Manual
- Estado de la Máquina: Libre, en Movimiento, Perforando, Cambio de Herramienta, Error, Parada Manual.

Captación, Acondicionamiento y Presentación. El proceso de adquisición de los datos funcionales de la máquina es llevado a cabo a través de los VI's VISA los cuales permiten establecer una comunicación serial con la interfaz electrónica (Escribir, Leer y Cerrar la sesión). Puesto que la comunicación en la interfaz electrónica es manejada por el microcontrolador, las subrutinas del mismo aseguran una comunicación Half Duplex, que limita el tiempo de muestreo y actualización de datos.

La VI's VISA maneja la información en cadenas de caracteres de diversos formatos, para la comunicación con la máquina se trabajó en Hexagecimal String, por lo tanto se hizo necesario convertir las coordenadas de la perforaciones, teniendo en cuenta la resolución de los Codestrip y Codewheel.

- Eje X: se trabajó con una resolución de 5000 líneas por 30,5 cm, es decir: aproximadamente 16,4 líneas por mm.
- Eje Y: se trabajó con una resolución de 25.000 líneas por 34 cm, es decir; aproximadamente 73,5 líneas por mm.
- Eje Z: se trabajó con una resolución de 492 líneas por 3 cm, es decir; 16,4 líneas por mm.

Posición Actual. La posición de la máquina es leída a través del puerto serial desde la interfaz, enviando un comando al microcontrolador para indicarle la información requerida. Esta operación se lleva a cabo cada 100 ms. Los datos leídos son separados en un arreglo y convertidos a formato decimal (teniendo en cuenta, de nuevo, la resolución).

Los datos correspondientes a la posición de las coordenadas son visualizados en la interfaz implementando controles numéricos.

Programación Máquina. Para realizar finalmente el proceso de manufactura, se han dado dos posibilidades al usuario (Figura 83):

- *Automática.* La opción automática sólo requiere que el usuario oprima un botón para que el proceso de manufactura se lleve a cabo, y lo único que debe hacer el usuario es realizar el cambio de herramienta cada vez que el proceso lo solicite.
- *Manual.* La opción manual requiere que el usuario interactúe permanentemente con la interfaz gráfica. El proceso de manufactura manual, cuentan con botones de programación para cada eje, junto con los botones de inicio y parada del proceso.

Bloque de Programación. Este proceso, se trabaja en una estructura caso conectada a la salida de un control “Radio Button”, el cual le permite al usuario elegir entre estos dos tipos de programación de la máquina, de manera excluyente.

Una vez el botón de ejecutar ha sido oprimido (verdadero), los subprogramas realizados dentro de la estructura caso que espera el valor de verdad del botón, son ejecutados.

Para realizar la transmisión de las coordenadas de perforación se considera el estado actual de la máquina antes de efectuar la transferencia, estos son:

L: libre
M: en Movimiento
B: Colocando Herramienta
P: Perforando
E: Error
S: Parada Manual

Estos estados son suministrados por el microcontrolador. Estos códigos son colocados en un arreglo y son comparados (en la estructura For), con los datos leídos por la interfaz gráfica, suministrados por el microcontrolador. Para

posteriormente ejecutar un subprograma en una estructura caso de acuerdo al código correspondiente.

L: Este subprograma permite que la interfaz envíe un dato a la interfaz electrónica, a través de la ya mencionada VI's VISA.

- *Modo Automático.* Las instrucciones por herramienta, ubicadas en el arreglo correspondiente, son escritas en el puerto con el código "P", adicionalmente, para indicarle a la máquina que debe perforar en este punto. Para enviar un dato a la vez, el arreglo es indexado por una estructura For del LabView. El proceso de indexado de la estructura For, además de permitir que los datos del arreglo entren a ella uno a la vez, hace que el For se ejecute de acuerdo a la cantidad de datos del arreglo, sin necesidad de conocer su longitud previamente.

- *Modo Manual.* En este caso el dato enviado a la interfaz electrónica corresponde al código "M" que le indica a la máquina que debe moverse en la coordenada seleccionada por el usuario.

Finalmente dentro de la estructura se lleva a cabo el proceso de escritura del dato en el puerto con la VI's VISA.

B: Indica que se realizó el cambio correspondiente de herramienta y por ende la máquina está posicionada en el home, por lo tanto el inicio del proceso de perforaciones de una herramienta puede efectuarse.

E: Cuando ocurre un error de posición excesivo, un error de comunicación, no encuentra el home o cualquier problema mecánico. La interfaz cambia el estado del botón ejecutar (falso) y los subprogramas dejan de correr, es decir el proceso se detiene. El usuario visualiza esta situación, con una ventana de mensaje desplegada en la interfaz.

S: La máquina cuenta con un botón externo que el usuario puede utilizar para detener el proceso de manufactura. Esto puede ser preciso por ejemplo, si la herramienta se rompe, ajuste o posicionamiento de la baqueta, etc. Si ello ocurre el proceso se detiene cambiando el valor de verdad del botón ejecutar.

Estos códigos son monitoreados por el subprograma de escritura y lectura del puerto, cada vez que se establece la comunicación, o sea; cada 100mseg.

Al igual que los subprogramas mencionados en la etapa de captación y acondicionamiento de los archivos CAM, los subprogramas de adquisición, acondicionamiento y presentación de los datos funcionales de la máquina, se encuentran dentro de una estructura while, con el fin de que su ejecución sea continuada.

Figura 83. SubPrograma Funciones de la Máquina

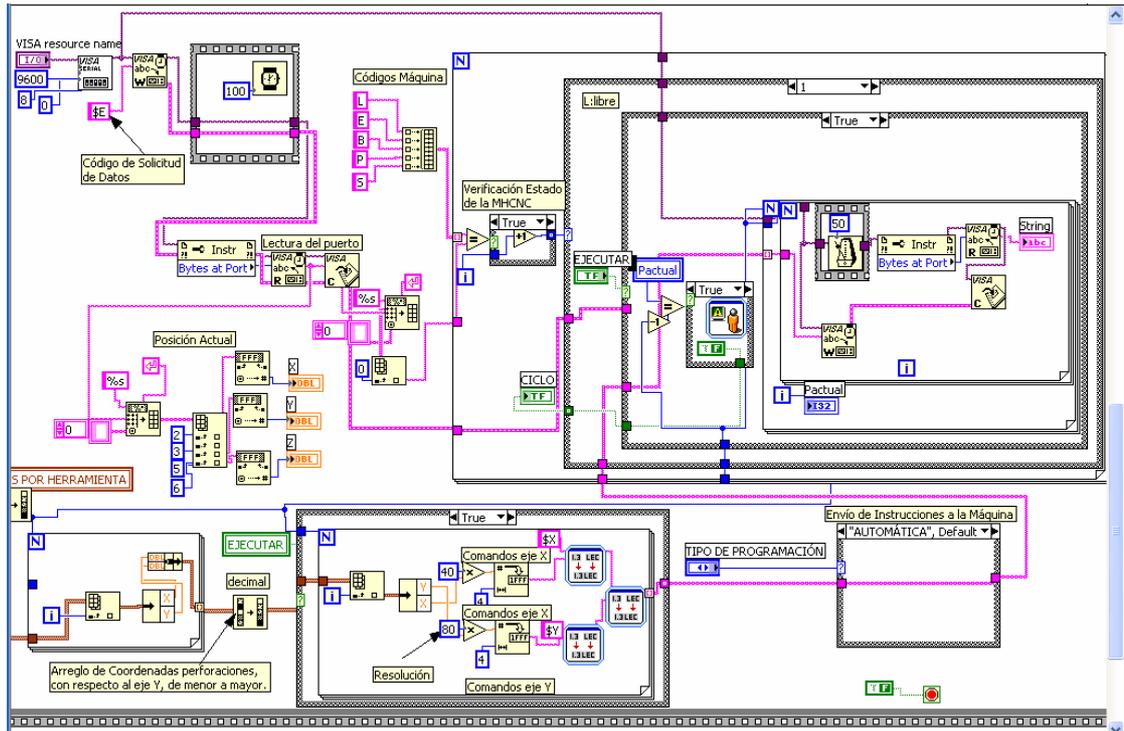


Figura 84. Panel de Frontal: Funciones de la Máquina

INFORMACIÓN PREVIA | HERRAMIENTAS Y PERFORACIONES | FUNCIONES DE LA MÁQUINA

POSICIÓN ACTUAL (mm) COORDENADA X: 59,3 COORDENADA Y: 0,03333 COORDENADA Z: 0			TIPO DE OPERACIÓN Y ESTADO AUTOMÁTICA: OFF/DN MANUAL: OFF/DN MOVIENDOSE: ON (Red) PERFORANDO: ON (Red) LIBRE: ON (Green)		
POSICIÓN AUTOMÁTICA DESEADA (mm) COORDENADA X: 0 COORDENADA Y: 0			MANUFACTURA AUTOMÁTICA PERFORACIONES: 0 EN PROCESO, 20 TOTAL BROCA (mm): 1,2 EJECUTAR		
VELOCIDAD MOTOR PRINCIPAL Velocidad en rpm: 59 (Vmax to Vmin scale)			MANUFACTURA MANUAL X: 59, Y: 0, Z: 0 Cambio de Broca: 1/2		

6. CONCLUSIONES Y RECOMENDACIONES

Aunque el software de diseño mecánico es una versión de evaluación y no cuenta con un con herramientas de análisis mecánico como el COSMOSxpress y el COSMOSmotion, el desarrollo del mecanismo del prototipo MHCNC para la perforación de circuitos impresos, no se podría haber ensamblado sin contar con el programa de diseño CAD que permitiera esbozar las ideas y discutir las con diseñadores y fabricantes, definir dimensiones para fabricarlas, comprobar si las piezas son físicamente realizables y los ensamblajes permitan su posterior montaje. También el software EAGLE permite el diseño exitoso de las artes de los circuitos impresos para el montaje de los componentes electrónicos del prototipo MHCNC. Si bien, en el desarrollo de este prototipo se limitan los paquetes de software al proceso de dibujo, considérese las otras miles de aplicaciones que no son utilizadas por no ser difundidas en nuestro medio o por que las versiones profesionales tienen un alto costo, y su verdadera potencialidad es su uso en sistemas CAD/CAM. Aun mas, si se analizan los diferentes propósitos de distintos software, se vera que muchos necesitan integrarse a procesos automatizados. Por tal razón, se sugiere que al proceso de optimización del prototipo MHCNC, se integre un paquete completo de los programas implementados, u otros, que permitan manejar un mayor número de tareas de ingeniería.

Los sistemas de desplazamiento, sujeción, y engranajes de la MHCNC fueron construidos con las herramientas que el presupuesto permitió adquirir, sin embargo; existen mecanismos fabricados con alta tecnología como los ejes de precisión, tornillos de esferas recirculantes, engranajes armónicos, sistemas y guías lineales, que aunque presentan costos elevados, maximizaría y potenciarían la fabricación y operación de los diversos sistemas que integran la MHCNC.

Varios métodos pueden usarse para determinar los parámetros de un PID para una aplicación particular. Sin embargo, debe considerarse el poseer un conocimiento profundo de las características del controlador. El ajuste debe someterse a un estudio adecuado con el fin de maximizar y potencializar el uso del control del PID, así como los resultados obtenidos en el proceso de diseño.

Los resultados obtenidos por medio de la correcta sintonización del filtro de compensación PID, no implicaron un conocimiento exacto del modelo matemático de la planta, demostrando que la estructura del PID ofrece suficiente flexibilidad para dar excelentes resultados en el control industrial.

El programa principal de la MHCNC para la perforación de circuitos impresos, realiza la integración del diseño de circuitos impresos y manufactura, de manera transparente al usuario, a través de las interfaces electrónica y gráfica. El programa, no exige que el operario de la máquina tenga conocimientos en control numérico, este sólo debe poseer conocimientos básicos en sistemas y seguridad industrial reduciendo su labor al inicio, monitoreo, colocación de la baqueta y la herramienta.

El desarrollo de la Máquina Herramienta CNC para la Perforación de Circuitos Impresos, esbozada en el presente trabajo, proporciona una idea mediante la cual se puede llevar a cabo una etapa de manufactura de circuitos impresos, comprobada la factibilidad de su implementación, se abre paso a la posibilidad de automatizar otros procesos más de esta etapa de manufactura como la impresión de las pistas en la baqueta, etc.

El prototipo MHCNC para la perforación de circuitos impresos, es una pieza de ingeniería colombiana que pone al alcance de la industria la posibilidad de automatizar diversos procesos integrando el control numérico a los mismos, a un bajo costo.

El desarrollo de Máquina Herramienta con CNC, no se limita a un proceso específico. La fabricación de estas máquinas permite integrar cualquier tipo de programa de diseño por computador al correspondiente proceso de manufactura.

BIBLIOGRAFÍA

Paul H Lewis. Sistemas de Control en Ingeniería. Madrid. 1^{ra} ed: Prentice Hall Iberia, 1999, 480p.

Katsuhiko Ogata. Sistemas de Control en Tiempo Discreto. México. 2^{da} ed : Pearson Educación, Reg. Núm 1524, s.p.

LabVIEW [CD-ROM]. [USA]: National Instruments.

SolidWorks Office Premium [Tour and Demo CD-ROM]. [MA, USA]: SolidWorks Corporations, 2005.

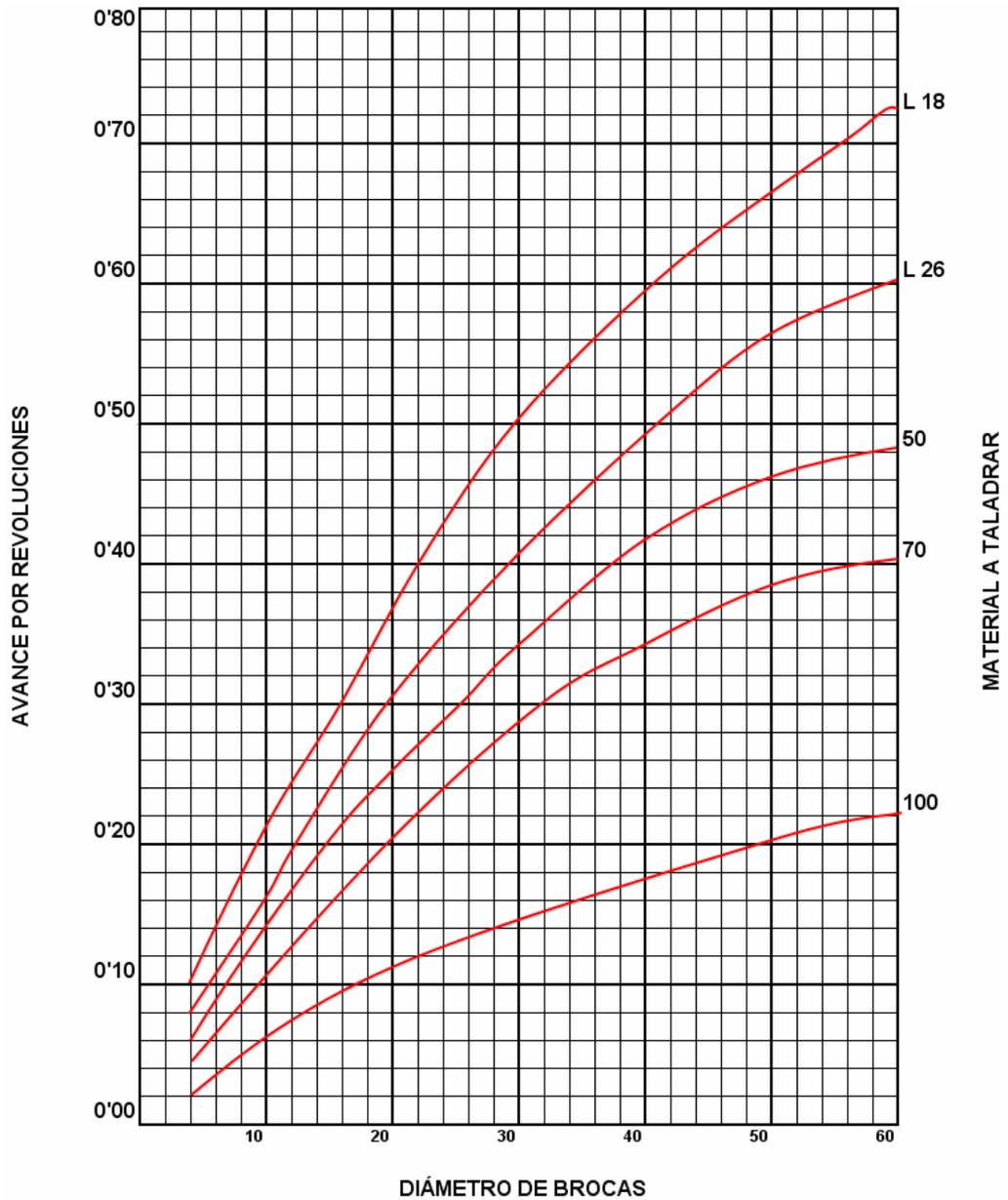
LabVIEW Fundamentals (en línea): National Instruments. Disponible en Internet: www.ni.com, NI Developer Zone, Knowledge Base, Product Manuals Library.

Data Sheet LM629 (en línea). Disponible en Internet: www.ni.com.

Data Sheet HDS9700 Serie (en línea). Disponible en Internet: www.avagotech.com.

Anexos

Anexo A. Avance de las Brocas para Distintos Materiales



L 18 = Latón - Aluminio Metal blando y Fundición hasta 18Kgs/mm²
L 26 = Cobre - Bronce y Fundición hasta 26Kgs/mm²
50= Acero hasta 50Kgs/mm²
70= Acero hasta 70Kgs/mm²

Fuente: A.L.Casillas. Máquinas: Cálculos de Taller. 23 Edición. Madrid España: Gráficas Reunidas

Anexo B. Características Principales Del Lm629



January 2003

LM628/LM629 Precision Motion Controller

General Description

The LM628/LM629 are dedicated motion-control processors designed for use with a variety of DC and brushless DC servo motors, and other servomechanisms which provide a quadrature incremental position feedback signal. The parts perform the intensive, real-time computational tasks required for high performance digital motion control. The host control software interface is facilitated by a high-level command set. The LM628 has an 8-bit output which can drive either an 8-bit or a 12-bit DAC. The components required to build a servo system are reduced to the DC motor/actuator, an incremental encoder, a DAC, a power amplifier, and the LM628. An LM629-based system is similar, except that it provides an 8-bit PWM output for directly driving H-switches. The parts are fabricated in NMOS and packaged in a 28-pin dual in-line package or a 24-pin surface mount package (LM629 only). Both 6 MHz and 8 MHz maximum frequency versions are available with the suffixes -6 and -8, respectively, used to designate the versions. They incorporate an SDA core processor and cells designed by SDA.

Features

- 32-bit position, velocity, and acceleration registers
- Programmable digital PID filter with 16-bit coefficients
- Programmable derivative sampling interval
- 8- or 12-bit DAC output data (LM628)
- 8-bit sign-magnitude PWM output data (LM629)
- Internal trapezoidal velocity profile generator
- Velocity, target position, and filter parameters may be changed during motion
- Position and velocity modes of operation
- Real-time programmable host interrupts
- 8-bit parallel asynchronous host interface
- Quadrature incremental encoder interface with index pulse input
- Available in a 28-pin dual in-line package or a 24-pin surface mount package (LM629 only)

LM628/LM629 Precision Motion Controller

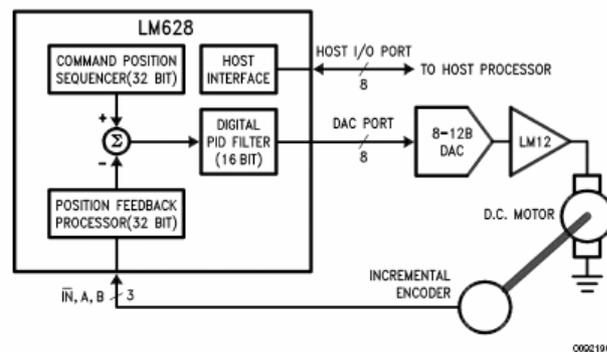


FIGURE 1. Block Diagram

TRISTATE® is a registered trademark of National Semiconductor Corporation.

AC Electrical Characteristics (Continued)(V_{DD} and T_A per Operating Ratings; f_{CLK} = 6 MHz; C_{LOAD} = 50 pF; Input Test Signal t_r = t_f = 10 ns)

Timing Interval	T#	Tested Limits		Units
		Min	Max	
STATUS BYTE READ TIMING (See Figure 4)				
Chip-Select Setup/Hold Time	T7	0		ns
Port-Select Setup Time	T8	30		ns
Port-Select Hold Time	T9	30		ns
Read Data Access Time	T10		180	ns
Read Data Hold Time	T11	0		ns
RD High to Hi-Z Time	T12		180	ns
COMMAND BYTE WRITE TIMING (See Figure 5)				
Chip-Select Setup/Hold Time	T7	0		ns
Port-Select Setup Time	T8	30		ns
Port-Select Hold Time	T9	30		ns
Busy Bit Delay	T13		(Note 3)	ns
WR Pulse Width	T14	100		ns
Write Data Setup Time	T15	50		ns
Write Data Hold Time	T16	120		ns
DATA WORD READ TIMING (See Figure 6)				
Chip-Select Setup/Hold Time	T7	0		ns
Port-Select Setup Time	T8	30		ns
Port-Select Hold Time	T9	30		ns
Read Data Access Time	T10		180	ns
Read Data Hold Time	T11	0		ns
RD High to Hi-Z Time	T12		180	ns
Busy Bit Delay	T13		(Note 3)	ns
Read Recovery Time	T17	120		ns
DATA WORD WRITE TIMING (See Figure 7)				
Chip-Select Setup/Hold Time	T7	0		ns
Port-Select Setup Time	T8	30		ns
Port-Select Hold Time	T9	30		ns
Busy Bit Delay	T13		(Note 3)	ns
WR Pulse Width	T14	100		ns
Write Data Setup Time	T15	50		ns
Write Data Hold Time	T16	120		ns
Write Recovery Time	T18	120		ns

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond the above Operating Ratings.

Note 2: When operating at ambient temperatures above 70°C, the device must be protected against excessive junction temperatures. Mounting the package on a printed circuit board having an area greater than three square inches and surrounding the leads and body with wide copper traces and large, uninterrupted areas of copper, such as a ground plane, suffices. The 28-pin DIP (N) and the 24-pin surface mount package (M) are molded plastic packages with solid copper lead frames. Most of the heat generated at the die flows from the die, through the copper lead frame, and into copper traces on the printed circuit board. The copper traces act as a heat sink. Double-sided or multi-layer boards provide heat transfer characteristics superior to those of single-sided boards.

Note 3: In order to read the busy bit, the status byte must first be read. The time required to read the busy bit far exceeds the time the chip requires to set the busy bit. It is, therefore, impossible to test actual busy bit delay. The busy bit is guaranteed to be valid as soon as the user is able to read it.

Anexo C. Formato Sieb&Meyer

276

Sieb&Meyer

1.0 Application The Sieb&Meyer data contain the information on the hole coordinates of a CAD layout.

The Sieb&Meyer format is a standard file structure which makes the data exchange between CAD (development) and CAM (production) possible.

A tool-list in which the tool assignment is defined, belongs to each Sieb&Meyer data set.

2.0 File format The file format must be ASCII.
We prefer the m.n.-format 3.3mm.

2.1 m.n-Format see Document „m.n.-Format [S.177]“

2.2 Drill types **Standard-Drill** : Data set and tool list are separated.
Extended-Drill : The tool-list is included in the header of the data.

We prefer "Extended drill".

2.3 Structure The basic structure of Sieb&Meyer data is:

1.Block

X-, Y-coordinate	Tool	End of Block
------------------	------	--------------

2.Block

X-, Y-coordinate	End of Block
------------------	--------------

etc...

The X and Y-coordinates are always indicated without ".".
Each block must be finished by "Return" and "Linefeed" (CR/LF).
Two "%" symbols at the beginning of the file define the data as CNC data and initialise the machine.
The command "M30" indicates the end of programme.

2.4 Tools The tool change is defined by the indication of a new tool, the tool is positioned behind the first coordinate to be drilled with this tool.

The tool designation consists of the letter "T" and a number (example: T1, T2, ...).

© Copyright by ILFA GmbH / Edition 7.2 - 2005



**CAD and CAM
Specifications**

File : Sieb&Mm.1
Issued : 11.12.1996 / jvt
Revision : 16.03.2005 / Wi

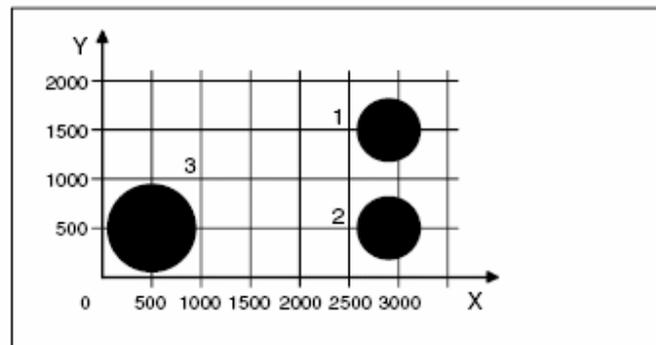
Sieb&Meyer

2.4.1 Example

Sieb&Meyer data (SM1000)

File content	Function	No.
%%1000	initialise machine	
X2800Y1500T1	approach coordinate, choose tool 1, drill	1
X2800Y500	drill coordinate	2
X500Y500T2	approach coordinate, choose tool 2, drill	3
M30	end of programme	

Drilling picture



2.5 Tool-list

see Document „Tool-list [S.305]”

3.0 Note

The following aspects are not described here:

- Command to set an origin (G90).
- Step&Repeat command (M31, M30, M50, V*, W*).
- Command for inspection stop or notes for operators (M20, M47)

3.1 Use

Sieb&Meyerfiles are necessary for:

- the CAM processing (panels)
- the CNC processing (drilling)
- the electronic test (construction of adapters, test file)

4.0 Cross-references (hyperlinks)

„Tool-list [S.305]” ; „m.n.-Format [S.177]” ;
 „Drills and Holes [S.74]” ; „Non-Plated Holes [S.217]” ;
 „Drilling plan [S.58]” ; „Excellon [S.94]”



**CAD and CAM
Specifications**

File : Sieb&Mfm.2
 Issued : 11.12.1996 / jvi
 Revision : --

Anexo D. Principales Característica del Microcontrolador



PIC16F87X

28/40-Pin 8-Bit CMOS FLASH Microcontrollers

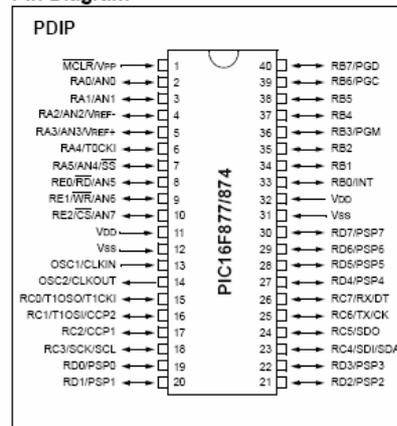
Devices Included in this Data Sheet:

- PIC16F873
- PIC16F876
- PIC16F874
- PIC16F877

Microcontroller Core Features:

- High performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory,
Up to 368 x 8 bytes of Data Memory (RAM)
Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and
Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC
oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low power, high speed CMOS FLASH/EEPROM
technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP) via two
pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial, Industrial and Extended temperature
ranges
- Low-power consumption:
 - < 0.6 mA typical @ 3V, 4 MHz
 - 20 µA typical @ 3V, 32 kHz
 - < 1 µA typical standby current

Pin Diagram



Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,
can be incremented during SLEEP via external
crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period
register, prescaler and postscaler
- Two Capture, Compare, PWM modules
 - Capture is 16-bit, max. resolution is 12.5 ns
 - Compare is 16-bit, max. resolution is 200 ns
 - PWM max. resolution is 10-bit
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI™ (Master
mode) and I²C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver
Transmitter (USART/SCI) with 9-bit address
detection
- Parallel Slave Port (PSP) 8-bits wide, with
external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for
Brown-out Reset (BOR)

Anexo E. Descripción de la Memoria RAM 6116



CMOS Static RAM
16K (2K x 8-Bit)

IDT6116SA
IDT6116LA

Features

- ◆ High-speed access and chip select times
 - Military: 20/25/35/45/55/70/90/120/150ns (max.)
 - Industrial: 20/25/35/45ns (max.)
 - Commercial: 15/20/25/35/45ns (max.)
- ◆ Low-power consumption
- ◆ Battery backup operation
 - 2V data retention voltage (LA version only)
- ◆ Produced with advanced CMOS high-performance technology
- ◆ CMOS process virtually eliminates alpha particle soft-error rates
- ◆ Input and output directly TTL-compatible
- ◆ Static operation: no clocks or refresh required
- ◆ Available in ceramic and plastic 24-pin DIP, 24-pin Thin Dip, 24-pin SOIC and 24-pin SOJ
- ◆ Military product compliant to MIL-STD-883, Class B

Description

The IDT6116SA/LA is a 16,384-bit high-speed static RAM organized as 2K x 8. It is fabricated using IDT's high-performance, high-reliability CMOS technology.

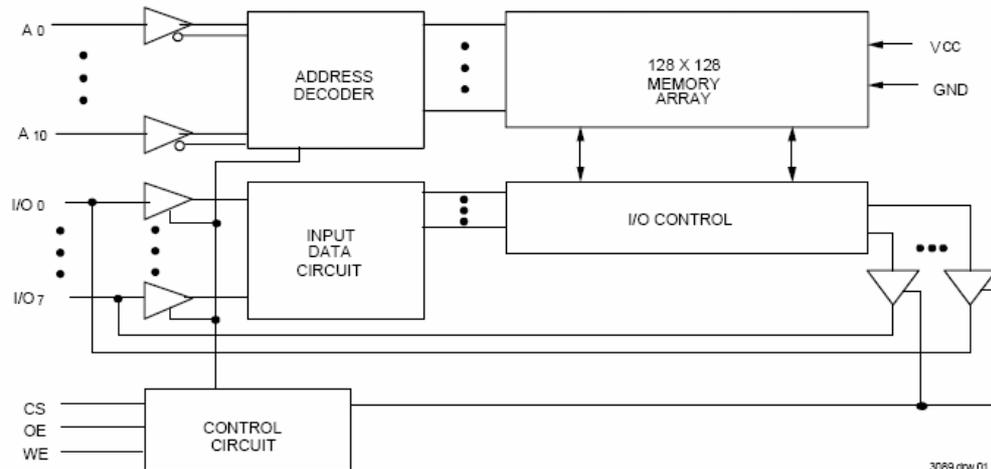
Access times as fast as 15ns are available. The circuit also offers a reduced power standby mode. When \overline{CS} goes HIGH, the circuit will automatically go to, and remain in, a standby power mode, as long as \overline{CS} remains HIGH. This capability provides significant system level power and cooling savings. The low-power (LA) version also offers a battery backup data retention capability where the circuit typically consumes only 1 μ W to 4 μ W operating off a 2V battery.

All inputs and outputs of the IDT6116SA/LA are TTL-compatible. Fully static asynchronous circuitry is used, requiring no clocks or refreshing for operation.

The IDT6116SA/LA is packaged in 24-pin 600 and 300 mil plastic or ceramic DIP, 24-lead gull-wing SOIC, and 24-lead J-bend SOJ providing high board-level packing densities.

Military grade product is manufactured in compliance to the latest version of MIL-STD-883, Class B, making it ideally suited to military temperature applications demanding the highest level of performance and reliability.

Functional Block Diagram



MARCH 2005



Small Optical Encoder Modules

Technical Data

HEDS-9700 Series

Features

- Small Size
- Low Cost
- Multiple Mounting Options
- Wide Resolution Range
- Linear and Rotary Options Available
- No Signal Adjustment Required
- Insensitive to Radial and Axial Play
- -40°C to +85°C Operating Temperature

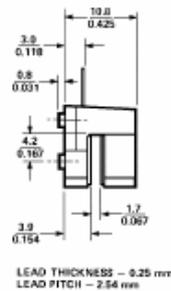
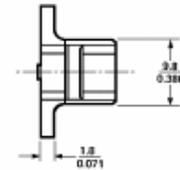
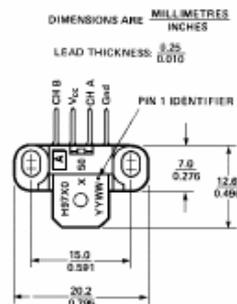
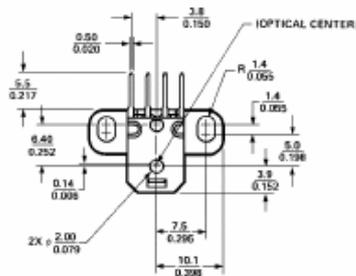
- Two Channel Quadrature Output
- TTL Compatible
- Single 5V Supply
- Wave Solderable

Description

The HEDS-9700 series is a high performance, low cost, optical incremental encoder module. When operated in conjunction with either a codewheel or codestrip, this module detects rotary or linear position. The



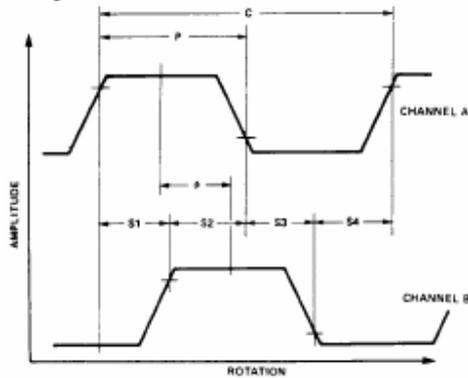
Package Dimensions



Mounting Option #50 - Standard (Baseplane Mounting) Contact Factory for Detailed Package Dimensions

ESD WARNING; NORMAL HANDLING PRECAUTIONS SHOULD BE TAKEN TO AVOID STATIC DISCHARGE.

Output Waveforms



Definitions

Count (N) – The number of bar and window pairs or counts per revolution (CPR) of the codewheel, or the number of lines per inch of the codestrip (LPI).

- 1 Shaft Rotation – 360 mechanical degrees
- N cycles
- 1 cycle (c) – 360 electrical degrees (°e)
- 1 bar and window pair

Pulse Width (P): The number of electrical degrees that an output is high during one cycle. This value is nominally 180°e or 1/2 cycle.

Pulse Width Error (ΔP): The deviation, in electrical degrees, of the pulse width from its ideal value of 180°e.

State Width (S): The number of electrical degrees between a transition in the output of channel

A and the neighboring transition in the output of channel B. There are 4 states per cycle, each nominally 90°e.

State Width Error (ΔS): The deviation, in electrical degrees, of each state width from its ideal value of 90°e.

Phase (Φ): The number of electrical degrees between the center of the high state of channel A and the center of the high state of channel B. This value is nominally 90°e for quadrature output.

Phase Error (ΔΦ): The deviation of the phase from its ideal value of 90°e.

Direction of Rotation: When the codewheel rotates counterclockwise, as viewed looking down on the module (so the marking is visible), channel A will lead channel B. If the codewheel rotates in the opposite direction, channel B will lead channel A.

Optical Radius (R_{op}): The distance from the codewheel's center of rotation to the optical center (O.C.) of the encoder module.

Absolute Maximum Ratings

Parameter	Symbol	Min.	Max.	Units	Notes
Storage Temperature	T _S	-40	85	°C	See Note
Operating Temperature	T _A	-40	85	°C	See Note
Supply Voltage	V _{CC}	-0.5	7	V	
Output Voltage	V _O	-0.5	V _{CC}	V	
Output Current per Channel	I _O	-1.0	5	mA	
Soldering Temperature			260	°C	t ≤ 5 sec.

Recommended Operating Conditions

Parameter	Symbol	Min.	Max.	Units	Notes
Temperature	T	-40	85	°C	
Supply Voltage	V _{CC}	4.5	5.5	V	Ripple < 100 mV _{pp}
Load Capacitance	C _L		100	pF	3.2 kΩ pull-up
Count Frequency			20	kHz	(Velocity (rpm) x N)/60

Note: The module performance is guaranteed to 20 kHz but can operate at higher frequencies. For operation at 30kHz for resolution options E, F, G and H, the maximum pulse width and logic state width errors are 40% and phase error is 15%.

Encoding Characteristics

Encoding Characteristics over Recommended Operating Range and Recommended Mounting Tolerances. These characteristics do not include codewheel/codestrip contributions.

Parameter	Symbol	Typ.	Case 1 Max.	Case 2 Max.	Units	Notes
Pulse Width Error	ΔP	7	30	40	%	
Logic State Width Error	ΔS	5	30	40	%	
Phase Error	Δφ	2	10	15	%	

Case 1: Module mounted on tolerances of ± 0.13 mm (0.005"). Case 2: Module mounted on tolerances of ± 0.25 mm (0.010")

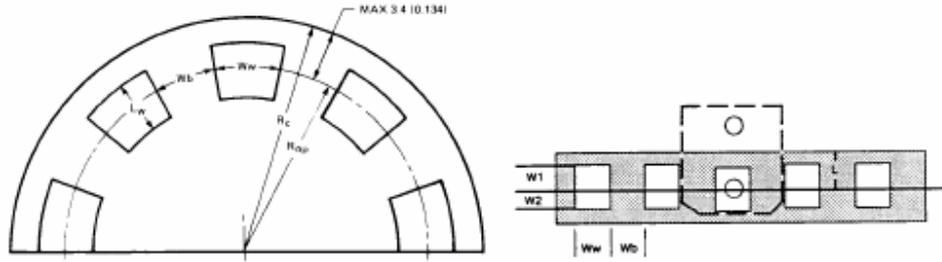
Note: See Figures in Mounting Considerations for details on Case 1 and Case 2 mounting tolerances.

Electrical Characteristics

Electrical Characteristics over Recommended Operating Range, Typical at 25°C.

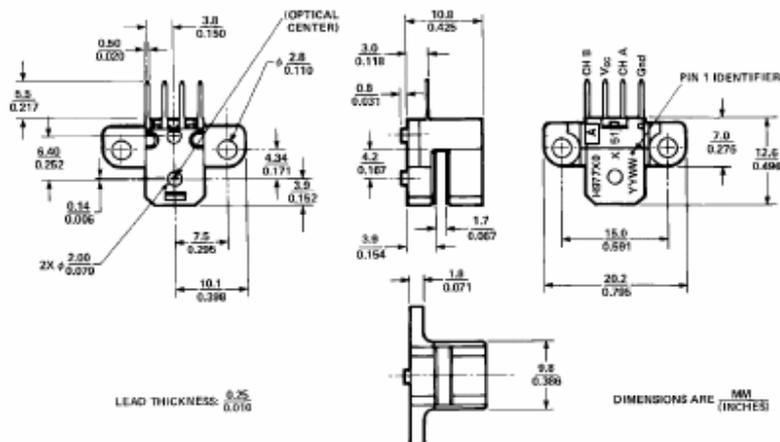
Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
Supply Current	I _{CC}		17	40	mA	
High Level Output Voltage	V _{OH}	2.4			V	I _{OH} = -40 μA
Low Level Output Voltage	V _{OL}			0.4	V	I _{OL} = 3.2 mA
Rise Time	t _r		200		ns	C _L = 25 pF, R _L = 11 kΩ
Fall Time	t _f		50		ns	C _L = 25 pF, R _L = 11 kΩ

Recommended Codewheel and Codestrip Characteristics



Parameter	Symbol	Min.	Max.	Units	Notes
Window/Bar Ratio	Ww/Wb	0.7	1.4		
Window Length (Rotary)	Lw	1.80 (0.071)	2.30 (0.091)	mm (inch)	
Absolute Maximum Codewheel Radius (Rotary)	Rc		Rop + 3.40 (Rop + 0.134)	mm (inch)	Includes eccentricity errors
Center of Post to Inside Edge of Window	W1	1.04 (0.041)		mm (inch)	
Center of Post to Outside Edge of Window	W2	0.76 (0.030)		mm (inch)	
Center of Post to Inside Edge of Codestrip	L		3.60 (0.142)	mm (inch)	

Optional Packages Available



Mounting Option #51 - Rounded Outline (Baseplane Mounting)



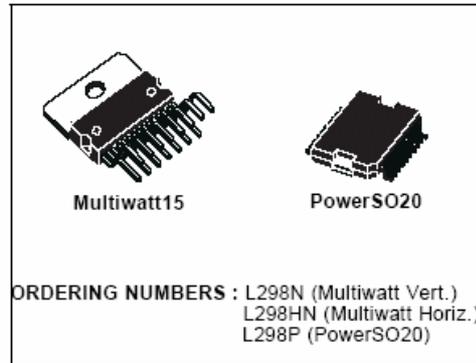
L298

DUAL FULL-BRIDGE DRIVER

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

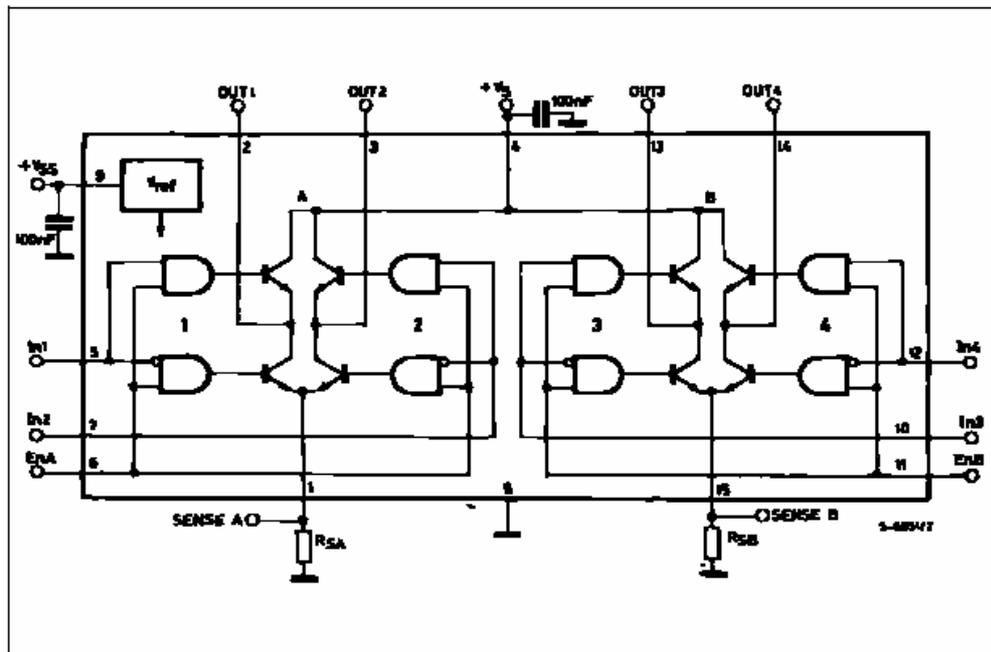
DESCRIPTION

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



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

BLOCK DIAGRAM

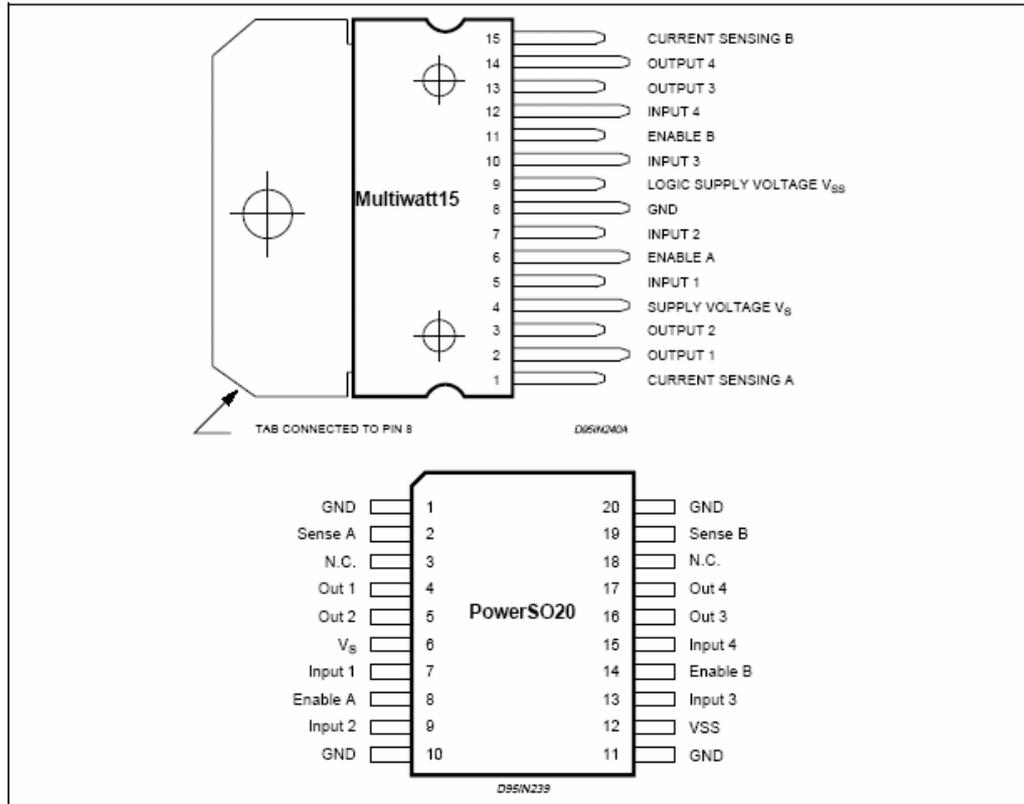


L298

ABSOLUTE MAXIMUM RATINGS

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

PIN CONNECTIONS (top view)



THERMAL DATA

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

(*) Mounted on aluminum substrate

Anexo H. Características Motores DC



[Global Sources Home](#)

Path: [All Categories](#) >> [Electronic Components](#) >> [globalsources.com - manufacturers](#) >> [electromechanical components](#) >> [Motors & Mechanisms](#) >> [DC motors](#)

Your Account

[Login / Register](#)
[Inquiry Basket](#)
[User Profile](#)

Search

[Products](#)
[Suppliers](#)
[Country](#)

For Buyers

[Product Alert](#)
[Sourcing Magazines](#)
[China Sourcing Reports](#)

For Suppliers

[Sell, List & Exhibit](#)

Trade Shows

[Trade Show Center](#)

Trade Information

[Travel, Cargo & News](#)
[Useful Contacts](#)

About Us

[Corporate Website](#)
[环球资源企业网](#)
[Partner With Us](#)

Help

[Help & FAQ](#)
[Contact Global Sources](#)
[Site Map](#)

Our Other Sites

[Global Sources Direct](#)
[EE Times - Asia](#)
[EE Times - India](#)
[电子工程专辑](#)
[国际电子商情](#)
[전자엔지니어링](#)
[電子工程專輯](#)
[世界经理人](#)
[电子系统设计](#)

[Bookmark this page](#)

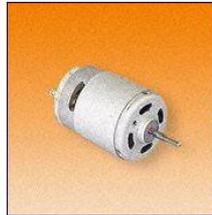
Kindly Industrial Co Ltd

Heavy-Duty Carbon Brush DC Motor

Platinum Supplier

Notify me when there are new [DC motors](#)

Product Details



[Larger photo](#)

Primary Competitive Advantages:

- Experienced Technical Staff
- Price

Main Export Markets:

- Worldwide

Heavy-Duty Carbon Brush DC Motor

KC380SP/KC385SP Heavy-Duty Carbon Brush DC Motor for Can Openers, Pepper Grinders, Vacuum Cleaners and More

[INQUIRE NOW](#) [ADD TO BASKET](#)

[View basket](#)

Key Specifications/Special Features:

- Typical applications: Electric power hand tools, radio control models, can openers, pepper grinders, vacuum cleaners, etc.
- Dimensions: 28.8(D) x 37.8(L)mm; Shaft length: 47mm, 51mm, 57mm
- KC385SP-3327
 - Operating range: 4.5 to 13.5V; Nominal: 12.0V
 - At free load: Speed: 25300rpm; Current: 0.60A
 - At max. efficiency: Speed: 21634rpm; Current: 3.541A; Torque: 134.74g-cm
 - Stall torque: 930 g-cm
- KC385SP-2560
 - Operating range: 6.0 to 24.0V; Nominal: 12.0V
 - At free load: Speed: 11300rpm; Current: 0.20A
 - At max. efficiency: Speed: 9500rpm; Current: 1.03A; Torque: 89.82g-cm
 - Stall torque: 550 g-cm
- KC380SP-21130
 - Operating range: 6.0 to 20.0V; Nominal: 12.0V
 - At free load: Speed: 9100rpm; Current: 0.15A
 - At max. efficiency: Speed: 7727rpm; Current: 0.85A; Torque: 62.99g-cm
 - Stall torque: 550 g-cm
- KC380SP-14250
 - Operating range: 9.0 to 30.0V; Nominal: 12.0V
 - At free load: Speed: 4800rpm; Current: 0.08A
 - At max. efficiency: Speed: 3798rpm; Current: 0.30A; Torque: 50.09g-cm
 - Stall torque: 240 g-cm
- For other requirements, please send us your inquiries including specifications

Products

More products from this category



[Automobile Washer Pump Motor](#)



[High-Quality DC Motor](#)



[Stamped Metal](#)



[Stamped Metal Brush DC Motor/Carbon Brush DC Motor](#)

[more >>](#)

Anexo I. Comandos de Usuario LM629

JUEGO DE COMANDOS DE USUARIO

Algunas de las órdenes pueden emitirse solos y algunos requieren una estructura de datos de apoyo. Como ejemplos, el comando STT "Inicio de Movimiento" (STArT motion) no requieren datos adicionales como si los necesita el comando LFIL "Cargar Parámetros del Filtro" (Load FILTer parameters) que requiere los parámetros del filtro (Intervalo de Muestreo del Término Derivativo y Parámetros del filtro). Los comandos son categorizados por su función: inicialización, control de interrupción, control del filtro, control de trayectoria, y reportes de datos.

Las órdenes se listan en la tabla 1 y se hace una breve descripción en los párrafos siguientes. Con cada nombre del comando se muestra el código del byte, y el número de datos que deben acompañarlo (bytes que serán escritos o leídos), y un comentario acerca de si la orden es ejecutable durante el movimiento.

Comando	Tipo	Descripción	Hex	Bytes por Dato	Nota
RESET	Inicializa	Restablece el LM628	00	0	1
PORT8	Inicializa	Selecciona la Salida del Puerto a 8-Bits	05	0	2
PORT12	Inicializa	Select 12-Bit Output	06	0	2
DFH	Inicializa	Define Inicio	02	0	1
SIP	Interrupción	Carga la Posición Índice	03	0	1
LPEI	Interrupción	Error de Posición Excesivo	1B	2	1
LPES	Interrupción	Parada por Error de Posición Excesivo	1A	2	1
SBPA	Interrupción	Punto de Señalización, Absoluto	20	4	1
SBPR	Interrupción	Punto de Señalización, Relativo	21	4	1
MSKI	Interrupción	Mascara de Interrupciones	1C	2	1
RSTI	Interrupción	Restablecimiento de Interrupciones	1D	2	1
LFIL	Filtro	Carga Parámetros del Filtro	1E	2 a 10	1
UDF	Filtro	Actualiza Filtro	04	0	1
LTRJ	Trayectoria	Carga Trayectoria	1F	2 a 14	1
STT	Trayectoria	Inicia Movimiento	01	0	3
RDSTAT	Reporte	Lee el Byte de Estados	Ninguno	1	1,4
RDSIGS	Reporte	Lee el Registro de Señales	0C	2	1
RDIP	Reporte	Lee la Posición del Índice	09	4	1
RDDP	Reporte	Lee la Posición Deseada	08	4	1
RDRP	Reporte	Lee la Posición Real	0A	4	1
RDDV	Reporte	Lee la Velocidad Deseada	07	4	1
RDRV	Reporte	Lee la Velocidad Real	0B	2	1
RDSUM	Reporte	Lee el Valor de la Integral	0D	2	1

Tabla 1. Comandos Generales del LM629

Note 1: Pueden ejecutarse las órdenes "Rápido" durante el movimiento.

Note 2: Los órdenes no son aplicables durante la ejecución del movimiento.

Note 3: La orden puede ejecutarse durante el movimiento si el parámetro de aceleración no fue cambiado.

Note 4: Los comandos no necesitan ningún código porque el byte de estado del puerto leído es totalmente apoyado por el hardware.

