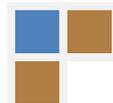
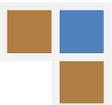


Capítulo 2: Componentes





2.1. Introducción

En este capítulo se muestran los componentes más destacados que han sido utilizados en la realización del presente proyecto y se hace una descripción de sus características principales, para finalmente obtener un robot de estructura paralela de 2 grados de libertad similar al que aparece en la ilustración 2.1.

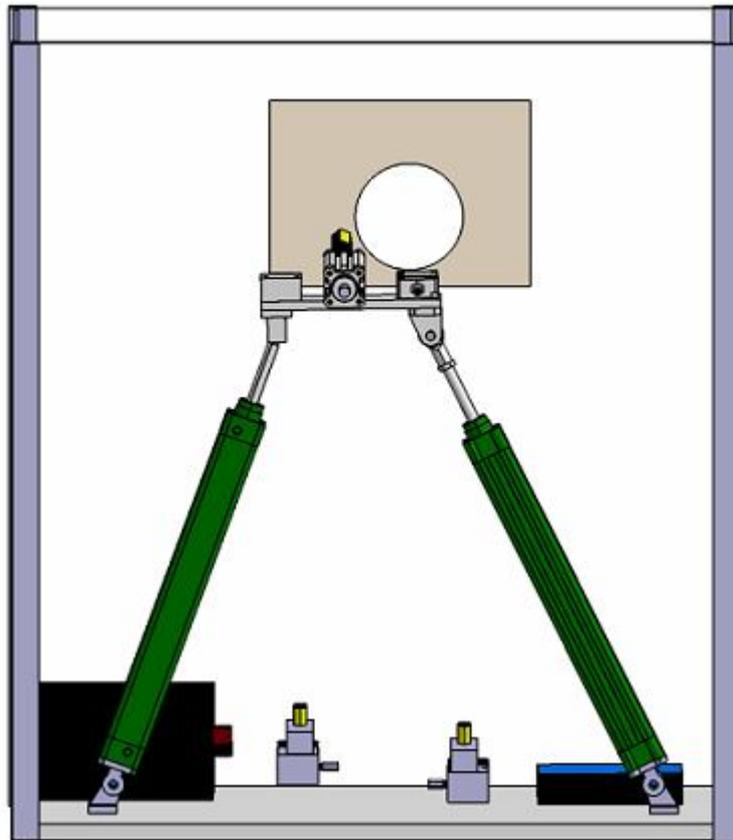


Ilustración 2. 1. Robot estructura paralela.

2.2. Elección de componentes

La elección de componentes será de vital importancia para conseguir unas buenas prestaciones, ya que se trata de una aplicación que requiere un alto grado de precisión pero, puesto que esto excede el campo de trabajo de este proyecto, no expondré por qué se han elegido unos componentes en lugar de otros, sino que paso directamente a la descripción de las características más destacadas de los mismos.

2.3. Descripción de componentes

2.3.1. Tarjeta de adquisición de datos.

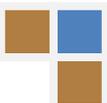
Inicialmente la tarjeta de adquisición de datos que iba a ser utilizada era el modelo *DT9812* de *Data Translation*, sin embargo finalmente he decido realizar el proyecto con el dispositivo de *National Instrument PCI-MIO16E-1*. Los motivos por los que he optado por un dispositivo en detrimento del otro están expuestos en el anexo A.4.

A continuación mostraré las características principales de los dos modelos con los que principalmente he trabajado.

2.3.1.a. Tarjeta DT9812

Se trata de un modelo de tarjeta con conexión al PC mediante puerto USB, la alimentación se hace a través del USB del ordenador. Para su uso debemos instalar primero el *CD ECON* incluido en el embalaje en el que venía la tarjeta; después, al conectar el cable al USB del ordenador, éste reconocerá el hardware y terminará su instalación. Antes de poder usar el *DT9812* o cualquier otro dispositivo de *Data Translation* en *Matlab* es necesario instalar el adaptador *DAQ Adaptor for Matlab* que sirve de interfaz entre el subsistema *Data Acquisition* de *Matlab* y la arquitectura *DT-Open Layers* de *Data Translation*.

Module	Analog Inputs	Analog Outputs	I/O Range	Sample Rate	Digital I/O
DT9812-10V	8 SE	2	± 10 V	50 kS/s	8 in/ 8 out



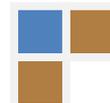
DT9812-2.5V and DT9812-10V Screw Terminal Assignments

Screw Terminal	Signal	Screw Terminal	Signal
20	USB +5 V Out	40	Ext Trigger
19	Ground	39	Ext Clock
18	Counter 0 In	38	Ground
17	Counter 0 Out	37	Digital Output 7
16	Counter 0 Gate	36	Digital Output 6
15	Ground	35	Digital Output 5
14	DAC 1	34	Digital Output 4
13	DAC 1 Return	33	Digital Output 3
12	DAC 0	32	Digital Output 2
11	DAC 0 Return	31	Digital Output 1
10	2.5 V Reference ^a	30	Digital Output 0
9	Analog Ground	29	Ground
8	Analog Input CH7	28	Digital Input 7
7	Analog Input CH6	27	Digital Input 6
6	Analog Input CH5	26	Digital Input 5
5	Analog Input CH4	25	Digital Input 4
4	Analog Input CH3	24	Digital Input 3
3	Analog Input CH2	23	Digital Input 2
2	Analog Input CH1	22	Digital Input 1
1	Analog Input CH0	21	Digital Input 0

Ilustración 2. 2. Terminales de conexión del dispositivo DT9812

2.3.1.b. Tarjeta PCI-MIO-16E-1

Esta es la tarjeta de adquisición de datos que se ha montado finalmente en el robot. La conexión al PC se efectúa mediante puerto PCI, de donde además obtiene la alimentación necesaria para la tarjeta.



AI 8	34	68	AI 0
AI 1	33	67	AI GND
AI GND	32	66	AI 9
AI 10	31	65	AI 2
AI 3	30	64	AI GND
AI GND	29	63	AI 11
AI 4	28	62	AI SENSE
AI GND	27	61	AI 12
AI 13	26	60	AI 5
AI 6	25	59	AI GND
AI GND	24	58	AI 14
AI 15	23	57	AI 7
AO 0	22	56	AI GND
AO 1	21	55	AO GND
AO EXT REF	20	54	AO GND
P0.4	19	53	D GND
D GND	18	52	P0.0
P0.1	17	51	P0.5
P0.6	16	50	D GND
D GND	15	49	P0.2
+5 V	14	48	P0.7
D GND	13	47	P0.3
D GND	12	46	AI HOLD COMP
PFI 0/AI START TRIG	11	45	EXT STROBE
PFI 1/AI REF TRIG	10	44	D GND
D GND	9	43	PFI 2/AI CONV CLK
+5 V	8	42	PFI 3/CTR 1 SRC
D GND	7	41	PFI 4/CTR 1 GATE
PFI 5/AO SAMP CLK	6	40	CTR 1 OUT
PFI 6/AO START TRIG	5	39	D GND
D GND	4	38	PFI 7/AI SAMP CLK
PFI 9/CTR 0 GATE	3	37	PFI 8/CTR 0 SRC
CTR 0 OUT	2	36	D GND
FREQ OUT	1	35	D GND

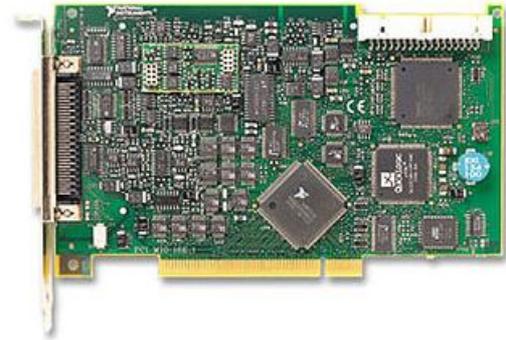


Ilustración 2. 3. Dispositivo PCI-MIO-16E-1.

Ilustración 2. 4. Terminales de conexión del dispositivo PCI-MIO-16E-1.

2.3.2. Cilindros normalizados de doble efecto.

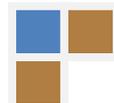
Elementos operativos y conexiones



- | | | | |
|---|--|---|---|
| 1 | Conexiones de aire comprimido | 6 | Vástago con cinta de medición magnética integrada |
| 2 | Ranuras para detector de proximidad | 7 | Doble plano para sostener el vástago durante el apriete |
| 3 | Conexión eléctrica del cabezal de medición | 8 | Anillo de centrado |
| 4 | Cabezal de medición | 9 | Conexión de tierra/masa |
| 5 | Tornillos con rosca interior para fijación | | |

Ilustración 2. 5. Elementos operativos y conexiones del cilindro DNCI-32-300-P-A.

El cilindro elegido es el modelo *DNCI-32-300-P-A* y se hará uso de dos de ellos que serán los encargados del movimiento del robot sobre el plano de trabajo, este sofisticado cilindro está pensado para transportar masas y lleva integrado un sensor de medición de desplazamiento tipo DNCI (normalmente el sensor es externo), la fijación se ha realizado con 4 tornillos en los agujeros roscados de la culata a la base del armario donde está colocado el robot y la unión al eje entre los dos cilindros, se realiza mediante una rótula para uno de los cilindros y roscada directamente a la pieza en el caso del otro cilindro; para el control de estos cilindros usaremos una válvula para cada uno, de tipo *MPYE-5-1/8-LF-010-B*.



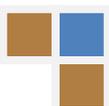
El sensor del cilindro genera una onda senoidal y otra cosenoidal y mediante la diferencia de fase entre estas dos señales se consigue conocer la posición del cilindro, luego será necesario el uso de convertidores *DADE-MVC-010* que aumentarán y traducirán esta señal a una señal lineal de 0 a 10 V que simplifica enormemente el uso de la señal adquirida ya que no será necesario hacer la conversión por "software" a valores lineales lo que permite además ahorrar recursos de computación, y permite también disminuir el número de conexiones a la tarjeta de adquisición de datos, de cuatro entradas diferenciales a sólo dos por cilindro.

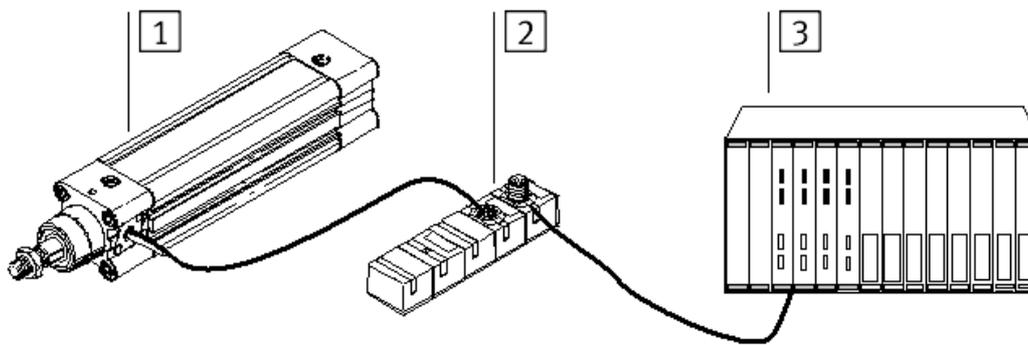
Pin	Color de cable	Designación	Asignación de pines (vista de la clavija)
1	negro	+Ub	
2	marrón	0 V	
3	rojo	Señal seno +	
4	naranja	Señal seno -	
5	verde	Señal coseno -	
6	amarillo	Señal coseno +	
7	Apantallamiento/ blindaje	Apantallamiento/ tierra	
8	-	n.c.	

Ilustración 2. 6. Cable del sensor (S2).

2.3.3. El convertidor de valor medido.

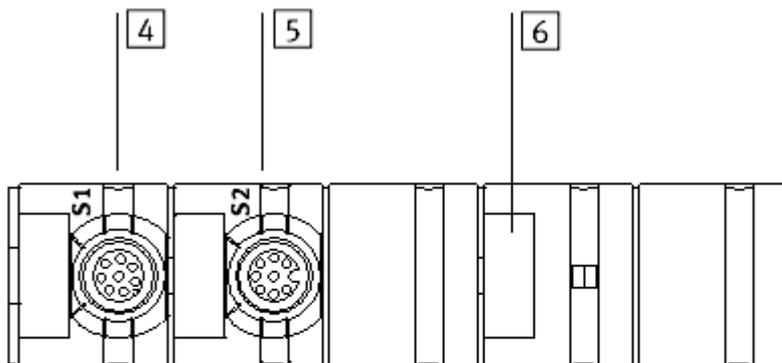
Realiza la conversión de pulsos digitales del sistema de medida DNCl en una señal analógica de 0 a 10 V. Realiza la función de adaptador de señal entre el sensor del cilindro y la tarjeta de adquisición de datos.





- 1 Actuator DNCI-... con vástago y encoder de desplazam. integrado
- 2 Convertidor de valor medido DADE-MVC
- 3 PLC/IPC

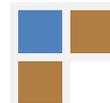
Conexiones y elementos operativos



- 4 S1: Conexión para PLC (con LED verde/rojo)
- 5 S2: Conexión para el encoder de desplazamiento del DNCI-... (con LED amarillo/rojo)
- 6 Etiquetas de identificación ISB-8x20 (accesorios)

Ilustración 2. 7. Conexiones y elementos operativos del convertidor DADE-MVC-010.

Son necesarios dos de estos convertidores *DADE-MVC-010* para controlar los dos cilindros *DNCI-32-300-P-A*.



Pin	Color	Asignación	Conexión S1
1	blanco	+24 V alimentación	
2	marrón	Señal medida (analógica)	
3	verde	Salida de referencia	
4	amarillo	0 V de la señal medida	
5	gris	Entrada de referencia	
6	rosa	Entrada de calibración	
7	azul	Salida Ready (preparado)	
8	rojo	0 V alimentación y entradas/salidas	

Ilustración 2. 8. Cable del convertidor (S1).

2.3.4. Válvulas de control proporcional

La señal analógica de entrada a la válvula permite una regulación progresiva del caudal volumétrico (mediante función progresiva de estrangulamiento) y permite también cambiar la dirección del caudal volumétrico. La válvula sirve al mismo tiempo para regular la velocidad como para posicionar el cilindro. Se trata de una válvula de

5/3 vías.



Se requieren dos unidades de válvulas *MPYE-5-1/8-LF-010-B* una para controlar cada uno de los dos cilindros normalizados de doble efecto.

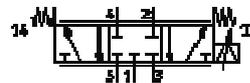
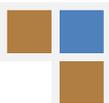


Ilustración 2. 9. Válvula MPYE-5-1/-LF-010-B.



2.3.5. Cilindro de doble efecto con vástago pasante

La elección de válvulas digitales para el control del posicionamiento del cilindro encargado de efectuar la presión del lápiz sobre el papel hace que sea muy conveniente que el cilindro encargado de esta tarea sea un cilindro de vástago pasante para que así, al introducir aire a la misma presión por ambos lados del embolo del cilindro, éste permanezca quieto, lo que no ocurre para cilindros de vástago pasante ya que la superficie del émbolo por cada lado es distinta, sabiendo que la presión es igual al producto de la fuerza aplicada por la superficie en que se aplica $P = F * A$, es evidente que los cilindros normales a igual presión desplazarán el vástago hacia el exterior.



Ilustración 2. 10. Cilindro DNC-32-100-PPV-A-S2.

En caso de que en el futuro sustituyamos las válvulas todo-nada por una válvula proporcional el uso de un cilindro de vástago pasante u otro normalizado es indiferente, porque la sustitución de las válvulas no implica realizar ningún otro desembolso añadido al de la compra de la válvula proporcional.

El cilindro elegido para efectuar la presión de la herramienta sobre el papel es el *DNC-32-100-PPV-A-S2* que a diferencia de los cilindros DNCl mostrados en el apartado 2.1.2. éste no posee ningún sistema de medida integrado, así que usaremos un dispositivo externo el *SMAT-8E-S50-IU-M8*.

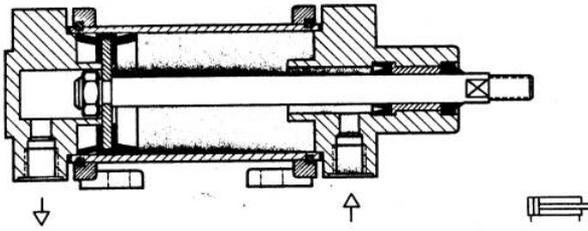


Ilustración 2. 12. Cilindro normalizado de doble efecto.

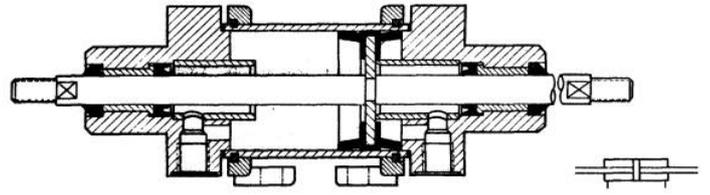


Ilustración 2. 11. Cilindro de doble efecto con vástago pasante.

2.3.6. Transmisor de posición para el cilindro de vástago pasante.

Se trata de un sensor *SMAT-8E-S50-IU-M8* que realiza la detección de la posición del cilindro de manera magnética y transmite la señal de salida a través de 2 salidas analógicas, una de 0 a 10 V y otra de 4 a 20 mA. (Usaremos únicamente la salida de 0 a 10 V).

La señal de salida es proporcional a la posición siempre que ésta esté dentro del campo de medición. En caso de que el cilindro esté fuera del campo de medida se encenderá un led de color rojo.

El sensor se ha anclado al cilindro mediante la colocación dentro de la ranura en T y puede desplazarse para ajustarse al rango de funcionamiento deseado.



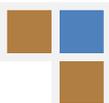
Ilustración 2. 13. SMAT-8E-S50-IU-M8.

2.3.7. Válvula reguladora de presión.

Utilizaremos la válvula reguladora LFR-D-MINI, que regula la presión y compensa las fluctuaciones a la entrada; además, efectúa el filtrado de las partículas de suciedad y el condensado de agua por lo que se trata de una unidad muy compacta y completa.



Ilustración 2. 14. LFR-D-MINI.



Especificaciones técnicas

Presión de entrada p1 max.	16 bar (sin purga automática de condensados) 12 bar (con purga automática de condensados)
p1 min	min. p1 1,5 bar; 1 bar (sólo LR-...)
Margen de presión de trabajo máx. p2	de 0,5 a 7 bar (con LR-/LFR-...- D7 -...) de 0,5 a 12 bar (con LR-.../LFR-...- D -...)
Margen de temperaturas:	-10° C ... +60° C (Almacenaje, fluído, ambiente) +1,5° C ... +60° C (Temp. del fluído con LFM...-...) +1,5° C ... +30° C (Temp. del fluído con LFX...-...)
Posición de montaje	Vertical ($\pm 5^\circ$); indiferente (sólo LR-...)
Conexiones del manómetro	G1/8 (en LR-/LFR-... MINI -...) G1/4 (en LR-/LFR-... MIDI -/ MAXI -...)
Finura del filtro	40 μm (con LF-/LFR-...- D -...); 5 μm (LF-/LFR-...- D-5M -...); 1 μm (con LFMB-...- D -...); 0,01 μm (LFMA-/ LFMBA -...- D -...)
Fluido	Aire comprimido: - filtrado a finura de $\leq 40 \mu\text{m}$ con LR-... - no lubricado, filtrado a finura de $\leq 5 \mu\text{m}$ con LFMB-.../ $\leq 1 \mu\text{m}$ con LFMA-.../ $\leq 0,01 \mu\text{m}$ con LFX-...
Materiales: Cuerpo: GD-Zn; brida de conexión: Al, GD-Zn; protección: Al; vaso: PC (macrolone); piezas internas: POM, PA; pomo de ajuste de la presión: PA ; medio del filtro: PE (40 μm , 5 μm), microfibras (1 μm y 0,01 μm); carbón activo (LFX-...-AC); juntas: NBR	

Ilustración 2. 15. Especificaciones técnicas LFR-D-MINI.

2.3.8. Válvulas MOFH-3-1/8.

Por medio de un accionamiento eléctrico externo y neumático, la válvula presuriza ramales de aire comprimido en sistemas

neumáticos.

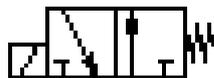
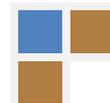


Ilustración 2. 16. Válvula neumática MOFH-3-1/8.



2.3.9. Transformador de tensión.

Para la alimentación eléctrica de todos los componentes, excepto la tarjeta de adquisición de datos, que es alimentada a través del puerto PCI del ordenador, necesitaremos una fuente capaz de suministrar 24 V de corriente continua c.c. y al menos 3 A de intensidad (ver tabla de consumo).

Consumo de los componentes		
Convertidor DADE-MVC-010.		2 X 25* mA = 50 mA
Solenoides MOFH-3-1/8.	4,5 W*	2 X 187 mA = 375 mA
Válvulas proporcionales MPYE 5-1/8-LF-010-B	20 W*	2 X 833 mA = 1.66 A
Sensor SMAT-8E-S50-IU-M8.		24* mA
Total:		2,109 A

Tabla 2. 1. Consumo de los componentes.

*Datos obtenidos de las especificaciones técnicas de los componentes (incluidas tanto al final de la memoria descriptiva como en el CD-ROM).

2.3.10. Otros elementos utilizados.

Tubos flexibles, distribuidores múltiples, rácores, silenciadores y tapones para las válvulas, son otros elementos importantes empleados para el conexionado neumático del robot.

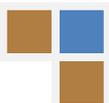


2.4. Software

Para la realización del proyecto se ha necesitado hacer uso de las siguientes aplicaciones informáticas:

- Controladores de la tarjeta de adquisición: ni-daq 7.3

Viene incluido al adquirir el producto, es necesario instalarlo para su correcto funcionamiento y para un posible mantenimiento del producto.



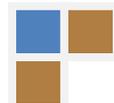
- *Matlab*: versión 7.1 o superior.

El código necesario ha sido realizado en m-files y no se ha usado el *Simulink* de *Matlab* ya que sólo dispone de bloques para unos pocos modelos de tarjetas y la nuestra no se encuentra entre ellas, en caso de poder usarlo si se dispone de un modelo con un bloque ya creado en *Simulink*, se podrá mejorar el código creado y facilitar el diseño del regulador, pero en muchos casos la solución es realizar todo directamente en *Matlab* y usar *Simulink* únicamente como una herramienta de apoyo, con la que podemos simular el diseño una vez hallada la función de transferencia del sistema de forma aproximada.

Matlab dispone de una multitud de *toolboxes* que facilitan la creación de las aplicaciones ya que incluyen numerosas funciones que nos ahorrará tener que implementarlas, en el código para el funcionamiento del robot he hecho uso principalmente del *Data Acquisition toolbox* que lleva implementadas funciones para la adquisición de datos, esta herramienta ha sido utilizada entre otras cosas para la lectura de las entradas analógicas y salidas tanto analógicas como digitales.

También he utilizado el *GUIDE* de *Matlab* para la creación de *GUI's* (*Grafical User Interface, interfaz grafica de usuario*), con esta herramienta se genera fácilmente el aspecto de nuestra interfaz así como las funciones para cada uno de los botones de distinto tipo, barras deslizantes o textos dinámicos que deseemos y luego tendremos que abrir el m-file con el mismo nombre de la figura creada y realizar el código que deseemos para cada uno de los botones.

Por último, todo el código ha sido compilado con el compilador interno de *Matlab*, de este modo el programa podrá ser utilizado en cualquier ordenador, aunque no disponga de *Matlab*. Esto unido a la facilidad de uso que se ha pretendido conseguir con la interfaz gráfica permite también que cualquier usuario pueda utilizar el robot aunque no conozca el funcionamiento de *Matlab*.



Una vez finalizada la aplicación que controla el robot, será compilada con el compilador de *Matlab*, con lo que se podrá usar incluso en ordenadores que no dispongan de este programa, que requiere por lo general un ordenador de grandes prestaciones (dependiendo esto de la versión), de esta forma la aplicación se podrá ejecutar incluso en ordenadores con recursos computacionales y de almacenamiento bastante modestos.

- Código propio generado como *M-files* y *Scripts* de *Matlab* para el funcionamiento del robot.

- Orcad Capture.

Con este programa he realizado el esquemático del circuito eléctrico que sirve para activar las electroválvulas y también el interruptor de accionamiento eléctrico utilizado para introducir la señal necesaria que establece la referencia de los dos cilindros *DNCI*.

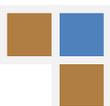
- PCB Wizard – Professional Edition.

Utilizado para realizar las vistas simuladas de los circuitos que aparecen en la memoria.

- Autocad 2007

Ha sido necesario usar este programa para la realización de los anexos y como complemento a Fluidsim, algo limitado en cuanto a componentes eléctricos se refiere, para la realización de los anexos.

- Festo Fluidsim.



En lo que concierne a este proyecto, junto con Autocad, ha simplificado considerablemente la realización de esquemas neumáticos que aparecen en los anexos de la memoria.

- Diferentes programas editores de texto así como editores de imágenes y video.

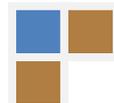
2.4.1. Matlab

MATLAB[®]
The Language of Technical Computing



2.4.1.1. Antecedentes históricos

El nombre *MATLAB* proviene de "MATrix LABoratory" (Laboratorio de Matrices). *Matlab* fue escrito originalmente para proporcionar un acceso sencillo al software matricial desarrollado por los proyectos *LINPACK* y *EISPACK*, que juntos representan lo más avanzado en programas de cálculo matricial.



2.4.1.2 Motivos para usar Matlab

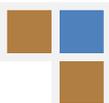
Con su amplio rango de herramientas para modelar sistemas de control, análisis, simulación y procesamiento de prototipos, *Matlab* es el sistema ideal para desarrollar sistemas avanzados de control.

Matlab es un entorno de cálculo técnico de altas prestaciones para cálculo numérico y visualización. Integra:

- Análisis numérico
- Cálculo matricial
- Procesamiento de señales
- Gráficos

en un entorno fácil de usar, donde los problemas y las soluciones son expresados como se escriben matemáticamente, sin la programación tradicional *Matlab* es un sistema interactivo cuyo elemento básico de datos es una matriz que no requiere dimensionamiento. Esto permite resolver muchos problemas numéricos en una fracción del tiempo que llevaría hacerlo en lenguajes como *C*, *Basic* o *Fortran*. *MATLAB* ha evolucionado en los últimos años a partir de la colaboración de muchos usuarios. En entornos universitarios se ha convertido en la herramienta de enseñanza estándar para cursos de introducción en álgebra lineal aplicada, así como cursos avanzados en otras áreas. En la industria, *Matlab* se utiliza para investigación y para resolver problemas prácticos de ingeniería y matemáticas, con un gran énfasis en aplicaciones de control y procesamiento de señales. *Matlab* también proporciona una serie de soluciones específicas denominadas *toolboxes*. Éstas son muy importantes para la mayoría de los usuarios de *Matlab* y son conjuntos de funciones *Matlab* que extienden el entorno *Matlab* para resolver clases particulares de problemas como:

- Procesamiento de señales
- Diseño de sistemas de control
- Simulación de sistemas dinámicos
- Identificación de sistemas



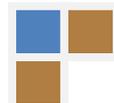
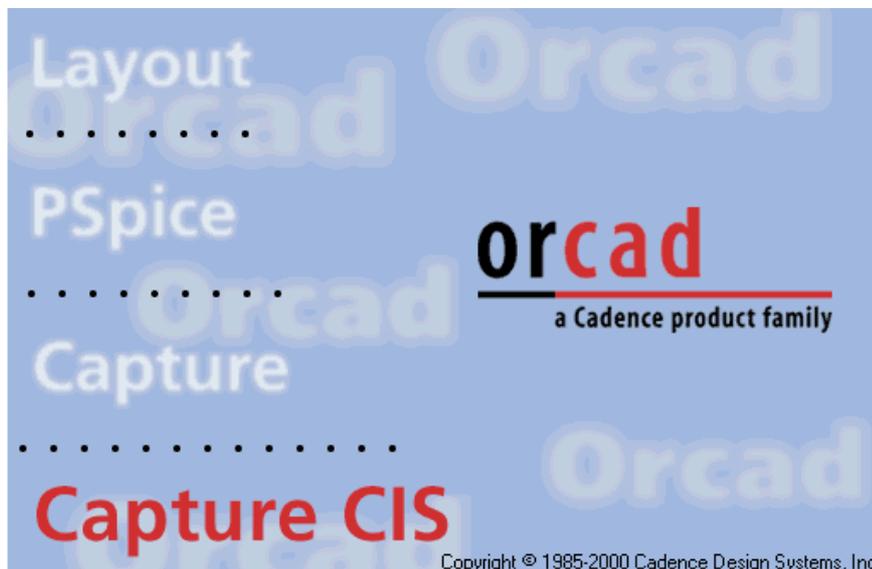
- Redes neuronales y otros.

Probablemente la característica más importante de *Matlab* es su capacidad de crecimiento. Esto permite convertir al usuario en un autor contribuyente, creando sus propias aplicaciones. En resumen, las prestaciones más importantes de *Matlab* son:

- Escritura del programa en lenguaje matemático.
- Implementación de las matrices como elemento básico del lenguaje, lo que permite una gran reducción del código, al no necesitar implementar el cálculo matricial.
- Implementación de aritmética compleja.
- Un gran contenido de órdenes específicas, agrupadas en *toolboxes*.
- Posibilidad de ampliar y adaptar el lenguaje, mediante ficheros de script y funciones .m.

La versión de *Matlab* utilizada ha sido la versión 7.1. (Release 14) Service Pack 3 que nos ofrece compatibilidad con muchos de los dispositivos de adquisición de datos actuales, así como la posibilidad de crear interfaces gráficas de una manera más intuitiva y con mayores posibilidades que versiones anteriores. Sin embargo, el código puede ejecutarse sin problemas en versiones anteriores.

2.4.2. Orcad Capture

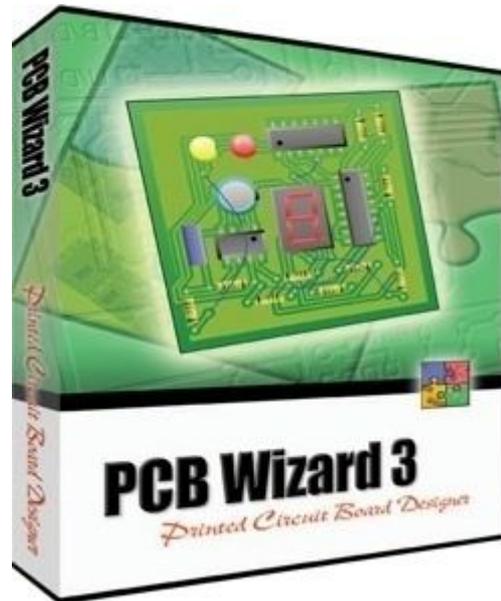


Una de las más completas aplicaciones dedicadas al diseño de circuitos electrónicos, permite realizar esquemas eléctricos, diseño de placas de circuito impreso y simulaciones electrónicas.

2.4.3. PCB Wizard

Este programa nos permite realizar una vista simulada del aspecto real de nuestros circuitos; así mismo, también permite realizar el diseño de las pistas y pads del circuito.

Se trata de una aplicación muy sencilla e intuitiva y que da muy buenos resultados para circuitos de una complejidad media. En caso de circuitos con muchas pistas y con componentes no normalizados o poco habituales es mejor utilizar el *layout* de *Orcad*.

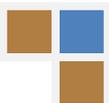


2.4.4. Autocad

Utilizado como complemento al programa *fluidsim* en la creación de los símbolos normalizados que no están disponibles en el programa de *Festo*, especialmente en los componentes eléctricos. Se trata de un programa muy utilizado en todo tipo de proyectos ya que permite realizar planos en 2D y/o 3D de los componentes deseados.

2.4.4.1. Evolución histórica de los sistemas CAD.

El CAD ha constituido un hito para el sector de la ingeniería, la arquitectura y la construcción, especialmente porque eliminó la necesidad de dibujar los planos a mano, permitiendo además incorporar los cambios con facilidad. Además, el CAD ha proporcionado una herramienta relativamente simple para la visualización 3D. La elaboración manual de perspectivas 3D se basaba antaño en técnicas de dibujo lentas



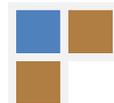
y laboriosas, que además no eran interactivas. El *CAD* ha alterado, por consiguiente, la naturaleza misma, definición y alcance del proceso de diseño [3]. A continuación se presenta una breve historia del *CAD*, desde la década de los años 50, cuando apareció el primer programa gráfico, hasta la década de los 90.

Antes de 1970. El primer *CAD* data de los años 50 para las Fuerzas Aéreas de USA. El primer sistema de gráficos, el *SAGE* (Semi Automatic Ground Environment) un sistema de defensa aérea, que fue empleado para visualizar datos de radar, fue desarrollado en colaboración con el *MIT*. En los 60, los sistemas *CAD* se utilizaron para diseñar espacios interiores de oficinas. En 1968 estaban ya disponibles los sistemas *CAD 2D* (muy básico, tal y como lo entendemos hoy en día). Estos sistemas funcionaban en terminales de grandes ordenadores (mainframes).

Años 70. A principios de esta década varias compañías empezaron a ofrecer sistemas de diseño/dibujo automatizado. Muchos de los productos y firmas más conocidas en la actualidad tuvieron sus inicios en este periodo. Algunos de estos nombres incluyen *CATIA* y *CADLink*. Podían encontrarse ya algunas capacidades 3D en programas de cálculo de sistemas *HVAC* (Heating, Ventilation and Air Conditioning). A finales de los 70, un sistema típico de *CAD* consistía en un mini-ordenador de 16 bits con un máximo de 512 Kb de memoria y de 20 a 300 Mb de disco duro.

Años 80. *Autodesk* entra en escena con el objetivo de crear un programa de *CAD* que funcione sobre un PC. En poco tiempo *Autocad* llegó a ser el programa más popular de *CAD*. Muchos otros programas de compañías diversas siguieron la misma senda. Durante esta década, los programas de *CAD* se utilizaban básicamente para desarrollos de ingeniería. Empiezan a desarrollarse los sistemas *GIS* (Geographical Information Systems).

Años 90. Se generalizan las visualizaciones en 3D. *Autocad versión 12* se convierte en el programa de *CAD* sobre *Windows* más vendido. A mediados de los 90 aparecen muchos programas de *CAD* para una gran variedad de usos y aplicaciones. A

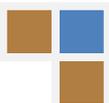


finales de los 90 mucha gente utiliza ya los programas de *CAD* de forma habitual, pero hay todavía una gran lucha por atraer la atención de los usuarios. Se realizan programas mejores para satisfacer las necesidades crecientes de la industria. Se desarrollan también muchos programas sencillos de *CAD*. Los programas de *CAD 3D* abundan en el mercado. Se ofrecen soluciones a segmentos verticales aportando soluciones específicas para cada uno de ellos (construcción, ingeniería civil, mecánica y fabricación, etc.).

2.4.5. Festo fluidsim

Entre las características más importantes se encuentran:

- Ofrece la posibilidad de crear y simular circuitos de electroneumática.
 - Posee una biblioteca de componentes neumáticos y eléctricos muy completa para la creación de circuitos nuevos.
 - Durante la animación, los conductos y cilindros aparecen coloreados, según el estado de cada uno.
 - El usuario puede controlar la velocidad de la simulación (con funciones como paso a paso, pausa...).
 - Las válvulas y los interruptores accionados manualmente pueden ser conmutados mediante un clic del ratón.
 - Permite construir circuitos nuevos, pero también se dispone de un conjunto de circuitos ejecutables.
 - Muestra en los aparatos de medición los valores exactos de presión, fluido y corriente.
 - Guarda una proporción del tiempo real durante la animación.



- Ofrece también el apoyo necesario para el conocimiento básico de la neumática, en un menú denominado *"Didáctica"*, que puede ser muy útil a la hora de explicar y entender los distintos contenidos relacionados con este tema.

-Contiene un apartado denominado *"Descripción del componente"*, donde explica cómo son los distintos componentes de los circuitos, muestra una foto, explica su funcionamiento e ilustra incluso cómo es el componente interiormente. Posee también animaciones que pueden ser de gran ayuda para entender el tema seleccionado, etc.

-En *"Exposiciones"* presenta distintos temas disponibles, ejercicios

-Posee también películas didácticas.

