



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 0 – Miércoles 19 de Abril 2006

Introducción

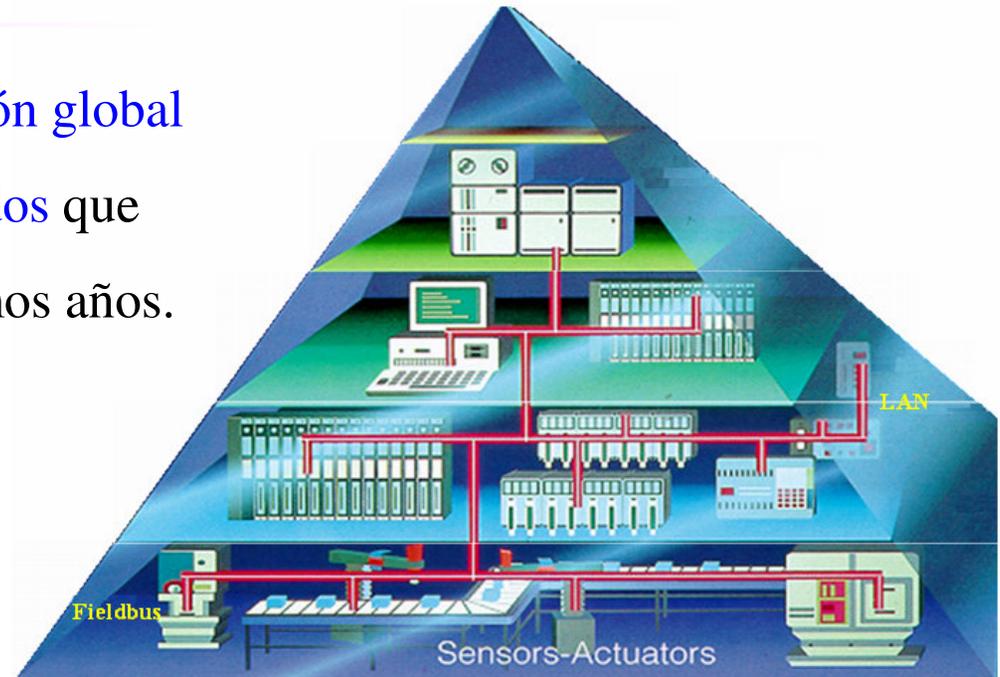


Objetivos:

Este curso pretende presentar una **visión global** de los **sistemas de producción integrados** que constituirán las factorías en los próximos años.

● **Temas Principales**

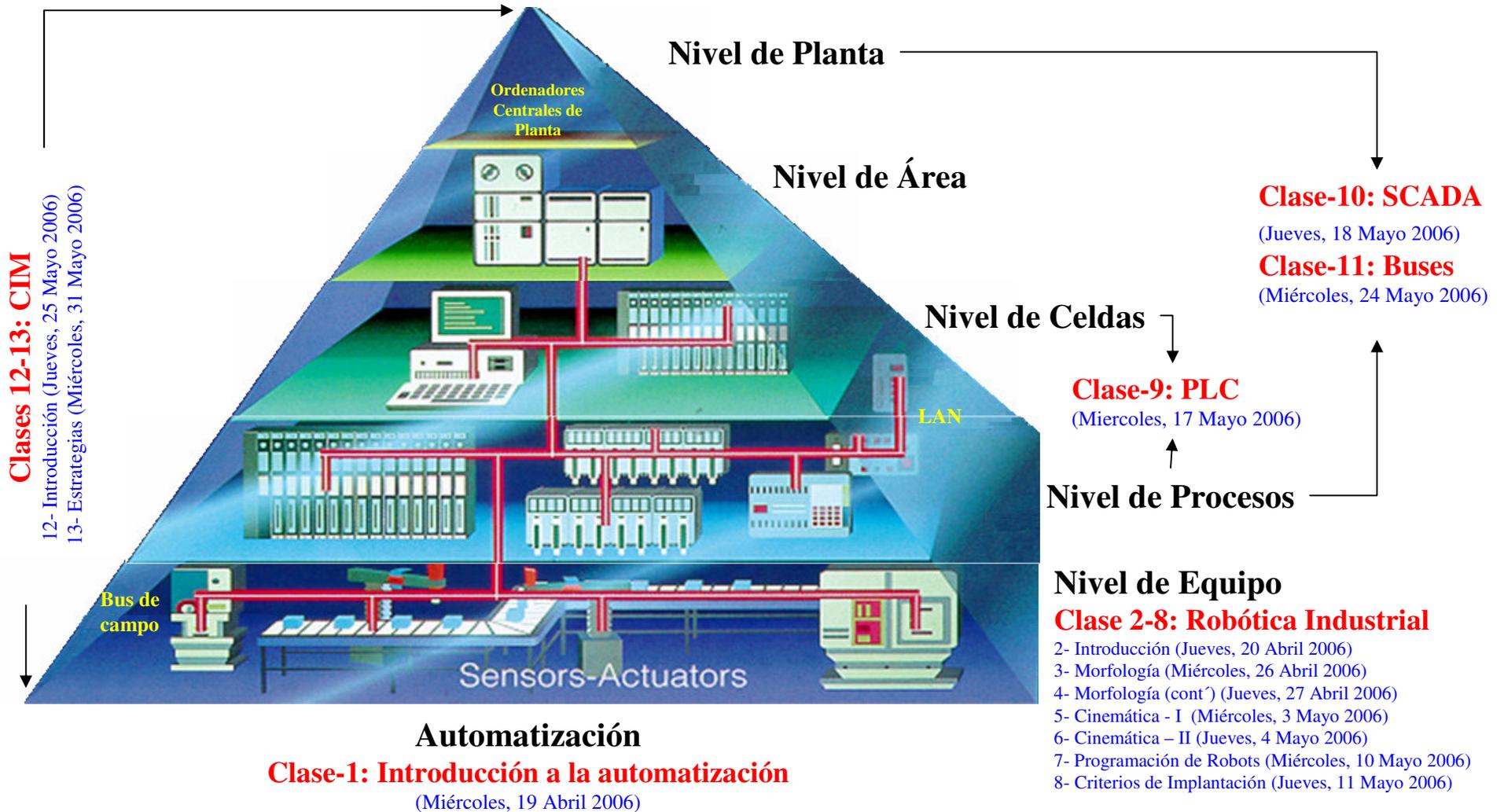
- ▶ Automatización
- ▶ Robótica Industrial
- ▶ Autómatas Programables (PLCs)
- ▶ Esquemas de comunicación
- ▶ SCADA
- ▶ CIM





Automatización Industrial -II

Programa:





Prácticas:

- **Visión Artificial**
3 y 5 Abril 2006
- **Simulación en Robótica**
24 y 26 Abril 2006
- **El Lenguaje RAPID**
8 y 10 Mayo 2006
- **Uso de los robots**
15, 17 y 22 Mayo 2006



Bibliografía:

● Libros

- A. Barrientos et al. *Fundamentos de Robótica*. McGraw-Hill, 1997
- A. R. Regh and H. W. Kraebber. *Computer-Integrated Manufacturing*. Prentice Hall, 2001.
- T. Chang, R. Wysk and H. Wang. *Computer-Aided Manufacturing*. 2nd ed., Prentice Hall, 1998.
- R. Rembold, B. Nnaji and S. Storr. *Computer Integrated Manufacturing and Engineering*. Addison-Wesley, 1993.
- J. J. CRAIG. *Introduction to Robotics. Mechanics and Control*. 2nd edition. Addison-Wesley, 1986
- A. W. SCHEER. *CIM-Toward the Factory of the Future*. Springer Verlag, 1991.

● Cursos Online

- MIT's OpenCourseWare, 2.854 Manufacturing Systems I (SMA 6304), Fall 2004, available at: <http://ocw.mit.edu/OcwWeb/Mechanical-Engineering>

● Apuntes

- Dr. Eladio Dapena, Automatización Industrial – II, Curso 2004-2005, UC3M.
- Prof. Darek Ceglarek, IE 605 Computer Aided Manufacturing, University of Wisconsin

● Otros

- Automation website: <http://www.automation.com/>
- The Society of Manufacturing Engineers: <http://www.sme.org>
- Journal of Manufacturing Science and Engineering: <http://scitation.aip.org/ASMEJournals/Manufacturing/>



Profesor:

Alaa Khamis

- **Email:** akhamis@ing.uc3m.es
akhamis@pami.uwaterloo.ca
 - **URL:** <http://roboticslab.uc3m.es>
 - **Despacho:** 1.3B12
 - **Tutoría:** - Martes 12:00-13:30H
- Después de la clase
-



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 0 – Miércoles 19 de Abril 2006

Introducción



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 1 – Miércoles 19 de Abril 2006

Introducción a la Automatización



Automatización:

- **Automatización Industrial**

Automatización es una palabra griega específica para hacer máquinas o dispositivos capaces de imitar las funciones y los movimientos de los seres vivos.

Automatización industrial es el uso de ordenadores para controlar las maquinarias industriales y procesos remplazando los operadores humanos.

- **Sistema automático**

Sistema automático es aquel capaz de tomar decisiones inteligentes, de manera autónoma, basado en el conocimiento que tiene de si mismo y del proceso que gobierna en una situación dada

- **Términos asociados**

Autónomo: Sistema que puede tomar decisiones.

Autoajutable: Selecciona estrategias de acuerdo a su estado.

Inteligente: Utiliza modelos inteligentes para evaluar alternativas y tomar decisiones



Automatización de procesos:

- **Automatización de procesos**

“Es una **secuencia**, u **orden definido**, de actividades químicas, físicas o biológicas que se llevan a cabo para la conversión, transporte o almacenamiento de material o energía”. Definición según normativa S88 de la ISA (Instrumentation Society of America)

- **Procesos Industriales**

- ▶ **Proceso *Batch***

Cuando el resultado de un proceso industrial es una cantidad finita de material, hablamos de proceso *batch*.

- ▶ **Proceso *Continuo***

En caso contrario se obtiene un flujo continuo de material y entonces hablamos de procesos *continuos*.

La **automatización**, entonces, se establece como **forma de ordenación a partir de la secuenciación automática de tareas y regulación de variables** para que sigan consignas impuestas.



¿Porqué automatizar?

● **Objetivos Económicos**

▶ **Realidad**

▪ **Una sociedad abierta**

- Precios, productos, costos de producción son conocidos por todos los participantes.
- Los consumidores seleccionan el mejor producto al mejor precio.
- La competencia tiene nuevos métodos de producción.
- La competencia responde rápidamente ante requerimientos del mercado.

▪ **Reingeniería del proceso costosa**

- Infraestructura difícil de cambiar



¿Porqué automatizar?

● **Objetivos Económicos**

▶ **Salida**

▪ **Mejorar la cadena productiva**

- Mejorar los procesos de transformación de la materia.
- Disminuir los costos de producción.
- Eliminar desperdicios.

▪ **Manejar la información del mercado**

- Determinar los mejores proveedores.
- Determinar precios y calidad esperada, oferta, demanda.

▪ **Responder eficientemente ante cambios**

- Ajustes internos en la cadena productiva.
- Manejo eficiente de los procesos.



¿Porqué automatizar?

● **Ingeniería**

▶ **Seguridad de las operaciones**

- Evitar daños al hombre, ambiente, equipos.
- Mantener la operación de manera segura y continua.

▶ **Comodidad de la operación**

- Facilitar la operación del hombre.
- Evitar labores tediosas.

▶ **Conocer mejor el proceso**

- Analizar los datos.
- Generar modelos: *¿Qué pasa si?*
¿Cómo hacer?
¿Qué hacer?



¿Cómo automatizar?

La **respuesta** de un sistema está determinada por las **leyes internas del sistema**.

La respuesta puede ser inducida de diferentes maneras:

- **Clasificación**

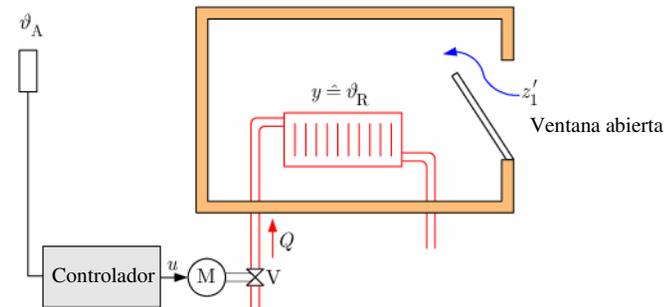
- ▶ **Mecanizados**

Realizan tareas repetitivas, no varían ante diferentes entradas.



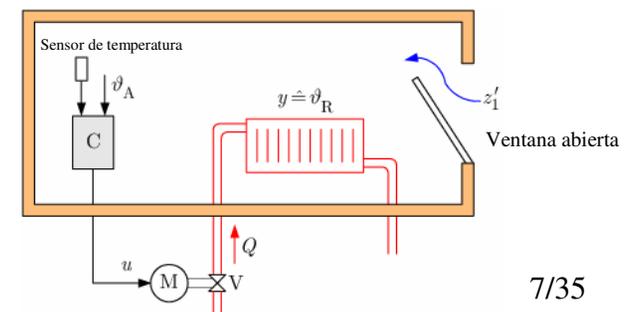
- ▶ **Sistemas a lazo abierto**

No presentan control. Su respuesta depende de las entradas.



- ▶ **Sistemas a lazo cerrado**

Las entradas son manipuladas con el fin de que el sistema de una respuesta esperada.





Evolución de la información para automatización

● Datos

Valores de variables de un proceso o del ambiente donde se desarrolla el proceso.

● Información

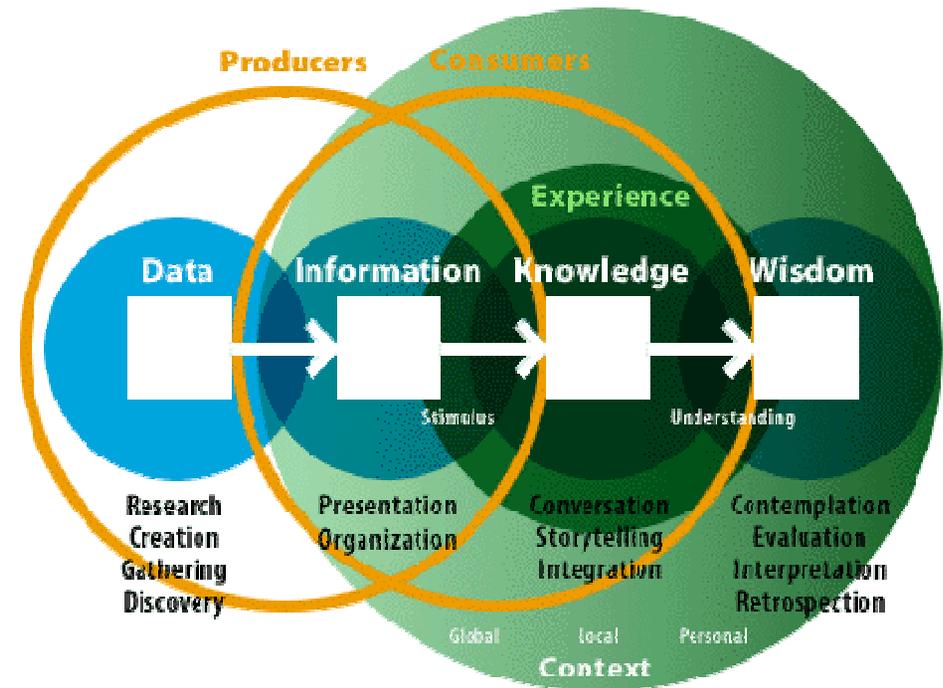
La selección de los datos, ubicación dentro de un contexto, suministran información que ayudan en la toma de decisiones.

● Conocimiento

Asociado a los mecanismos de toma de decisiones.

▶ Como manejar las situaciones

▶ Alternativas ante eventos





Instrumentos para lograr la automatización

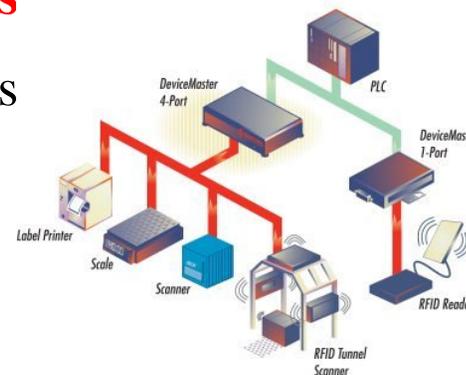
- **Desarrollo de la microelectrónica**

Incorporación de dispositivos a bajo costo en toda la cadena de producción.



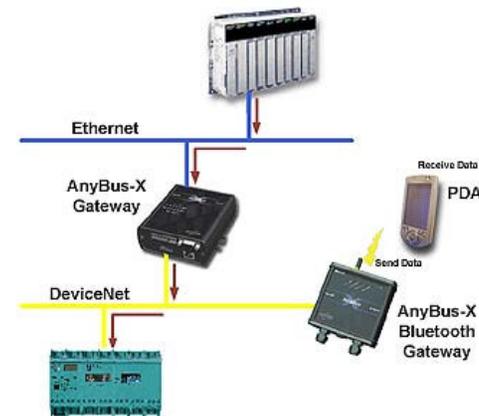
- **Desarrollo de las comunicaciones**

Transferencia de información entre los diferentes puntos de generación y consumo de información.



- **Nuevas técnicas y enfoques**

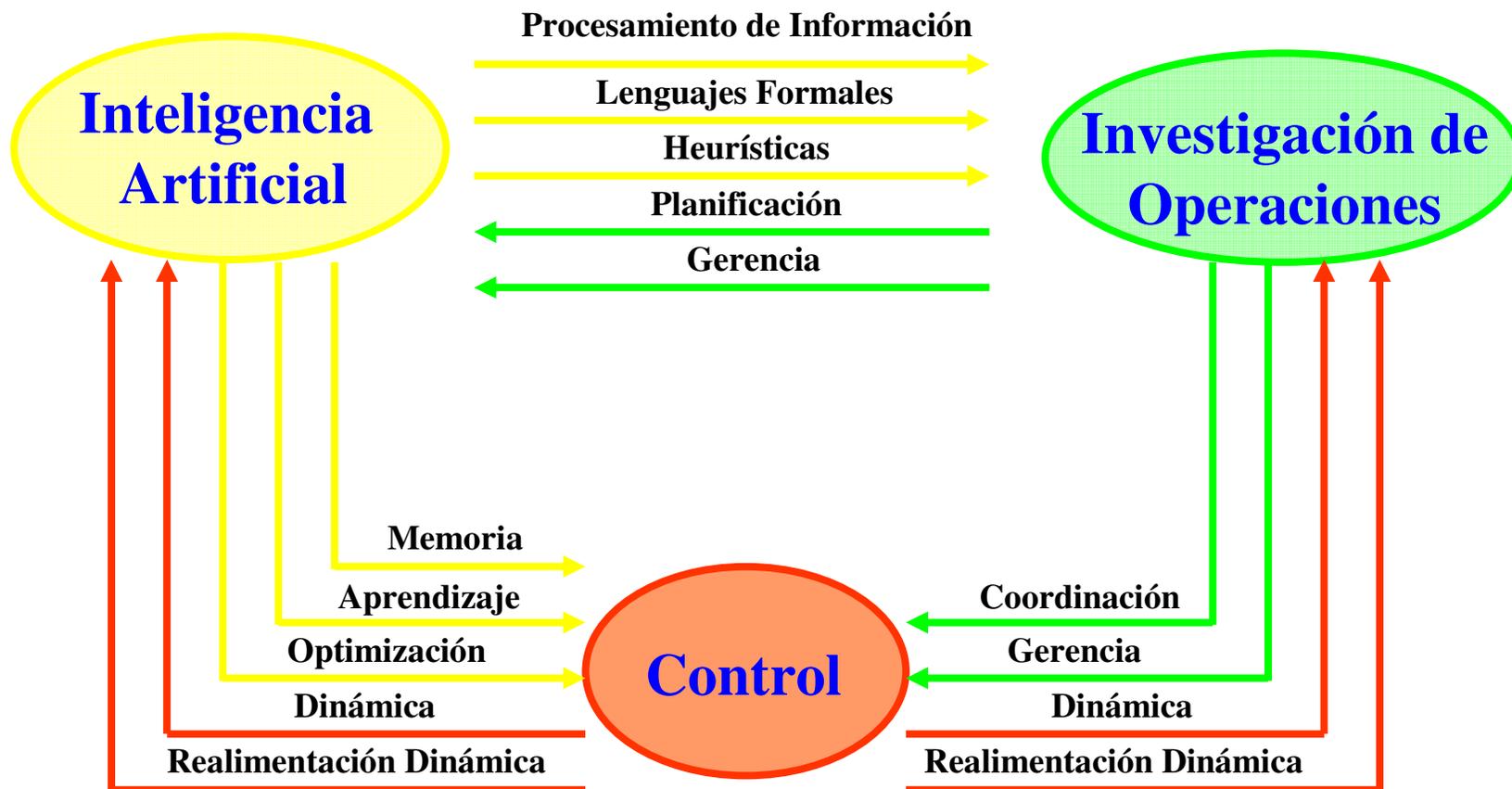
Desarrollo e integración de diferentes áreas del conocimiento en el proceso de modelado y construcción de sistemas de toma de decisiones.





Instrumentos para lograr la automatización

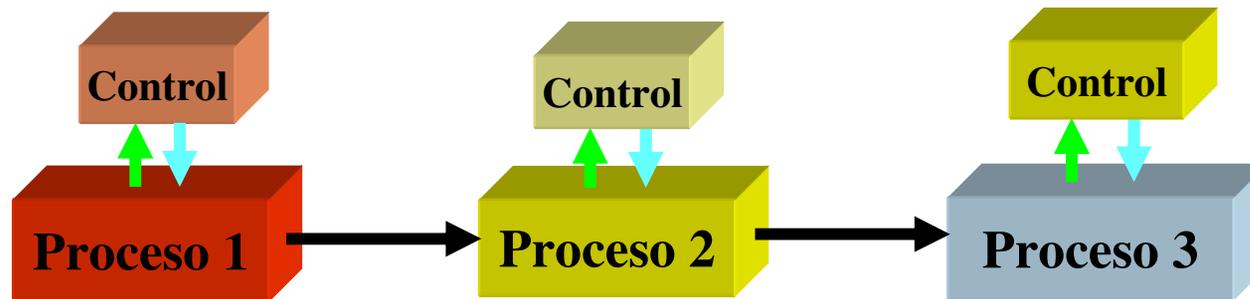
- **Integración de Áreas de Conocimiento en automatización**





Evolución en automatización

● **Etapa 1: Sistemas desconectados**

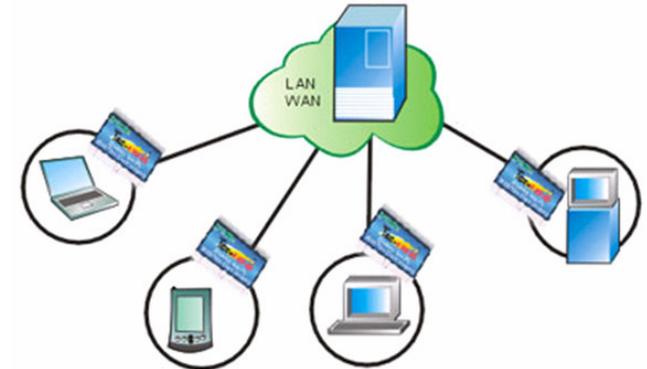
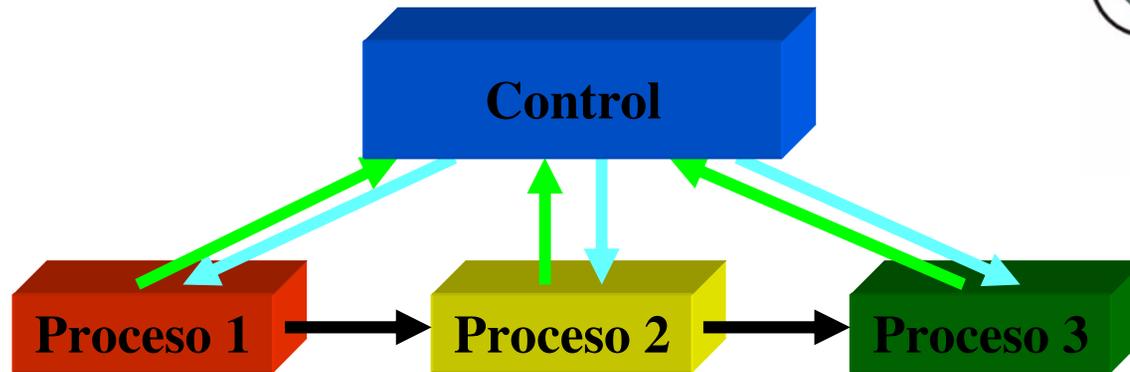


- ▶ El proceso se ve de manera **independiente**. Cada sistema de control se ocupa de una “parte” de la planta.
- ▶ No hay conexión con los otros elementos.
- ▶ Solo importan los aspectos de regulación del proceso.
 - ¿**Costos de producción**? Ignorados.
 - ¿**Disponibilidad del sistema**? Lo resuelve la gente de planificación.
- ▶ La tecnología es propietaria. Por ende cerrada.
- ▶ La coordinación entre los procesos productivos es manejada de manera manual. Se fijan a priori las consignas.



Evolución en automatización

- **Etapa 2: Sistemas Centralizados**

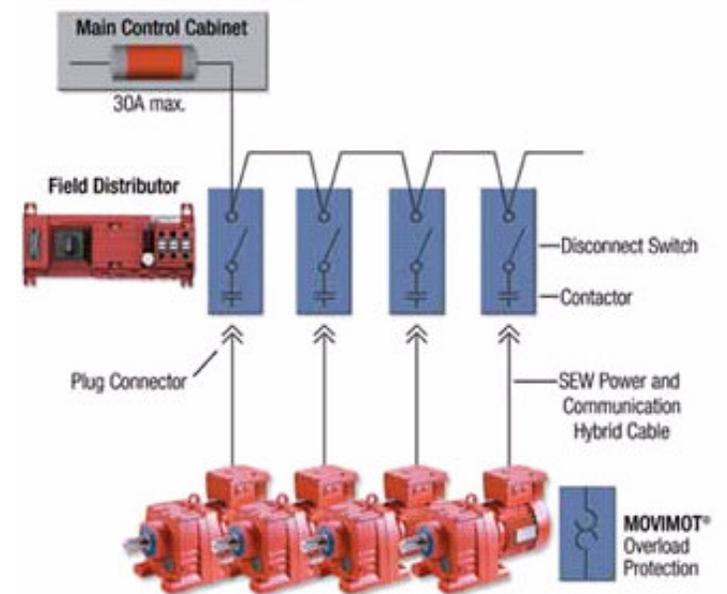
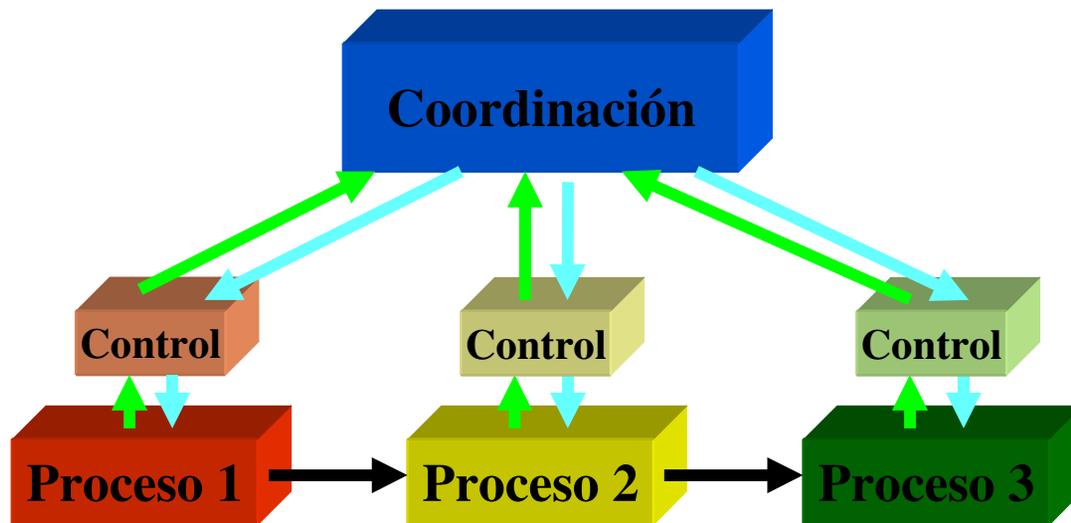


- ▶ Introducción de los aspectos de coordinación.
- ▶ Mayor reactividad.
- ▶ Se evitan incoherencias entre formas de operación.
- ▶ Sistemas de apoyo al operador. (Sistemas Expertos en Línea).
- ▶ Interfaces orientadas al usuario.
- ▶ Especialización de los equipos.



Evolución en automatización

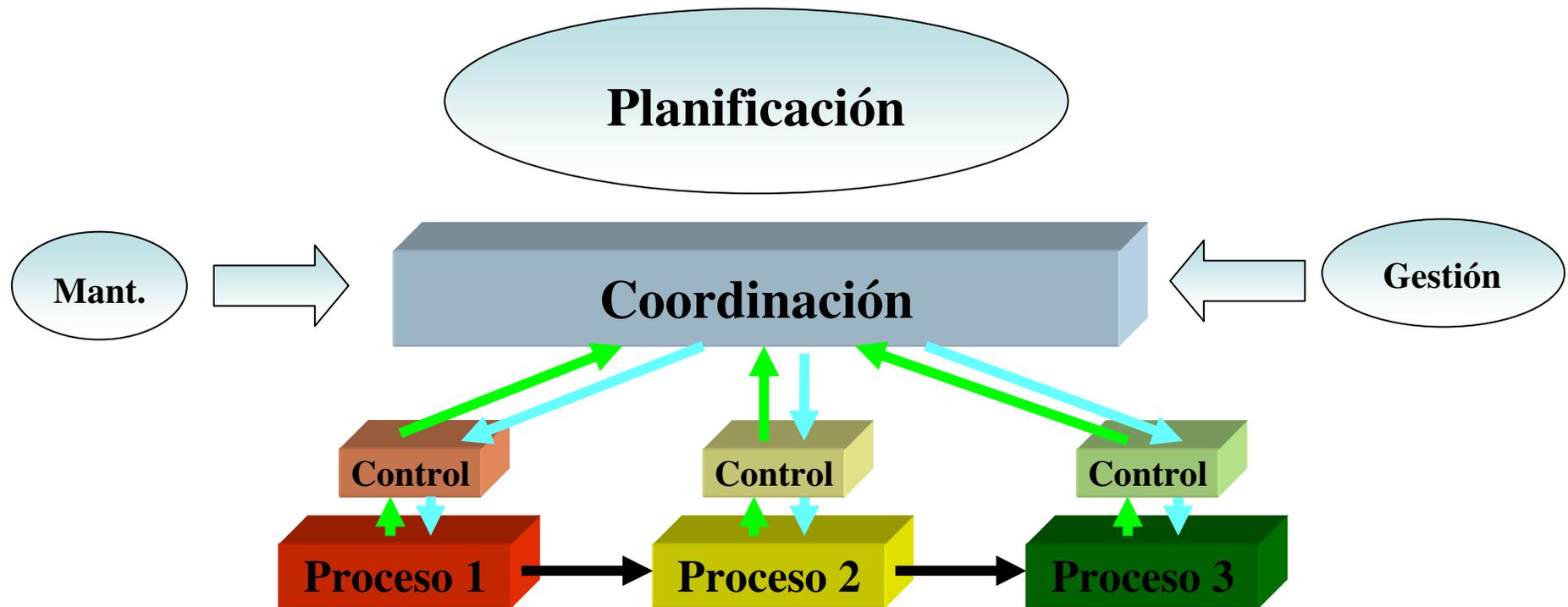
● Etapa 3: Sistemas Descentralizados





Evolución en automatización

● **Etapa 4: Descentralizados jerárquicos (Situación actual)**



▶ Coordinación, inteligencia en campo, optimización centralizada.

▶ Modelos de Integración. PERA, SP95.

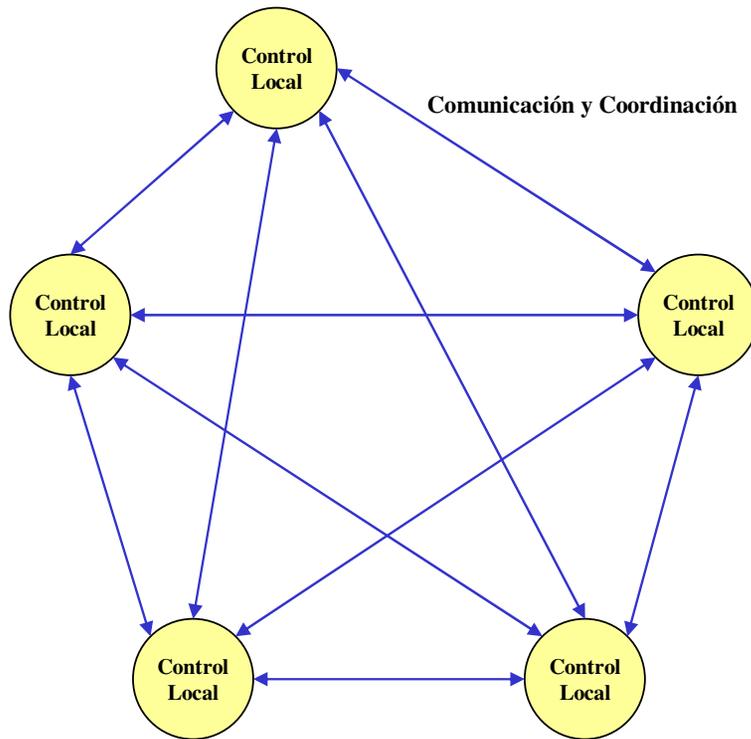
▶ Descripción utilizando XML.

▶ Uso de orientación a objetos.

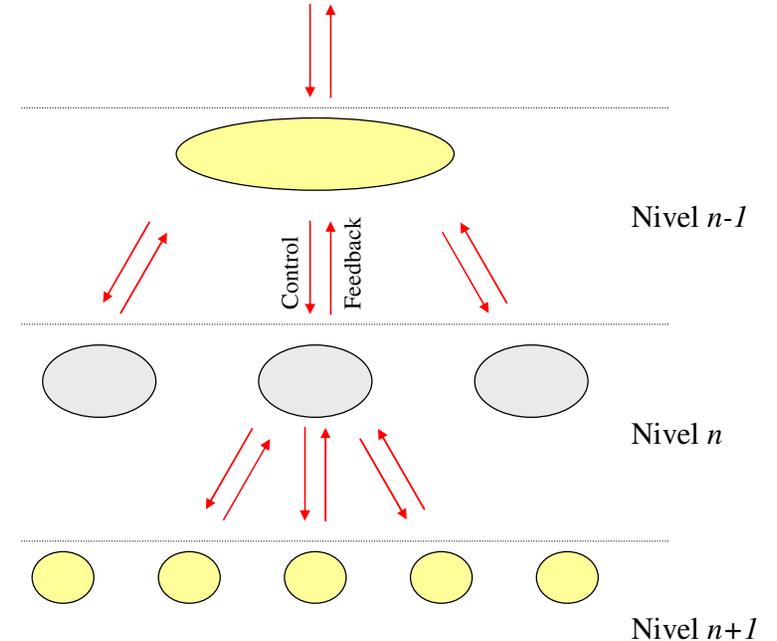


Evolución en automatización

● Etapa 5: Heterarchical (Tendencias)



Sistema Heterarchical



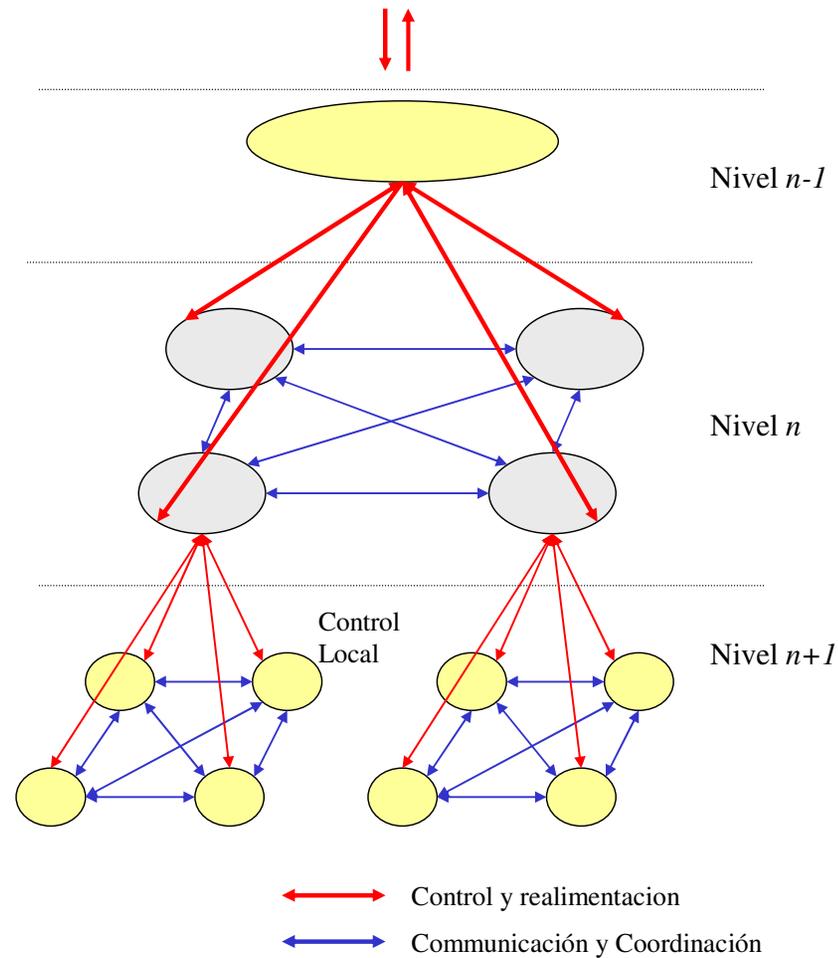
Sistema Jerárquico



Evolución en automatización

● Etapa 5: Sistema Holónico (Tendencias)

Sistema Holónico

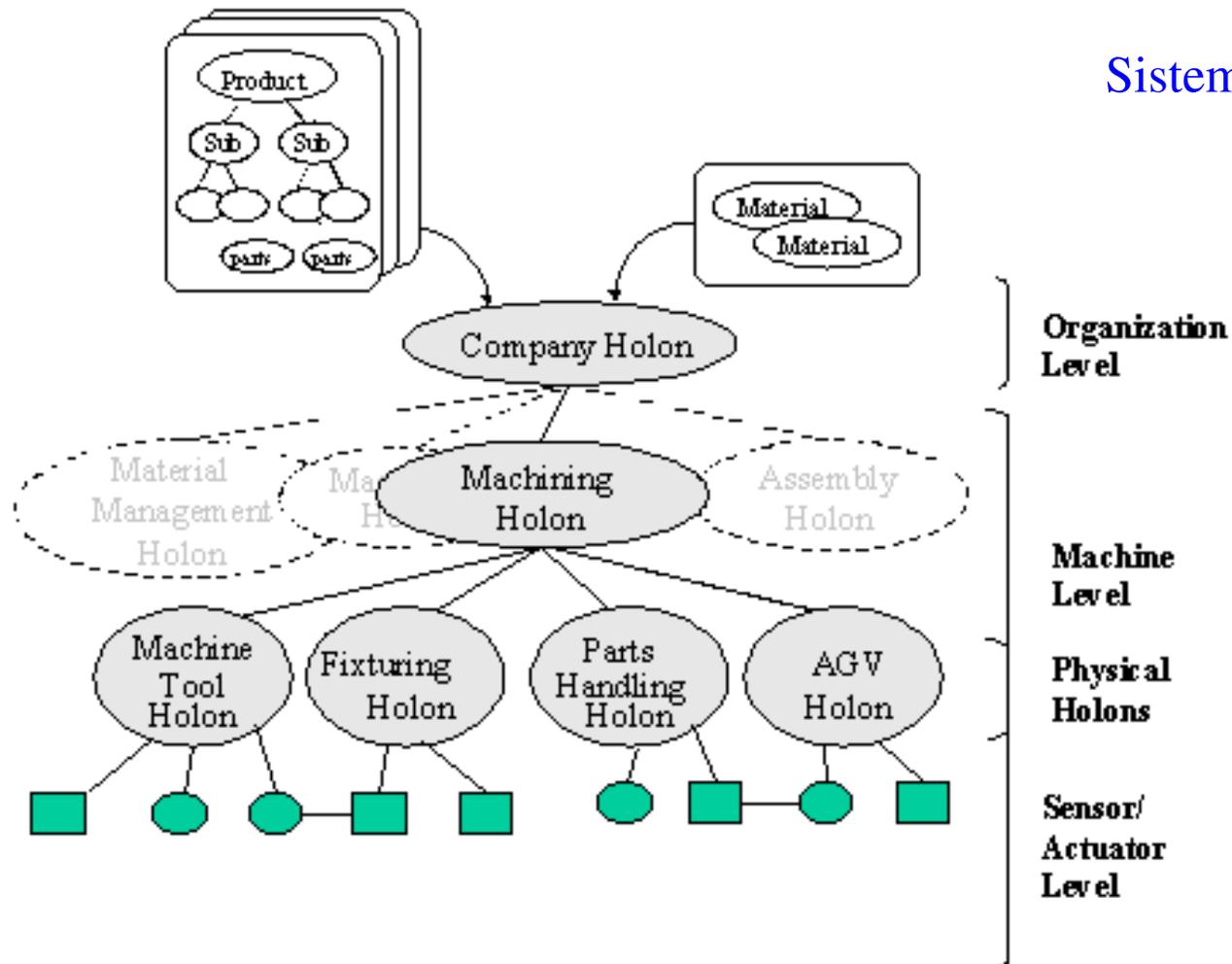




Evolución en automatización

- **Etapa 5: Sistema Holónico (Tendencias)**

Sistema Holónico





Sistemas de Manufactura

- **Definiciones Básicas**

- ▶ **Manufactura (Manufacturing)**

Es un conjunto correlacionado de **actividades y operaciones**, las cuales incluyen **diseño de productos, selección de material, planificación, producción, inspección, gerencia y mercadeo de productos**; para la industria de la manufactura

- ▶ **Manufactura producción (Manufacturing Production)**

Es una **serie de procesos** adoptados para **fabricar un producto**, procesos semejantes excluyendo las actividades de diseño, planificación y control de la producción.





Sistemas de Manufactura

● Definiciones Básicas

▶ Procesos de manufactura (Manufacturing Processes)

Son las **actividades de más bajo nivel** usadas para hacer o construir productos. Máquinas tradicionales como: Taladradoras, tornos, etc. Y avanzadas como: Electroquímicas, electrodescargas.





Sistemas de Manufactura

- **Definiciones Básicas**

- ▶ **Ingeniería Manufactura (Manufacturing Engineering)**

Involucra el **diseño operación y control de los procesos** de manufactura. Es el corazón del diseño, planificación y control de los sistemas de manufactura, que requiere el conocimiento de otras ingenierías tales como: **eléctrica, mecánica, química, materiales y sistemas, e informática.**

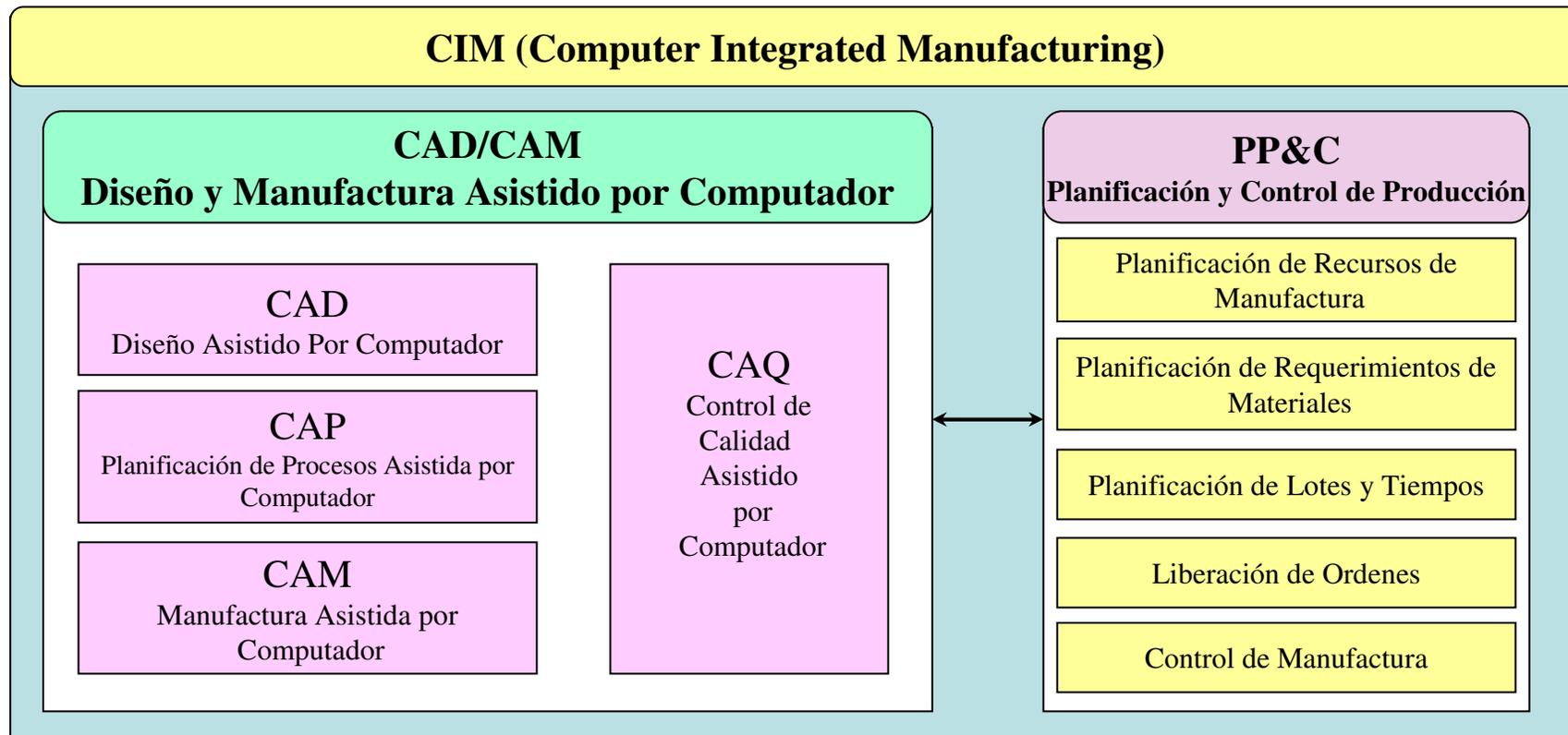
- ▶ **Sistemas de Manufactura (Manufacturing System)**

Es una organización que comprende **diferentes interrelacionados conjuntos de manufactura.** Su objetivo es la interfase a las funciones de producción con actividades fuera de ella, con el objeto de realizar la optimización del rendimiento total del sistema de producción. Las actividades de este subconjunto incluyen **diseño, planificación, manufactura y control.** Estos subconjuntos están también conectados con actividades como: **cuentas, mercado, financiamiento y personal.**



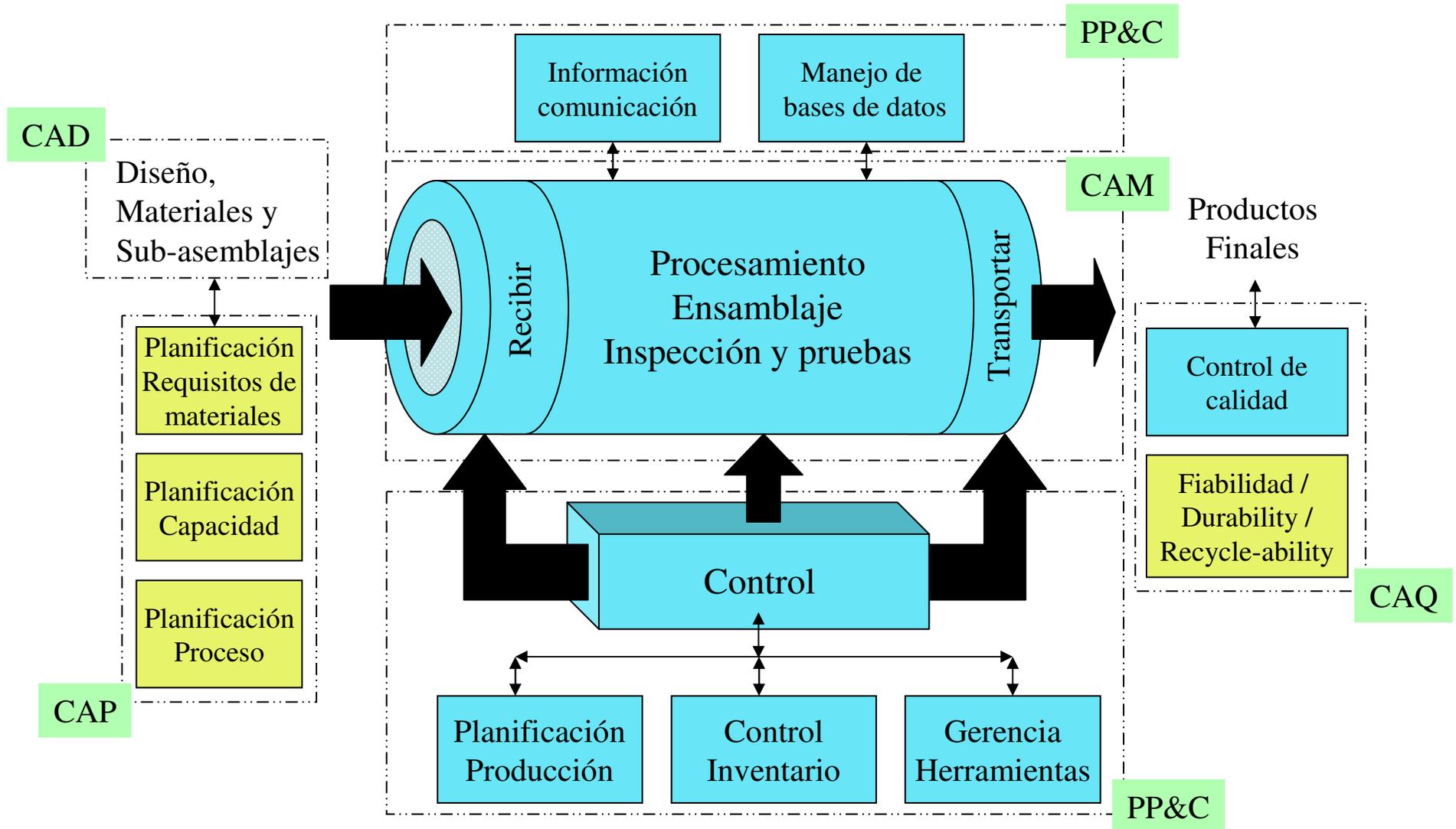
Computer Integrated Manufacturing (CIM)

CIM es la integración del negocio, ingeniería, manufactura y administración de la información, de las funciones de una compañía, desde el mercadeo hasta la distribución de sus productos.





Computer Integrated Manufacturing (CIM)



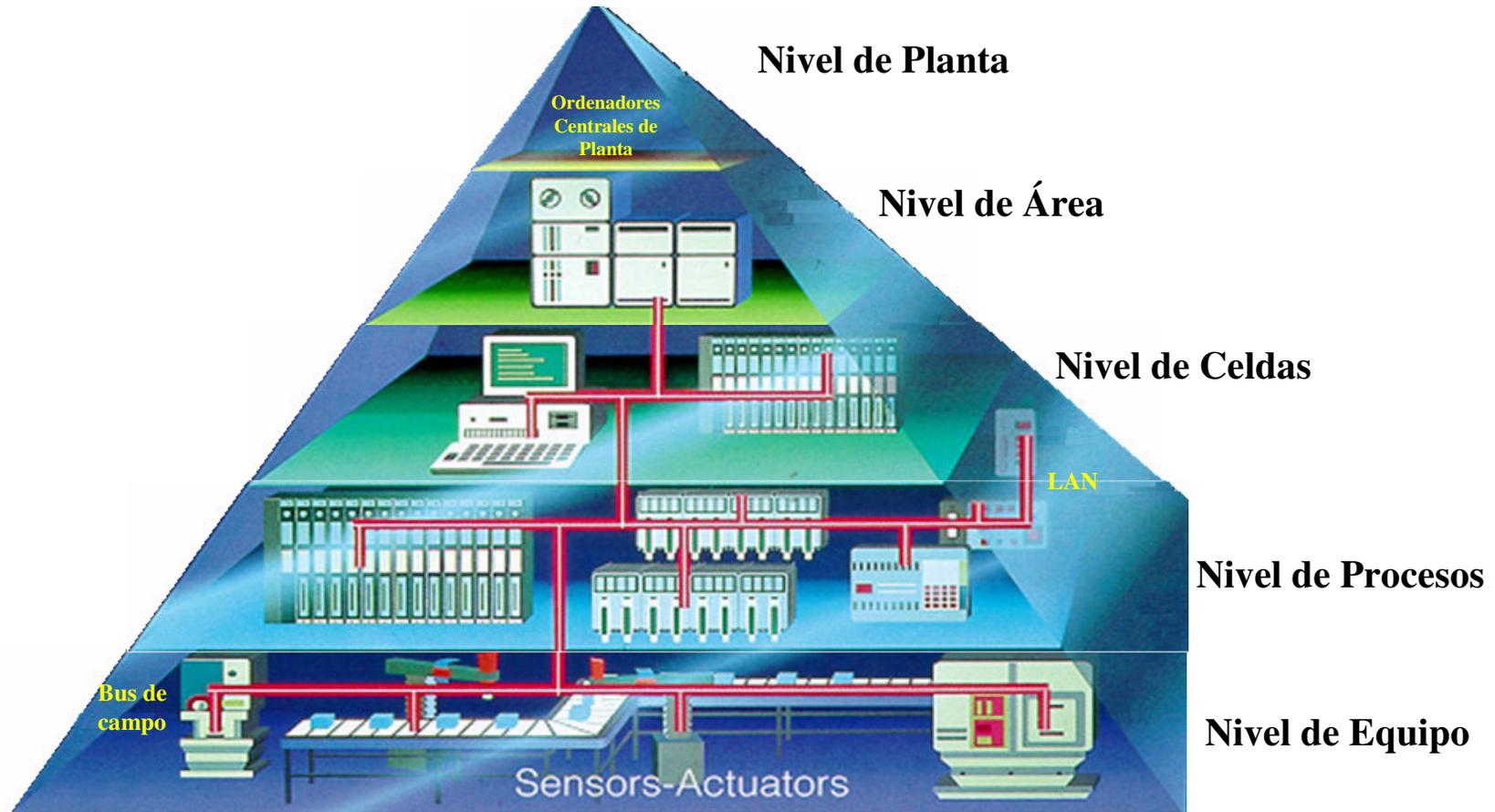


Computer Integrated Manufacturing (CIM)



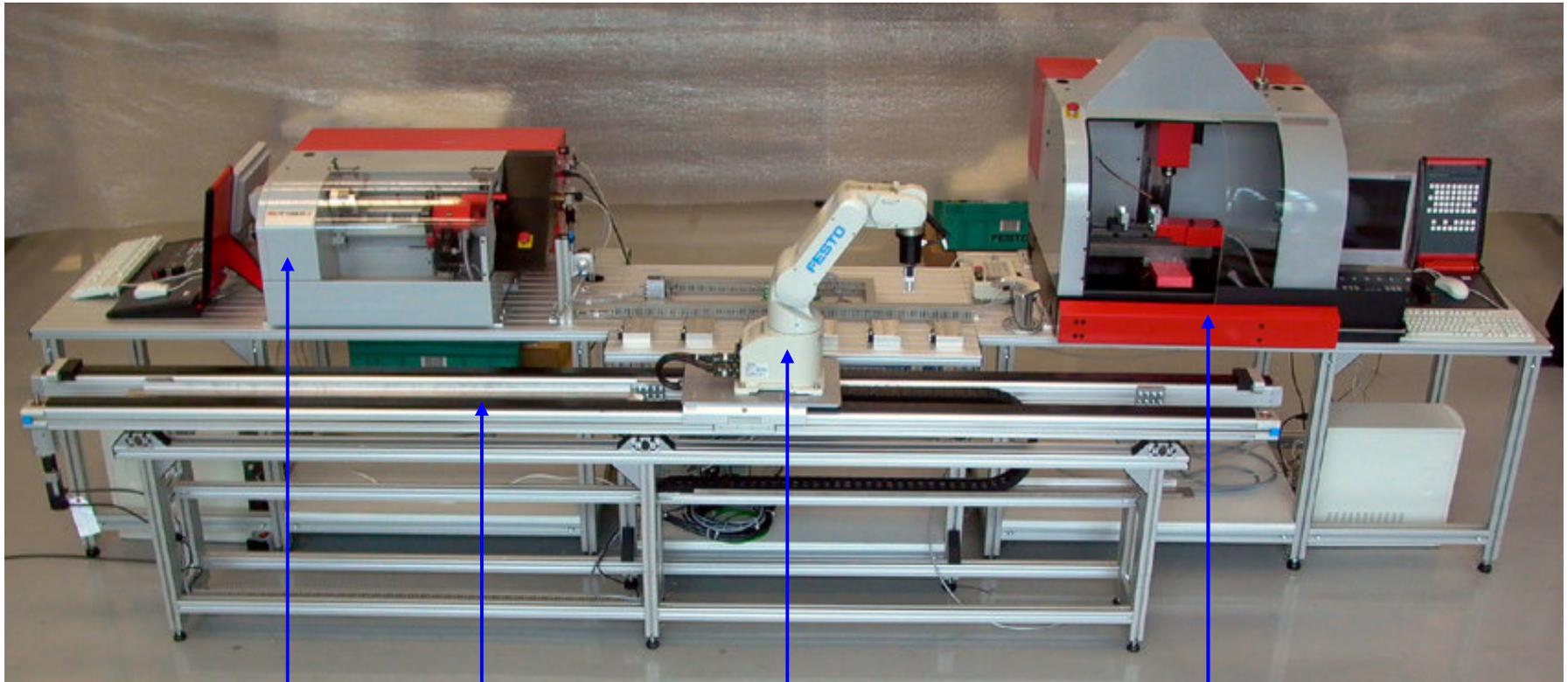


Componentes básicos de un sistema CIM





Nivel de equipo



Torno CNC

Transportador

Manipulador

Fresadoras CNC

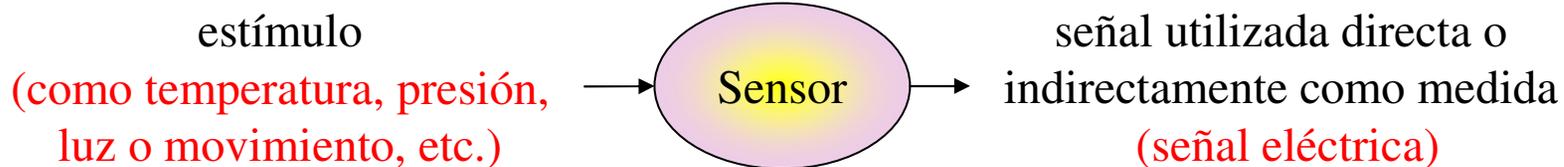
(EMCO/FESTO)



Nivel de equipo

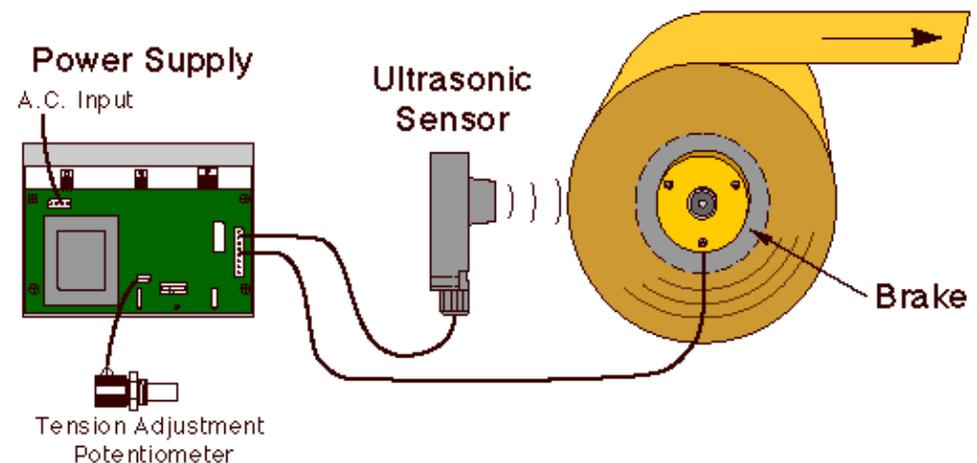
● Sensores

Un sensor es un dispositivo que detecta, o sensa manifestaciones de cualidades o fenómenos físicos, como la energía, velocidad, aceleración, tamaño, cantidad, etc.



Tipos:

- ▶ Sensores térmicos
- ▶ Sensores electromagnéticos
- ▶ Sensores mecánicos
- ▶ Sensores químicos
- ▶ Sensores ópticos
- ▶ Sensores ópticos y de radiación
- ▶ Sensores acústicos.





Nivel de equipo

● Actuadores

Actuación puede ser:

- **Acción física directa** sobre un proceso como parar una cinta;
- **Acción indirecta** como activar una señal eléctrica que tiene un efecto directo sobre un proceso. Un releé es un actuador que puede encender un horno a través de activar el circuito eléctrico del horno.

Tipos:

- ▶ Cilindros neumáticos o hidráulicos ▶ Solenoides ▶ Relees

● Accionadores

Como los actuadores, los accionadores actúan sobre un proceso.

- Los accionadores ejecutan **mas movimientos continuos** como la rotación;

Ejemplos: ▶ Accionadores eléctricos



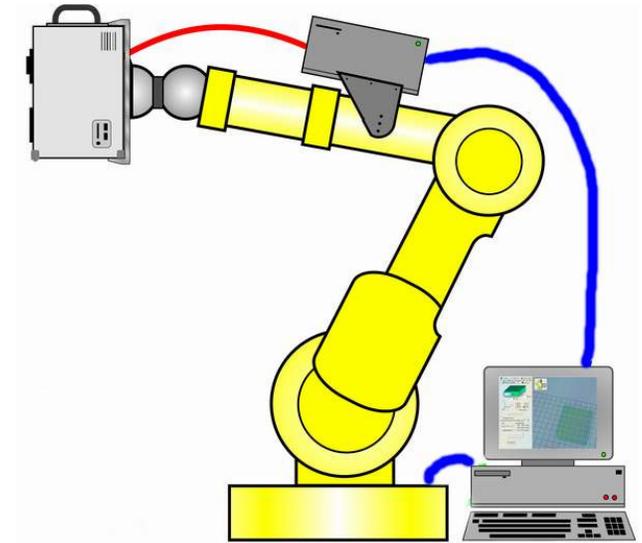


Nivel de equipo

Robots Industriales

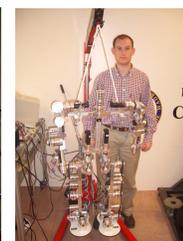
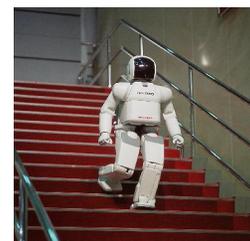
Un **robot industrial**, es un manipulador multi-funcional re-programable, capaz de mover cargas, piezas, herramientas o dispositivos especiales, según trayectorias varias, programadas para la ejecución de tareas diversas.

Robot Industry Association (RIA)



Otros tipos:

- ▶ Robots escaladores
- ▶ Robots Móviles Autónomos
- ▶ Humanoides
- ▶ Robots de Servicios
- ▶ Robots de espacio
- ▶ Nanorobots





Nivel de equipo

- **CNC - Computerized Numerically Controlled**

En una máquina CNC (**Control Numérico Computarizado**), un computador 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.

Aplicaciones:

Fresadoras, Torno, taladradoras, etc...

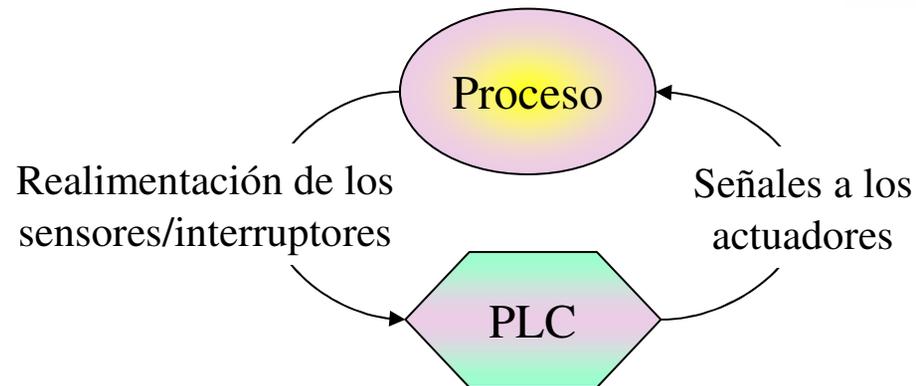




Control de procesos

● PLC – Programmable Logic Controller

Un autómata programable industrial (PLC) es un equipo electrónico de control con un cableado interno independiente del proceso a controlar, que se adapta a dicho proceso mediante un programa específico que contiene la secuencia de operaciones a realizar.



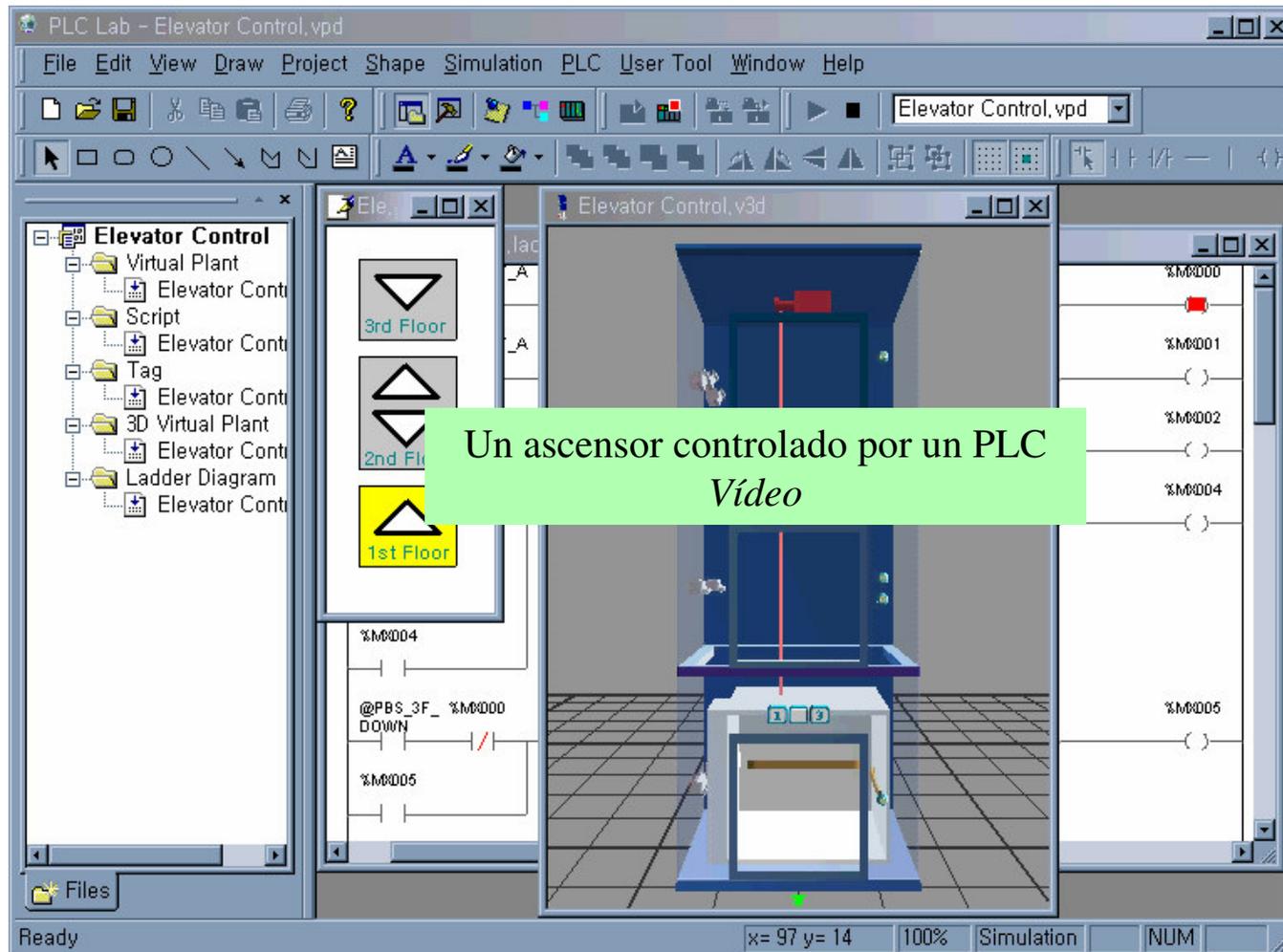
Programación de un PLC:

- ▶ Diagrama de contactos (Ladder logic programming);
- ▶ Lista de instrucciones (IL);
- ▶ Lenguaje literal estructurado (Structured Text).



Control de procesos

- **PLC – Programmable Logic Controller**



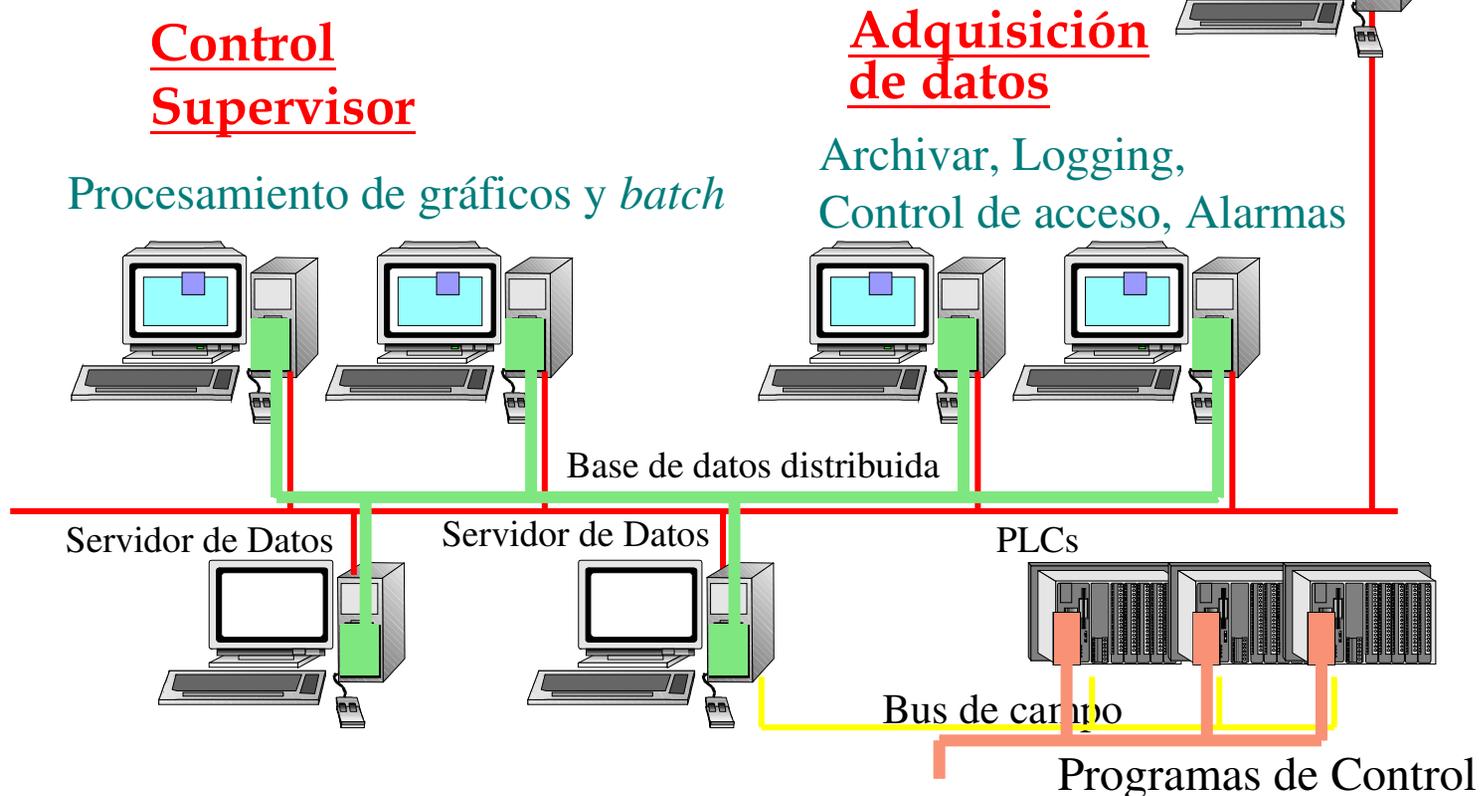


Monitorización y Control Supervisor

● SCADA– Supervisory Control And Data Acquisition

SCADA (**S**upervisión, **C**ontrol y **A**dquisición de **D**atos) es un sistema basado en computadores que permite supervisar y controlar a distancia una instalación de cualquier tipo.

Sistemas ERP
Sistemas Expertos





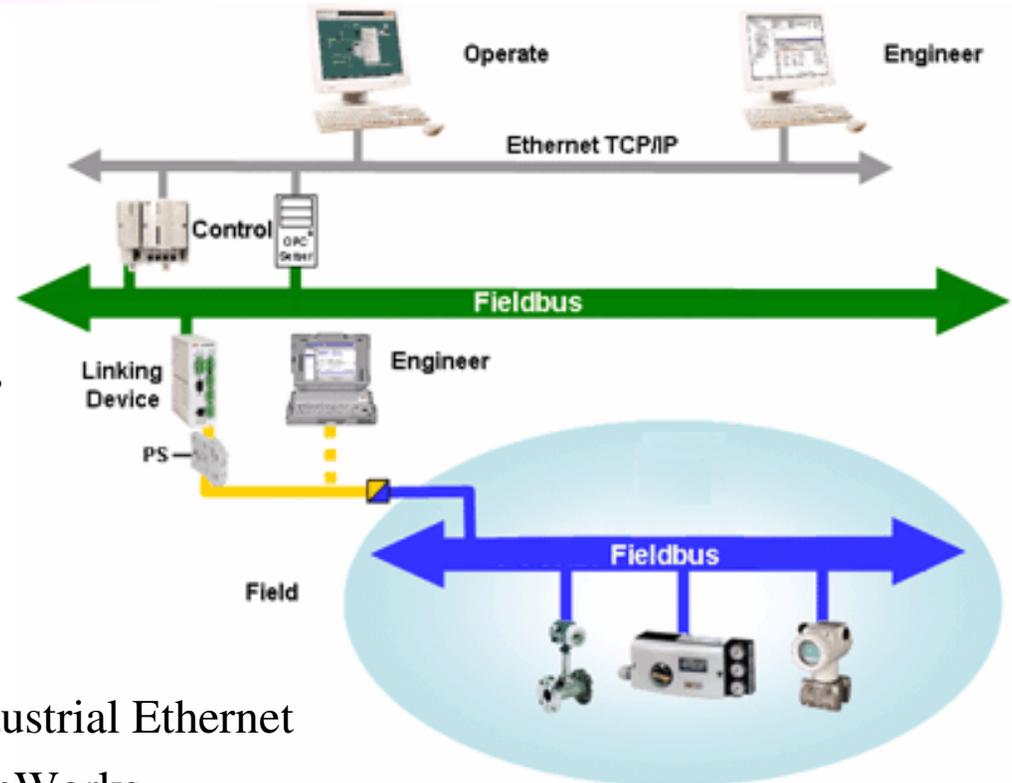
Esquemas de Comunicación

● Buses de campo (Fieldbus)

Un bus de campo es un sistema de transmisión de información (datos) que simplifica enormemente la instalación y operación de máquinas y equipamientos industriales utilizados en procesos de producción.

Estándares:

- ▶ AS-Interface
- ▶ CAN
- ▶ DeviceNet
- ▶ FOUNDATION fieldbus
- ▶ HART Protocol
- ▶ Industrial Ethernet
- ▶ LonWorks
- ▶ modbus
- ▶ Netbus
- ▶ Profibus (Process Field Bus) es el más popular.





Resumen:

- Los objetivos de automatizar un sistema son:
 - ▶ Mejorar la cadena productiva
 - ▶ Manejar la información del mercado
 - ▶ Responder eficientemente ante cambios
 - ▶ Seguridad de las operaciones
 - ▶ Comodidad de la operación
 - ▶ Conocer mejor el proceso
 - Los elementos básicos de un sistema **CIM** son:
 - ▶ Sensores/Actuadores/Accionamientos/Robots/Maquinas CNC
 - ▶ PLCs
 - ▶ SCADA
 - ▶ Buses
 - Las desventajas de automatización son:
 - ▶ Coste inicial elevado
 - ▶ Los sistemas automatizados requieren mantenimiento de manera continua.
 - Ventajas/desventajas ¿?
 - ▶ Impacto sobre el empleo
-



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 1 – Miércoles 19 de Abril 2006

Introducción a la Automatización



Automatización Industrial - II

56 – 10569 2° Cuatrimestre 2006

Clase 2 – Jueves 20 de Abril 2006





Qué es un robot?

- **Oxford English Dictionary**

Un robot es un aparato mecánico que se parece y hace el trabajo de un ser humano.

- **Robot Institute of America (RIA)**

Un robot industrial, es un manipulador **multi-funcional re-programable**, capaz de mover cargas, piezas, herramientas o dispositivos especiales, según trayectorias varias, programadas para la ejecución de tareas diversas.

- **International Federation of Robotics (IFR)**

Un robot es un manipulador **automáticamente controlado, reprogramable** en tres o más ejes, capaz de ejecutar varias tareas.

- **Japanese Industrial Robot Association (JIRA)**

A dispositivo con grados de libertad que pueden ser controlados.





Qué es un robot?

- **Asociación Francesa de Normalización (AFNOR)**

- ▶ **Manipulador:** mecanismo formado generalmente por elementos en serie, articulados entre si, destinado al **agarre y desplazamiento de objetos**. Es **multifuncional** y puede ser **gobernado** directamente por un operador humano o mediante un dispositivo lógico.



- ▶ **Robot:** manipulador automático **servocontrolado, reprogramable, polivalente, capaz de posicionar y orientar piezas**, útiles o dispositivos especiales, siguiendo trayectorias variables reprogramables, para la ejecución de tareas variadas”.



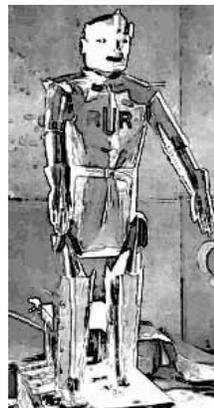
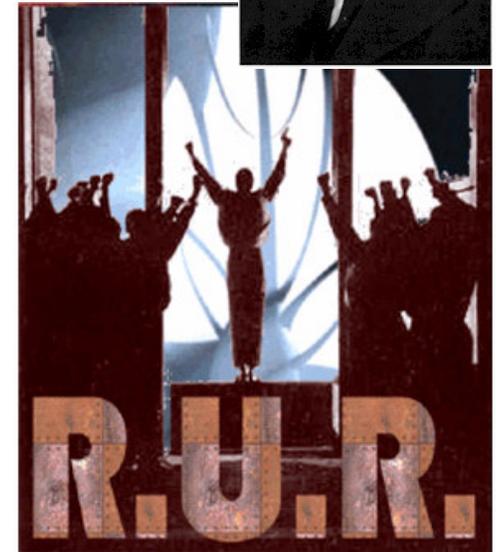
Automatización Industrial -II

Ciencia Ficción

- **Rossum's Universal Robot (R.U.R) - Karel Capek**

La palabra Robot fue utilizada por primera vez en el año 1921, cuando el escritor Karel Capek (1890–1938), estrena en el teatro nacional de Praga su obra : **“Rossum's Universal Robot (R.U.R)”**. Los robots eran máquinas androides.

Su origen es la palabra eslava **Robota**, que se refiere a trabajo realizado de manera forzada.

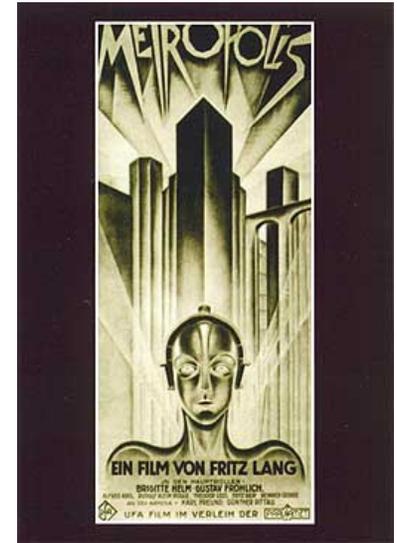




Automatización Industrial -II

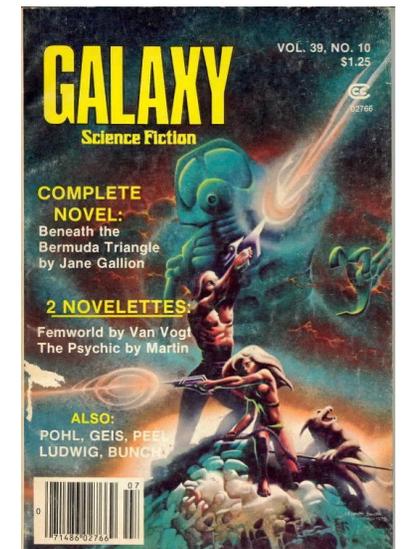
Ciencia Ficción

Así, en 1926, [Thea von Harbou](#) escribe “*Metrópolis*”, novela posteriormente llevada al cine por su marido Fritz Lang, en donde la masa obrera de una sociedad super-industrializada es manipulada por un líder androide llamado María.



Sin duda alguna , fue el escritor americano de origen ruso [Isaac Asimov](#) (1920 - 1992) el máximo impulsor de la palabra robot.

En octubre de 1945, Asimov publica en la revista *Galaxy Science Fiction*, una historia en la que por primera vez enunció sus tres leyes de la robótica.





Ciencia Ficción

- **Leyes de la Robótica - Isaac Asimov**

Primera Ley:

Un robot no puede perjudicar a un ser humano, ni con su inacción permitir que un ser humano sufra daño.

Segunda Ley:

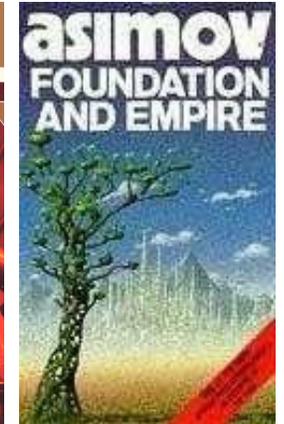
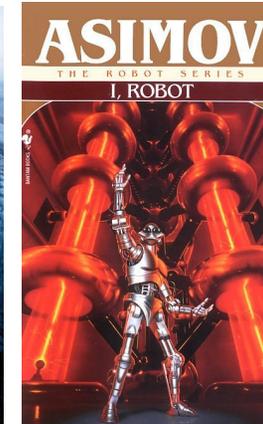
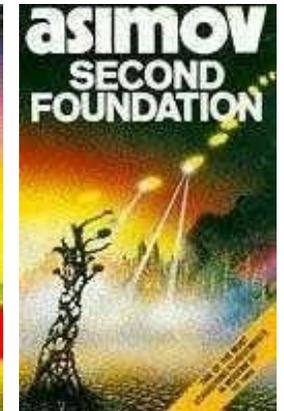
Un robot ha de obedecer las ordenes recibidas de un ser humano, excepto si tales ordenes entran en conflicto con la primera ley.

Tercera Ley:

Un robot debe proteger su propia existencia mientras tal protección no entre en conflicto con la primera o segunda ley.

Ley Cero:

Un robot no puede lastimar a la humanidad o, por falta de acción, permitir que la humanidad sufra daños.

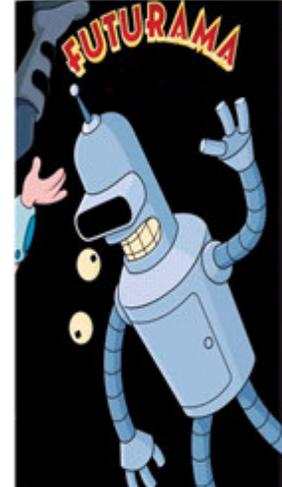
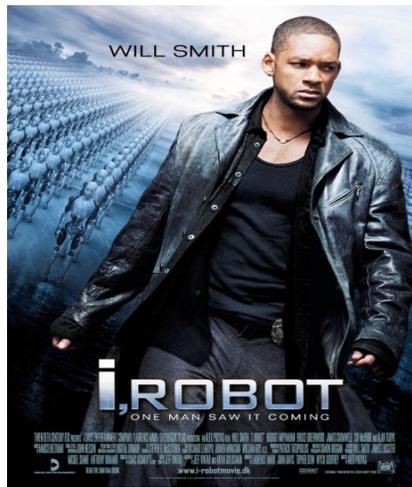
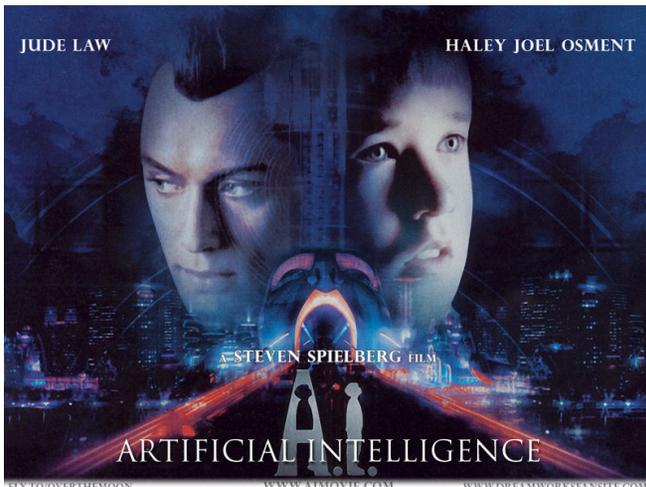
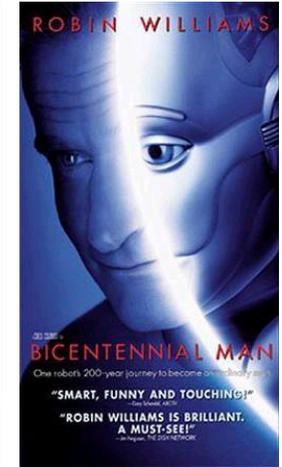
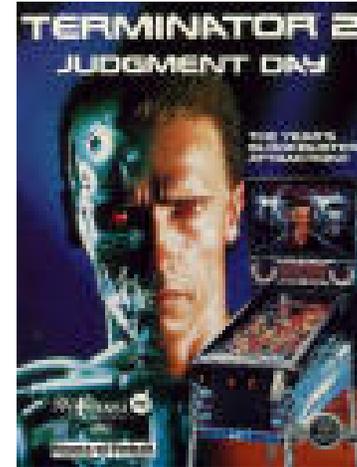
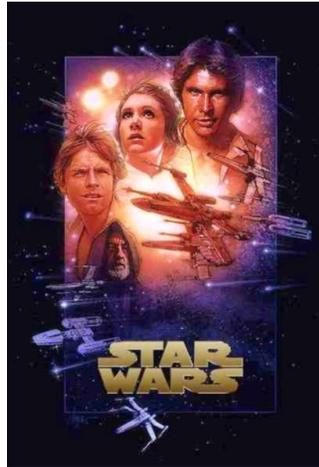




Automatización Industrial -II

Ciencia Ficción

- Clásicos del cine y nuevas películas





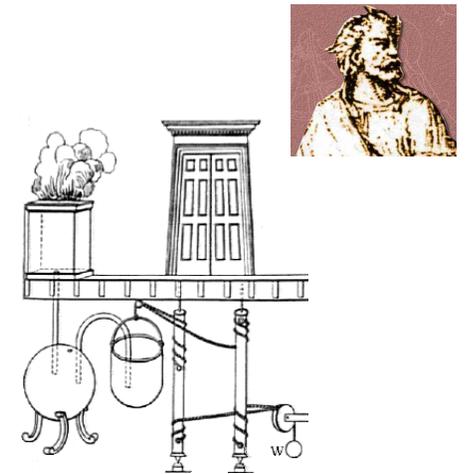
Antecedentes

● **Autómatas mecánicos**

- ▶ **Automatos** : Palabra griega específica para denominar máquinas y dispositivos capaces de imitar las funciones y los movimientos de los seres vivos.
- ▶ **Autómata** : Deriva de la palabra griega “máquina que imita la figura y movimientos de de un ser animado”.

85 d.c - Mecanismos animados de **Herón de Alejandría** se movían por intermedio de dispositivos hidráulicos, poleas y palancas y tenían fines eminentemente lúdicos.

1200 - Bebederos y lavaderos automáticos.





Antecedentes

- ▶ El reloj-efante: descrito en, *El libro del conocimiento y de los dispositivos mecánicos ingeniosos*, por Al-Jazari (1206 AD).



- ▶ Gallo de la catedral de Estrasburgo 1352.

Este es el autómata más antiguo que se conserva en la actualidad.

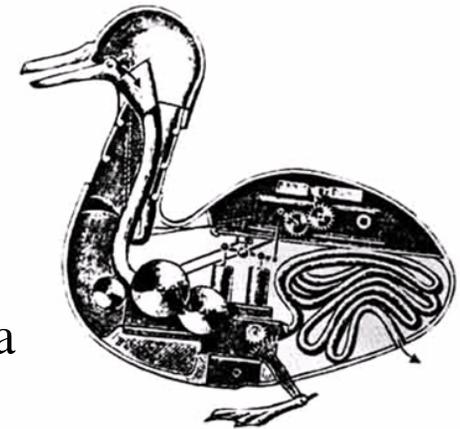




Antecedentes

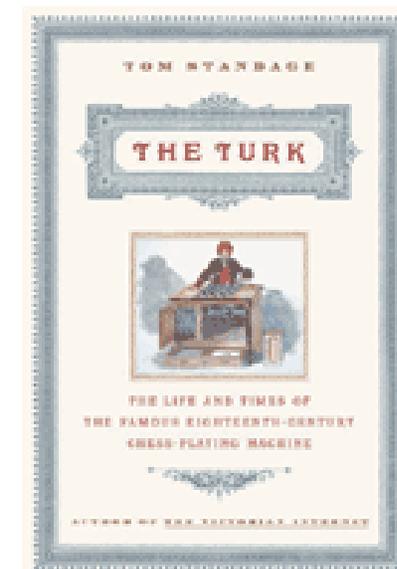
▶ Pato Vaucanson 1738

Según **Sir David Brewster** en un escrito de 1868, describe este pato diciendo que es "la pieza mecánica más maravillosa que se haya hecho". El pato alargaba su cuello para tomar el grano de la mano y luego lo tragaba y lo digería. Podía beber, chapotear y graznar, y también imitaba los gestos que hace un pato cuando traga con precipitación.



▶ El Turco de Von Kempelen - 1769

Un hombre mecánico, vestido con un atuendo oriental, sentado detrás de un gabinete de madera y capaz de jugar al Ajedrez.





Antecedentes

► Muñeca mecánica Familia Droz - 1770



Video



Video



Video

► Muñeca mecánica Maillardet - 1805





Antecedentes

● Historia de los Robots

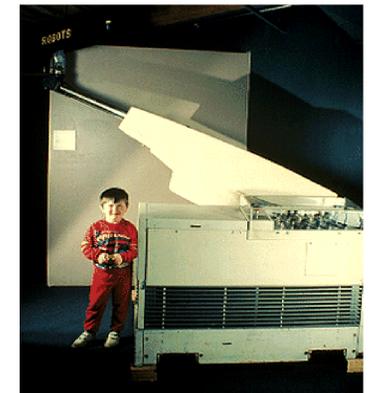
- ▶ 1928: El primer automata con un motor.
- ▶ 1948: **R. C. Goertz** del **Argonne National Laboratory** desarrolló el primer manipulador. El objetivo era **manipular elementos radioactivos** sin riesgo para el operador. Consistía en un elemento mecánico maestro-esclavo.
- ▶ 1954: Primer telemanipulador servocontrolado por el inventor británico **C. W. Kenward**.
- ▶ 1956: **G. Devol** y **J. Engelberger** crean **Unimation**.
- ▶ 1960: Primer robot industrial instalado en la factoría de **General Motors**.
- ▶ 1973: La firma sueca **ASEA** desarrolla el primer robot totalmente eléctrico.



George Devol



Joseph F. Engelberger



Unimate



Antecedentes

● Historia de los Robots

- ▶ 1978: **PUMA** (Programmable Universal Machine for Assembly) creado por Unimation y General Motors.
- ▶ 1982: Makino de Japón crea el concepto de robot **SCARA** (Selective Compliance Assembly Robot Arm).





Antecedentes

● Asociaciones

▶ 1972: JIRA (Japan Industrial Robots Assoc.)

▶ 1980: IFR (International Federation of Robots)



▶ 1984: RIA (Robotics Industries Assoc.) de Estados Unidos

▶ 1984: AFRI (Asociación Francesa de Robótica Industrial)

▶ 1985: AER (Asociación Española de Robótica)



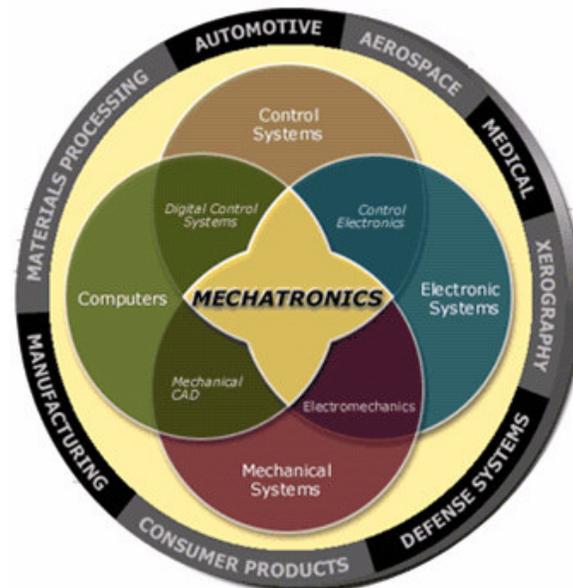


Definiciones básicas

● Robótica

La robótica es una ciencia o rama de la tecnología, que estudia el diseño y construcción de máquinas capaces de desempeñar **tareas realizadas por el ser humano o que requieren del uso de inteligencia.**

Campo tecnológico: Intersección de varias disciplinas.





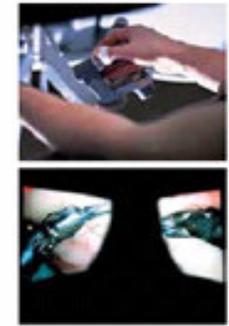
Definiciones básicas

- **Autonomía**

Los robots son capaces de **tomar decisiones inteligentes** en función de la su estado, la información que percibe del entorno y la misión que tiene asignada.

- **Tele-operación**

Dispositivos robóticos con brazos manipuladores, sensores y cierto grado de movilidad, controlados **remotamente** por un operador humano de manera directa o a través de un ordenador.



La primera operación transatlántica de cirugía, 2001
(Nueva York, EE.UU. – Strasbourg, Francia)

- **Tele-Presencia**

Existe una realimentación mecánica o por intermedio de un ordenador que permite al operador **percibir las fuerzas** (escaladas) realizadas por el esclavo.



Clasificación de los robots

● **Generaciones**

I Generación	<ul style="list-style-type: none">• Repite la tarea programada secuencialmente.• No toma en cuenta las posibles alteraciones en su entorno.
II Generación	<ul style="list-style-type: none">• Adquiere información limitada de su entorno y actúa en consecuencia.• Puede localizar, clasificar (visión), detectar esfuerzos y adaptar sus movimientos a ellos, etc.
III Generación	<ul style="list-style-type: none">• Su programación se realiza mediante el empleo de un lenguaje natural.• Posee capacidad para la planificación automática de tareas.



Clasificación de los robots

● **Asociación Francesa de Robótica Industrial AFRI**

Tipo A	<ul style="list-style-type: none">• Manipulador con control manual o telemando.
Tipo B	<ul style="list-style-type: none">• Manipulador automático con ciclos preajustados.• Regulación mediante fines de carrera o topes.• Control por PLC.• Accionamiento neumático, eléctrico o hidráulica.
Tipo C	<ul style="list-style-type: none">• Robot programable con trayectoria continua o punto a punto.• Carece de conocimiento de su entorno.
Tipo D	<ul style="list-style-type: none">• Robot capaz de adquirir datos de su entorno readaptando su tarea en función de estos.



Clasificación de los robots

- T. M. Knasel - Revista Robotics N 2. 1986**

Generación	Nombre	Control	Movilidad	Usos
I. 1982	Pick & Place	Fines de carrera. Aprendizaje	Ninguna	Manipulación servicio de máquinas
II. 1984	Servo	Servocontrol trayectoria continua. Programas con condiciones.	Una vía	Soldadura, Pintura
III. 1989	Ensamblado	Servos de precisión. Visión, Tacto. Programación off-line.	AGV	Ensamblaje, desbarbado.
IV. 2000	Móvil	Sensores inteligentes	Piernas - Ruedas	Construcción, Mantenimiento.
VI. 2010	-----	Inteligencia artificial IA	Andante - Saltarin	Militar – Espacial.



Clasificación de los robots

Aplicaciones	<ul style="list-style-type: none">– Robots Industriales.<ul style="list-style-type: none">• Control manual• Control automático• Programables• Adaptable al entorno– Robots de Servicio.
Funcionamiento	<ul style="list-style-type: none">– Automáticos.– Teleoperados.
Estructura	<ul style="list-style-type: none">– Manipuladores.– Móviles– Manipuladores Móviles
Inteligencia	<ul style="list-style-type: none">– Programables– Autónomos.



Tipos de robots

- Manipuladores (Robots Industriales)





Tipos de robots

- **Manipulador Móvil**

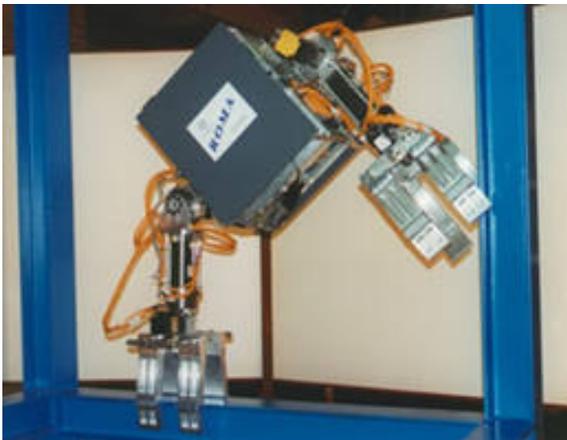
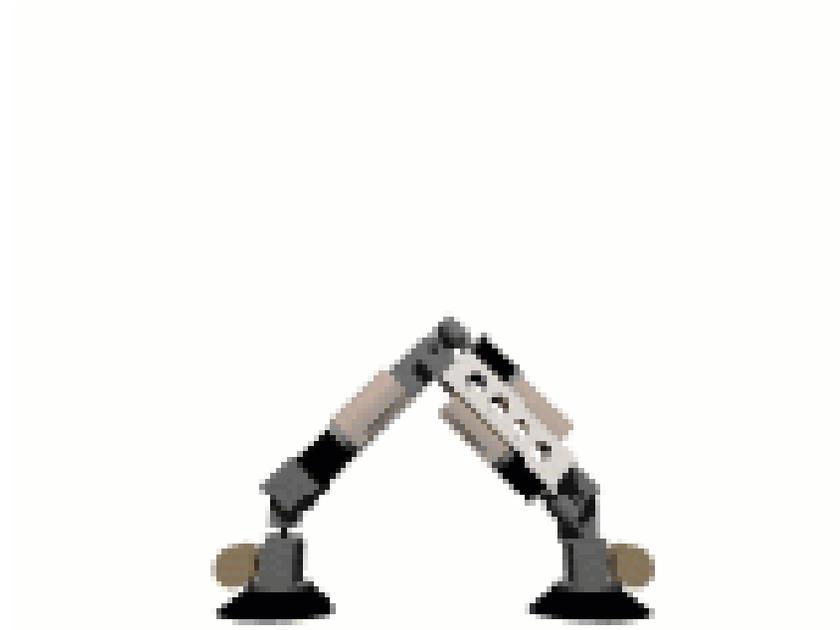


Otilio (UC3M)



Tipos de robots

- **Robots escaladores**



ROMA-I (UC3M)



Tipos de robots

- **Robots Exteriores**



ROCA (UC3M)



Tipos de robots

- Robots Interiores**



B21



Koala



Magellan Pro



Hemisson

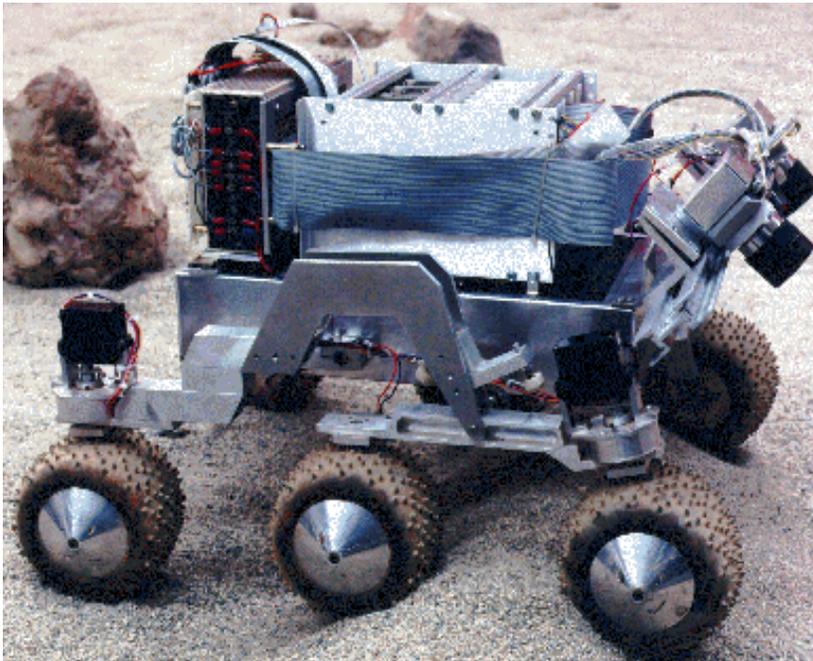


Khepera



Tipos de robots

- **Robots Espaciales**

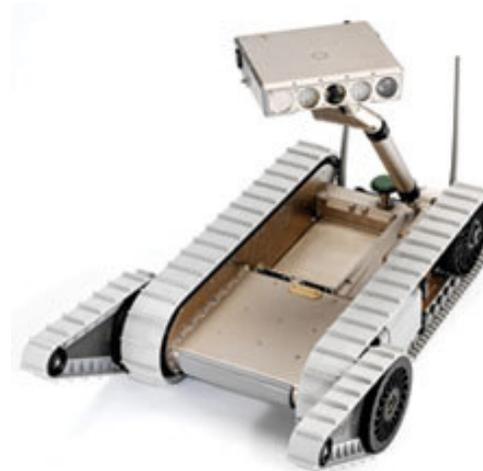
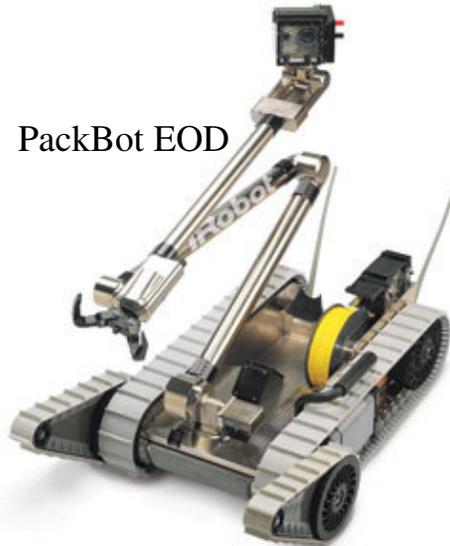




Tipos de robots

- **Robots Militares**

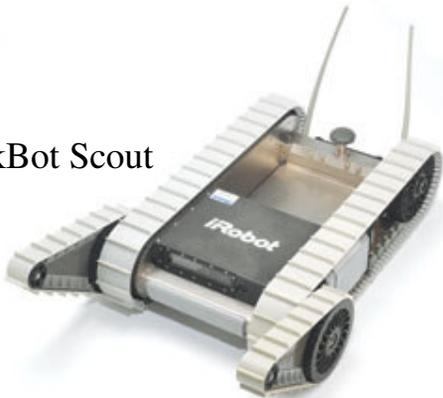
PackBot EOD



PackBot Explorer



PackBot Scout



R-Gator: Autonomous Unmanned Ground Vehicle

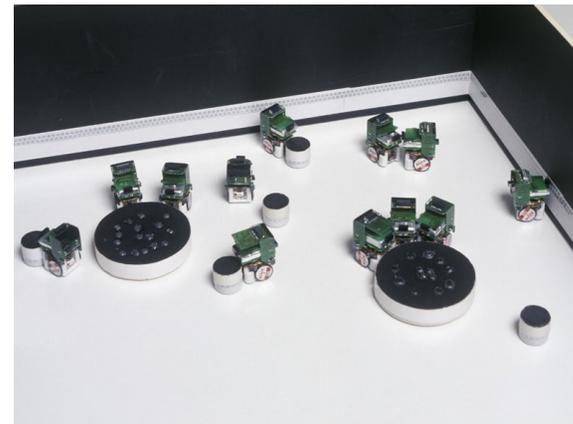


Tipos de robots

- Insectoides



Genghis II



Hormigas artificiales

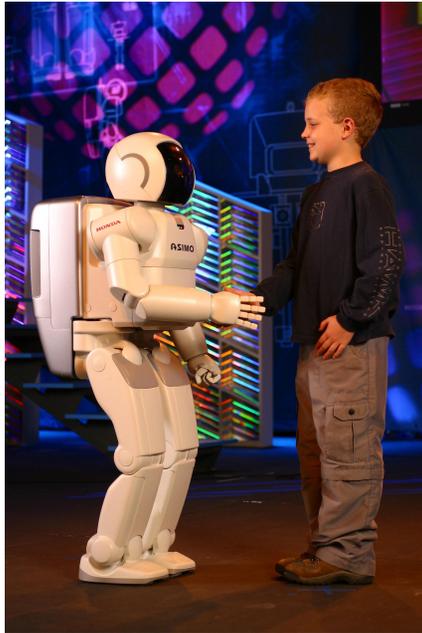


MIT AI Lab

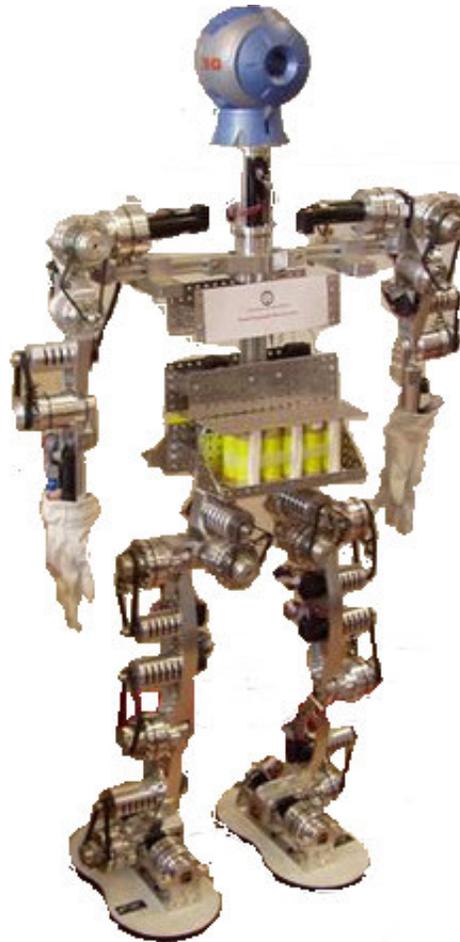


Tipos de robots

- **Humanoides**



ASIMO



RH (UC3M)



QRIO



Wakamaru



Tipos de robots

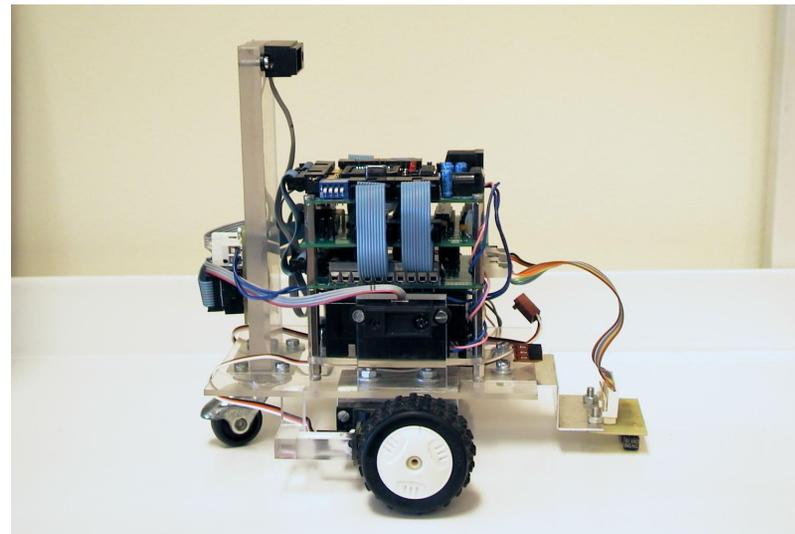
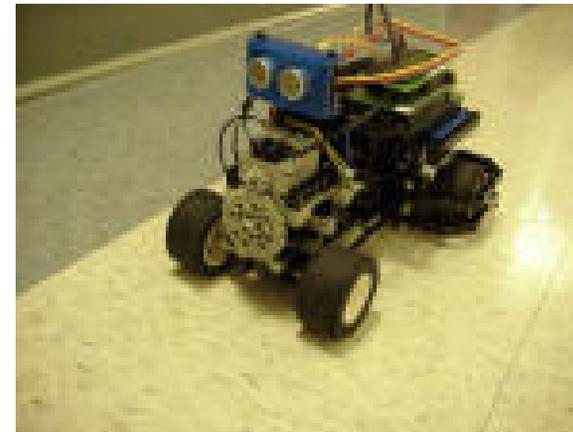
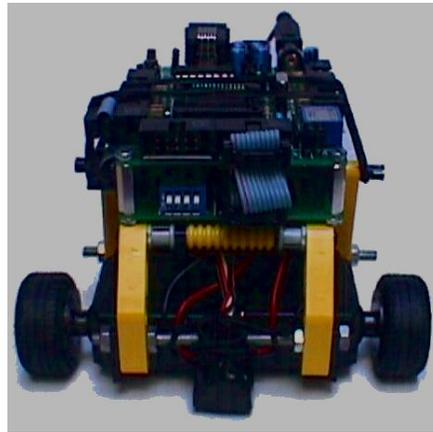
- **Submarinos**





Tipos de robots

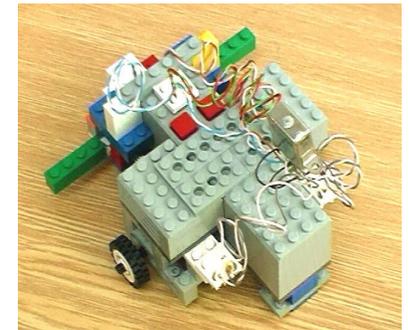
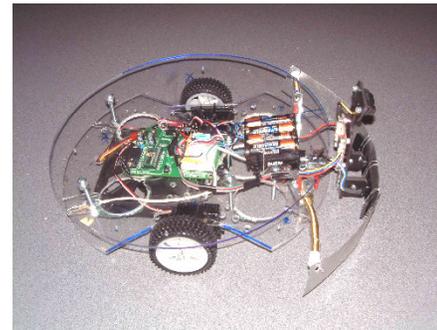
- **Microbots**





Tipos de robots

- Robótica de afición



<http://www.depeca.uah.es/alcabot/hispabot/index.html>

<http://www.usfirst.org/robotics/>



Tipos de robots

- **Nano robots**

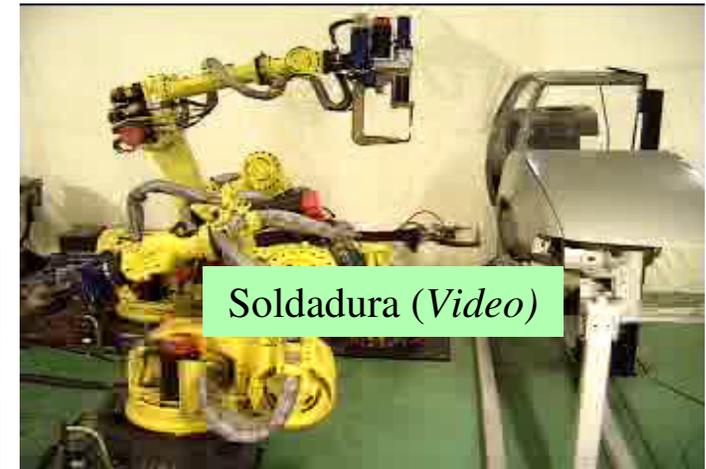




Aplicaciones

● Industria

- ▶ Manipulación de materiales (Pick&Place)
- ▶ Paletizado
- ▶ Maquinas de cargas y descargas
- ▶ Soldadura
- ▶ Pintura
- ▶ Ensamblado
- ▶ Inspección



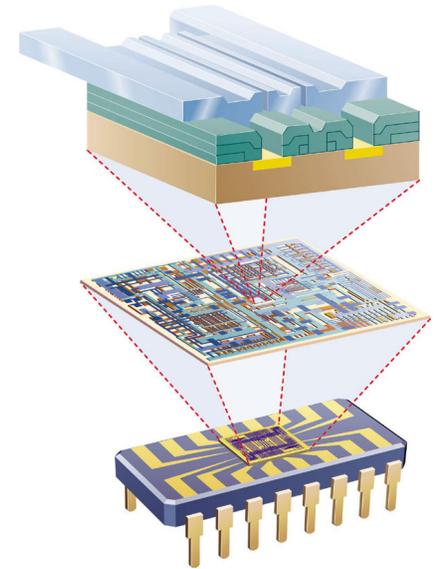


Automatización Industrial -II

Aplicaciones

- **Industria**

Fabricación de circuitos integrados
Video





Aplicaciones

- Agrícolas



Demeter
Ag-robots Farm Robots
Illinois University



National Robotics Engineering Consortium
<http://www.rec.ri.cmu.edu/projects/demeter/demeter.shtml>



Aplicaciones

- Servicios Limpieza



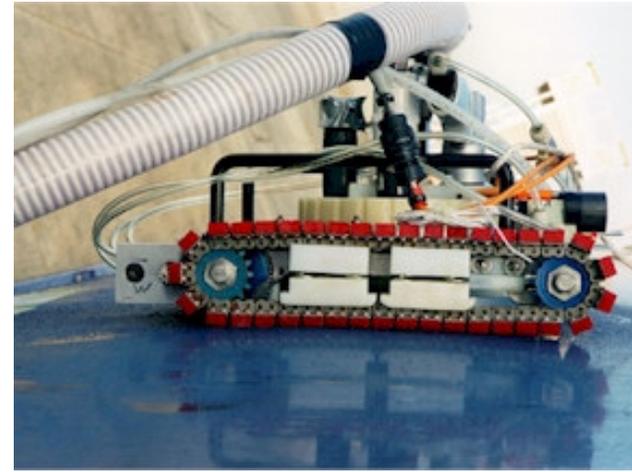


Aplicaciones

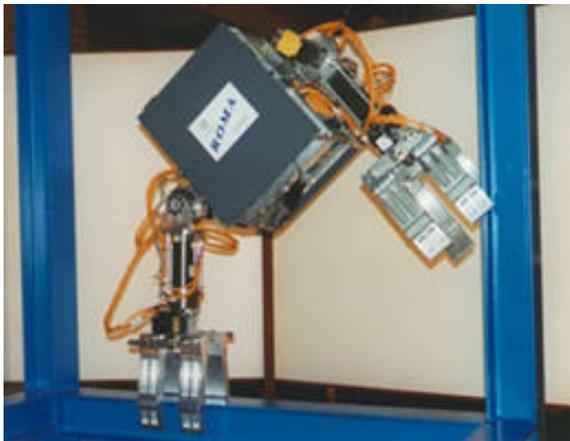
- **Servicios Mantenimiento**



M100-1



M400-1



ROMA-I (UC3M)



Automatización Industrial -II

Aplicaciones

- Servicios Asistencia



MATS (UC3M)



Maggie (UC3M)



RI-MAN

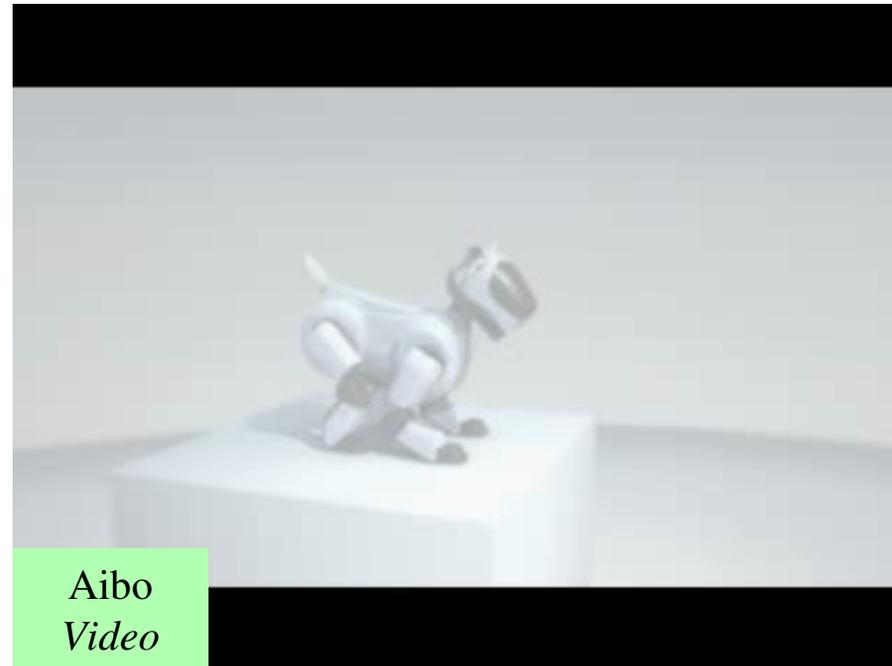


Nursebot



Aplicaciones

- **Servicios Entrenamiento**

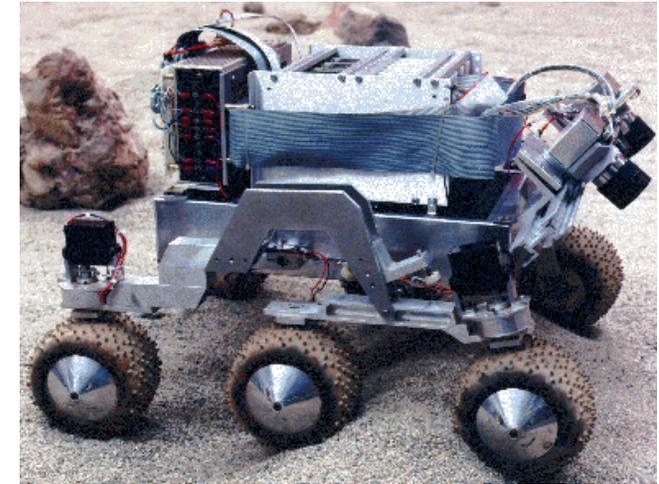




Automatización Industrial -II

Aplicaciones

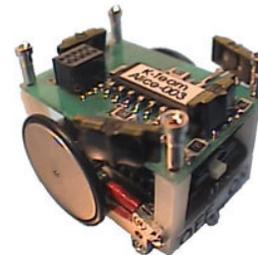
- Exploración





Aplicaciones

- Investigación





Aplicaciones

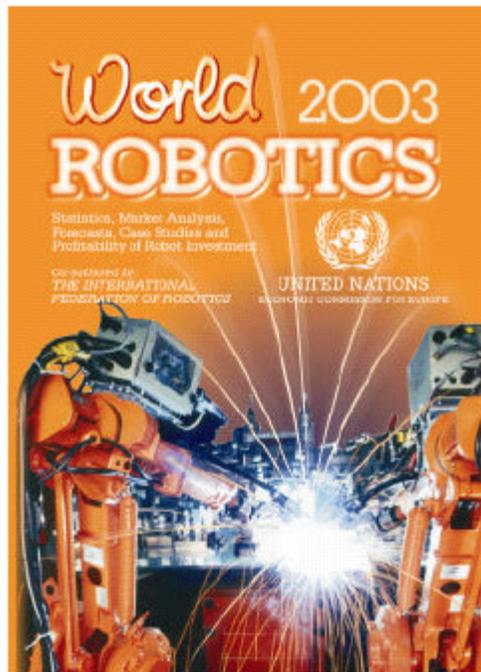
● Soccer Robots





Automatización Industrial -II

Datos estadísticos: Robots en operación

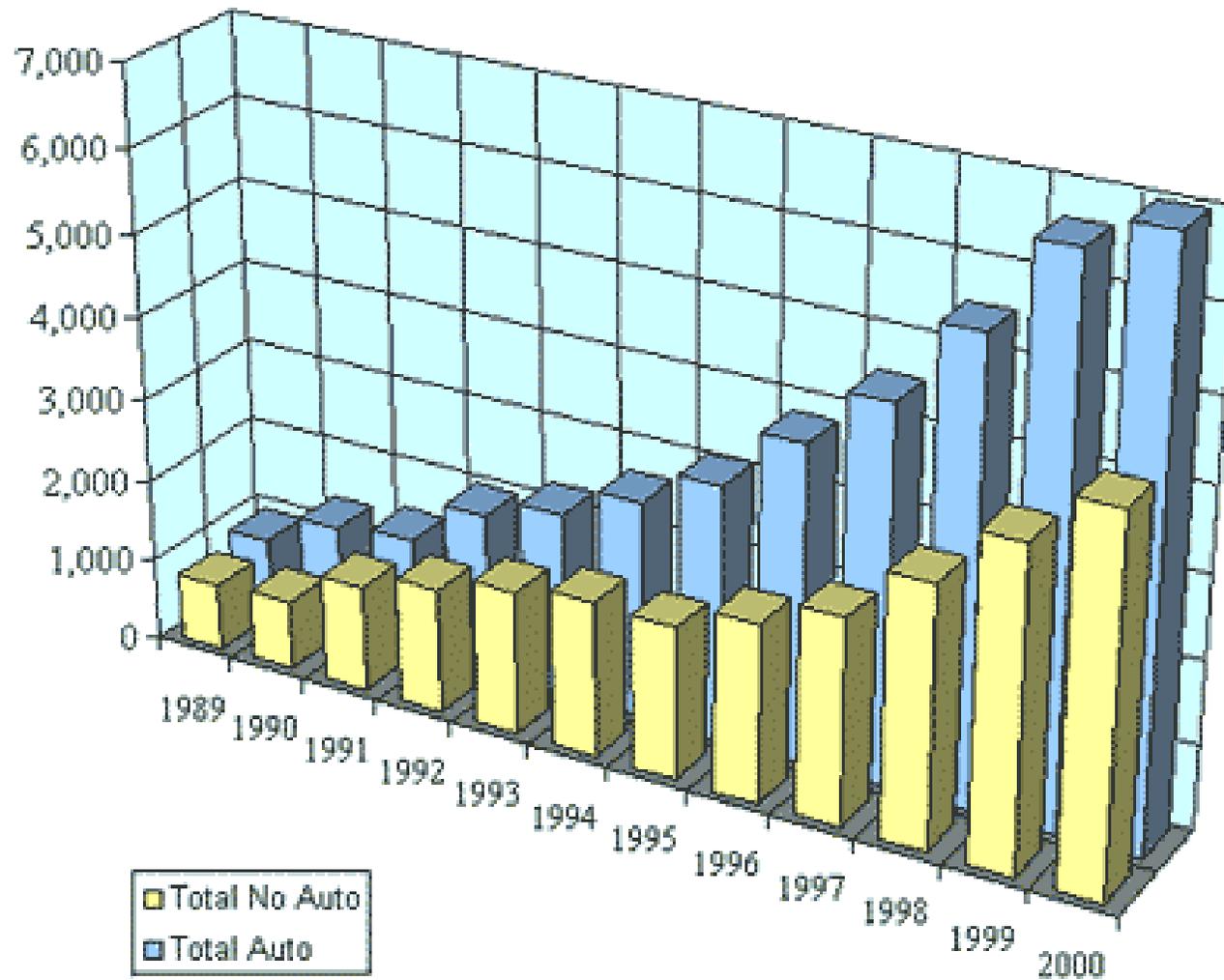


Country	Yearly installations				Operational stock at year-end			
	2001	2002	2003	2006	2001	2002	2003	2006
Japan	28,369	25,373	27,300	33,900	361,232	350,169	344,000	333,400
United States	10,813	9,955	11,400	14,500	97,257	103,515	111,100	135,200
European Union	30,735	25,866	26,600	31,800	219,515	233,139	248,100	303,500
Germany	12,706	11,867	12,000	13,900	99,195	105,217	111,300	136,400
Italy	6,373	5,470	5,700	6,600	43,911	46,881	50,500	62,000
France	3,484	3,012	2,900	3,300	22,753	24,277	25,900	31,700
United Kingdom	1,941	750	800	1,100	13,411	13,651	13,700	14,400
Austria a/	330	670			3,153	3,521		
Benelux a/	620	620			8,590	8,674		
Denmark	330	249			1,883	1,853		
Finland	408	248			2,927	3,023		
Portugal	100	100			800	844		
Spain	3,584	2,420			16,378	18,352		
Sweden	859	460			6,714	6,846		
Other Europe	698	744	800	1,100	11,002	11,013	10,500	12,100
Czech Rep. a/	70	90			985	1,025		
Hungary	27	64			120	176		
Norway	98	80			618	664		
Poland	20	150			520	644		
Russian Fed. a/	150	190			5,000	5,000		
Slovakia b/								
Slovenia b/								
Switzerland a/	333	170			3,759	3,504		
Asia/Australia	5,310	5,108	5,600	7,500	56,997	60,412	64,300	73,300
Australia	270	510			2,953	3,310		
Rep. of Korea (all types of industrial robots)	4,080	3,998			41,267	44,265		
Singapore a/	300	100			5,458	5,346		
Taiwan, Province of China	680	500			7,319	7,491		
Other countries a/	2,250	1,520	1,700	2,300	10,374	11,640	12,900	17,800
Subtotal, excl. Japan and Rep. of Korea	45,726	39,195	41,800	51,900	353,878	375,454	446,900	541,900
Total, including Japan and Rep. of Korea	78,175	68,566	73,400	91,100	756,377	769,888	838,400	875,300

Sources: UNECE, IFR and national robot associations.

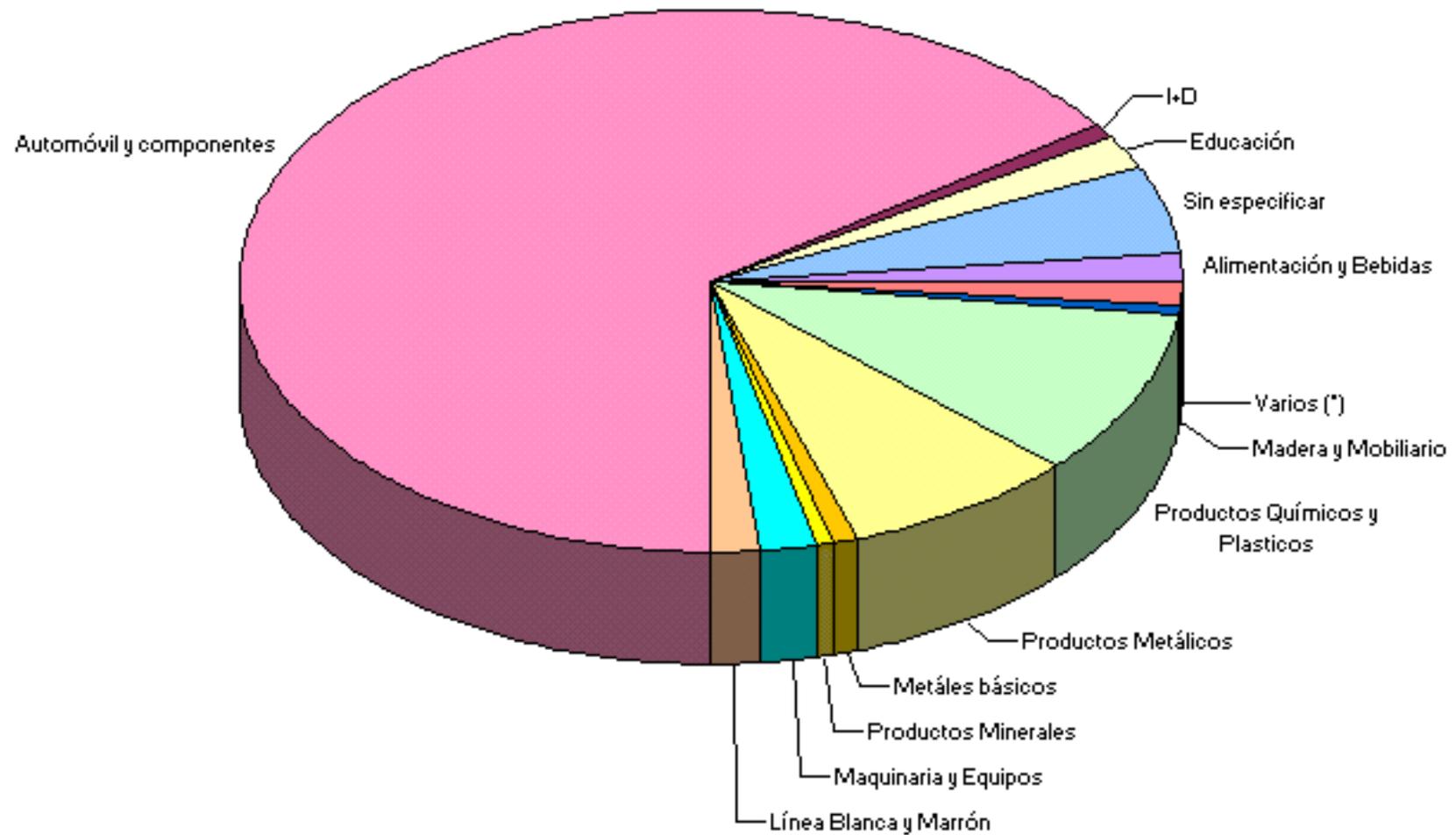


Datos estadísticos: Robots en operación España



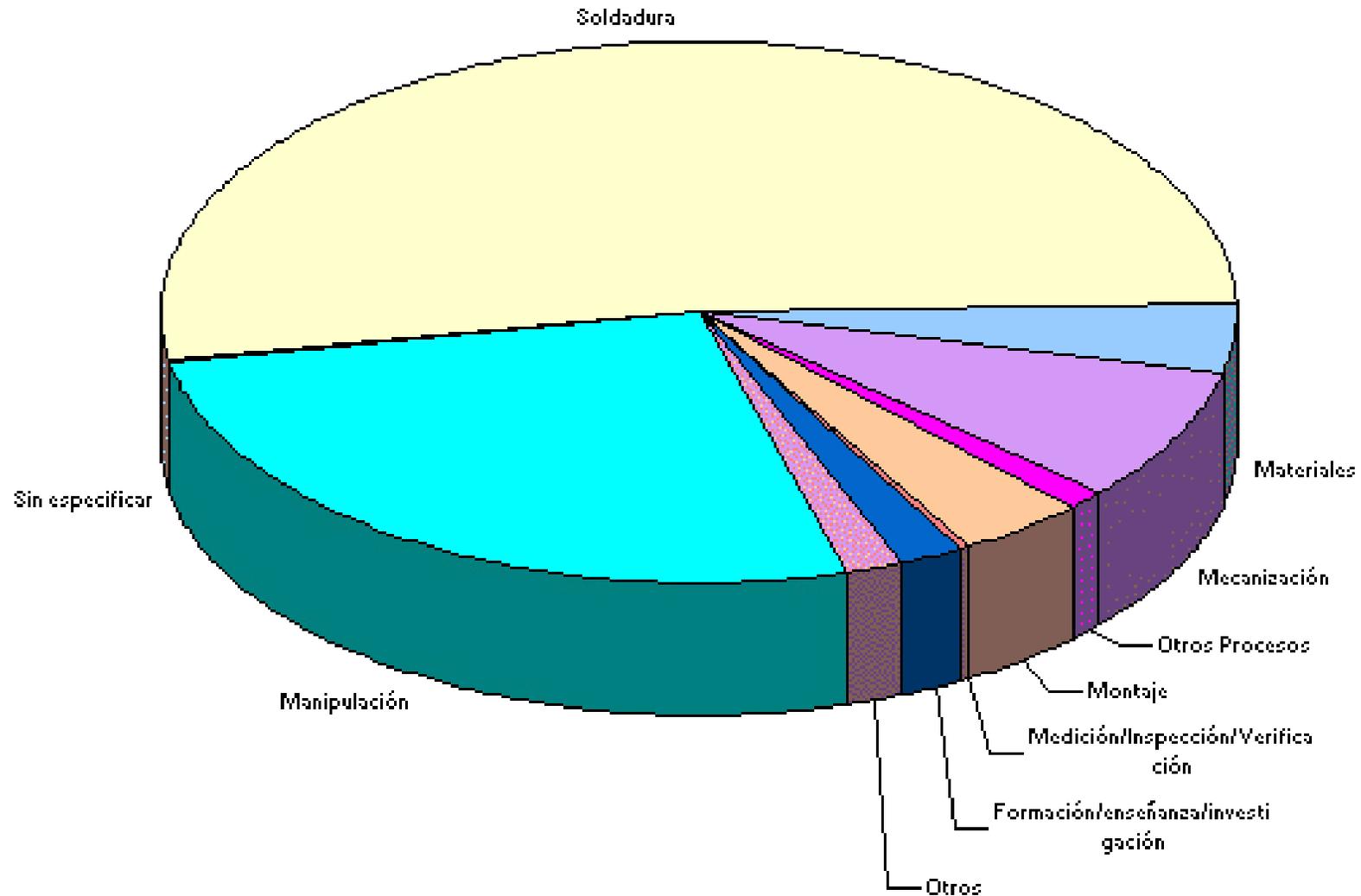


Datos estadísticos: Sectores en España





Datos estadísticos: Aplicaciones en España





Datos estadísticos: Densidad de robots

Definición

Número de robots instalados por cada 10.000 operarios en la industria manufacturera.

Refleja el grado de automatización y competitividad del tejido industrial productivo

Países	Densidad de robots	Indice de paro	Número de robots instalados
Japón	250	<3%	413.578
Estados Unidos	35	<6%	65.198
Alemania	58	<9%	56.175
Francia	33	<12%	14376
Italia	55	<12%	25.096
Suecia	60	<9%	5.911
España	22	<20%	5.346



Resumen:

➤ Un **robot industrial**, es un manipulador multi-funcional re-programable, capaz de mover cargas, piezas, herramientas o dispositivos especiales, según trayectorias varias, programadas para la ejecución de tareas diversas.

➤ Tipos de los robots:

- ▶ Robots Industriales
- ▶ Robots escaladores
- ▶ Robots Exteriores
- ▶ Robots Interiores
- ▶ Robots Espaciales
- ▶ Robots Militares
- ▶ Humanoides
- ▶ Submarinos
- ▶ Insectoides
- ▶ Microrobots
- ▶ Hoppy robots
- ▶ Nanorobots

➤ Aplicaciones:

- ▶ Industria
- ▶ Agricultura
- ▶ Servicios Limpieza
- ▶ Servicios Mantenimiento
- ▶ Servicios Asistencia
- ▶ Servicios Entrenamiento
- ▶ Exploración
- ▶ Investigación





Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 2 – Jueves 20 de Abril 2006





Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 3 & 4 – Miércoles & Jueves 26, 27 de Abril 2006

Morfología del robot



Robot Industrial

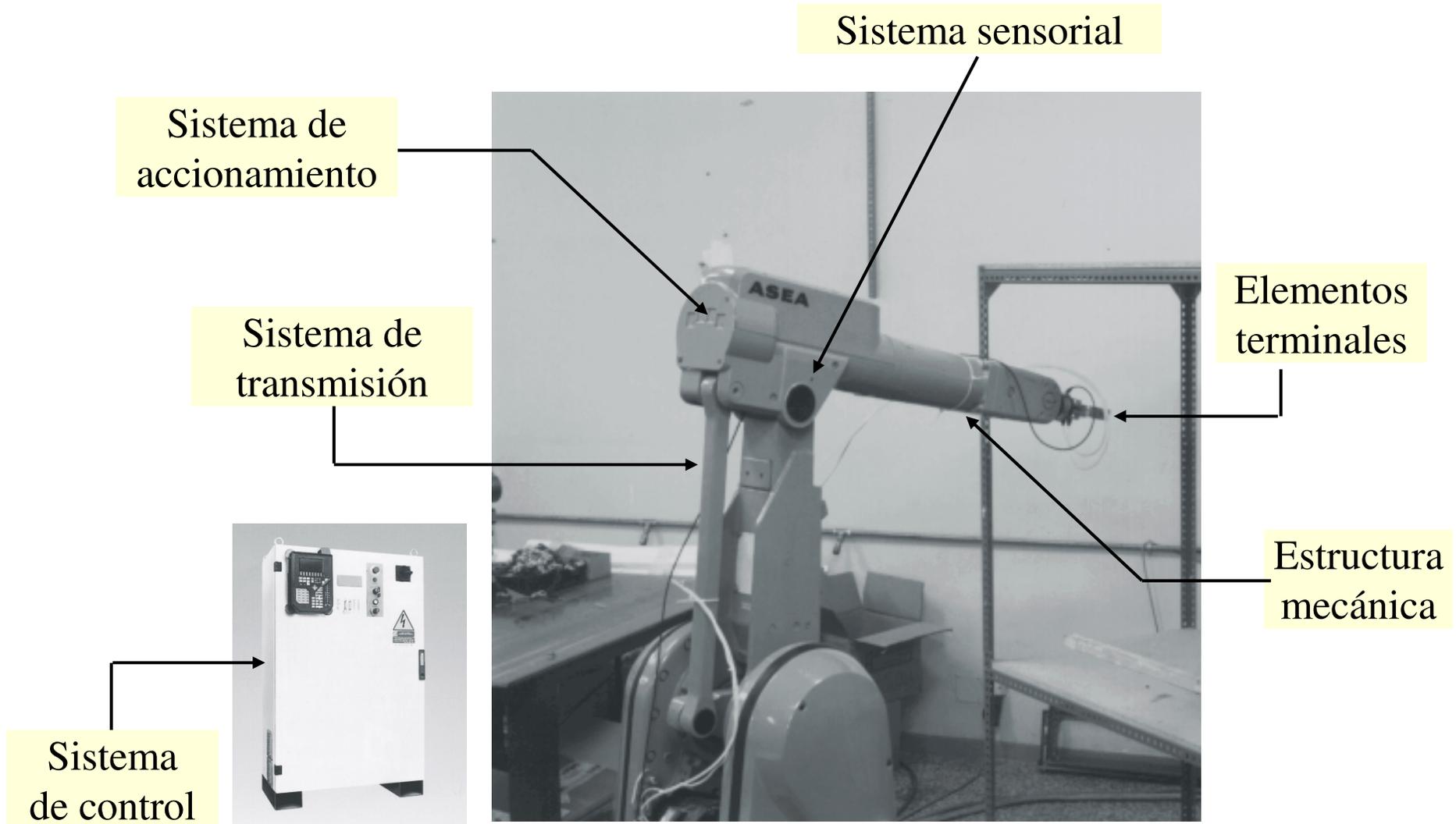
Un **Robot industrial** es un manipulador automático **servocontrolado**, **reprogramable**, **polivalente**, **capaz de posicionar y orientar** piezas, útiles o dispositivos especiales, siguiendo trayectorias variables reprogramables, para la ejecución de **tareas variadas**”.

Asociación Francesa de Normalización (AFNOR)



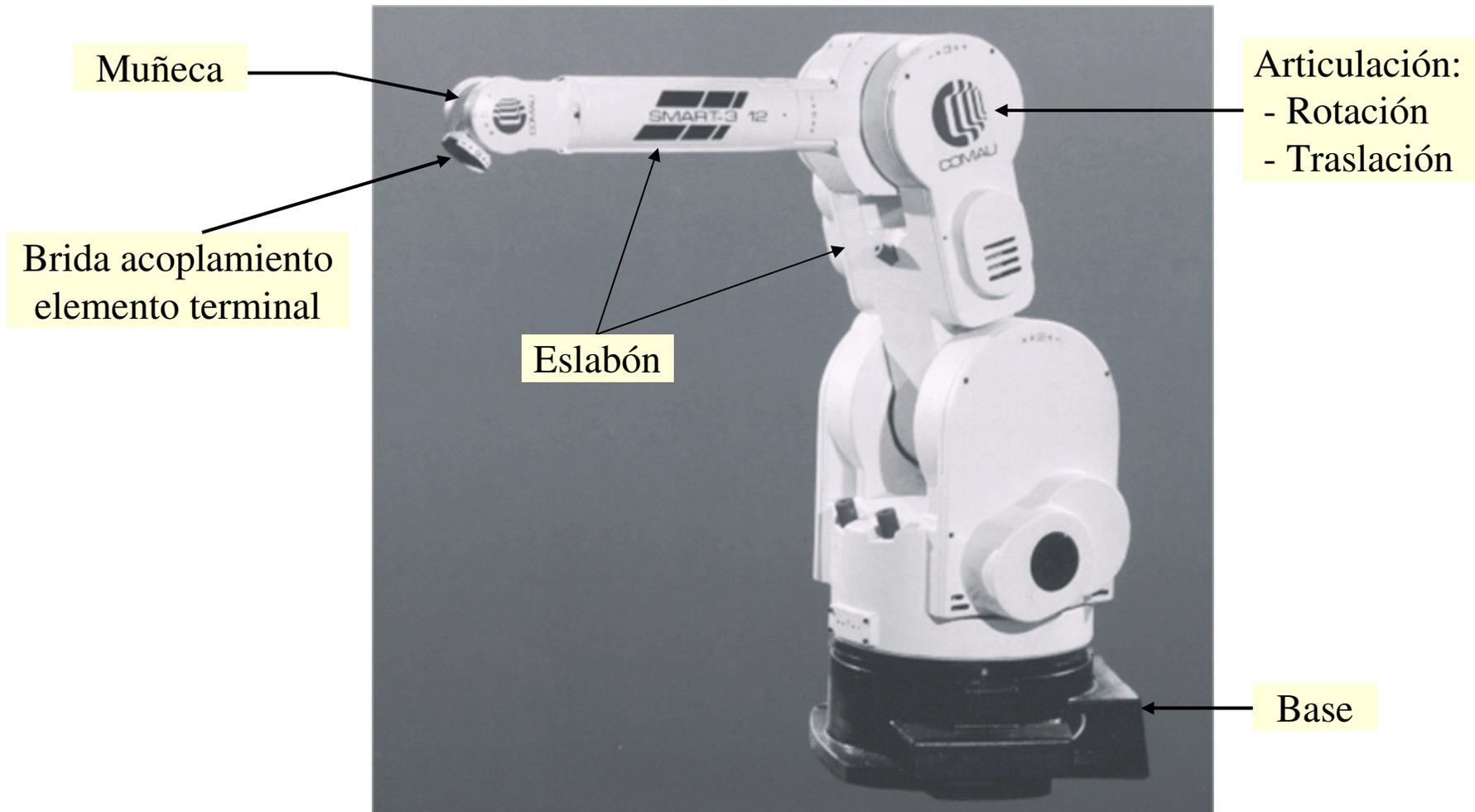


Elementos de un robot industrial





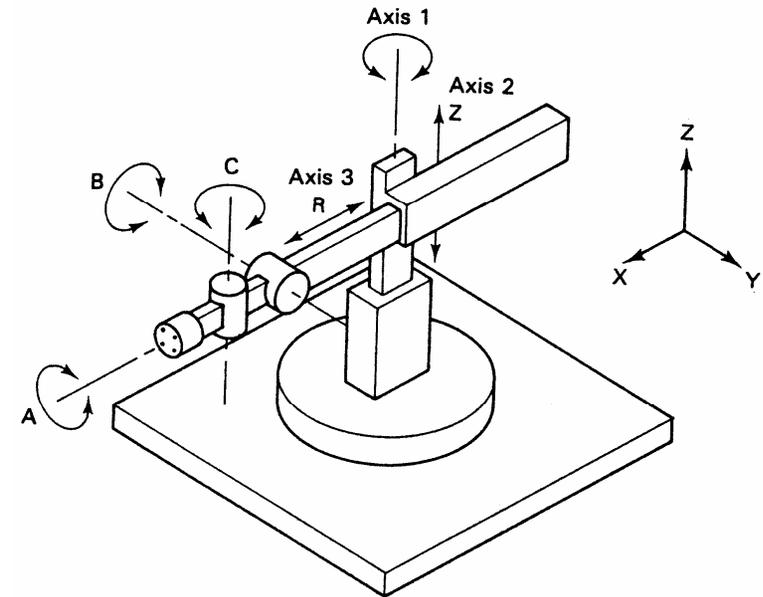
Elementos de un robot industrial: Estructura mecánica





Elementos de un robot industrial: Estructura mecánica

► Mecánicamente, un robot industrial o brazo está formado por **una serie de elementos o eslabones unidos mediante articulaciones** que permiten un movimiento relativo entre cada dos eslabones consecutivos.

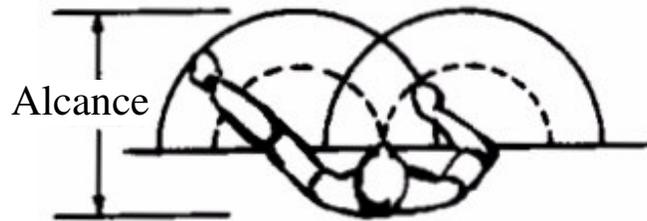


► Normalmente cada articulación permite sólo un movimiento simple, **traslación** o **rotación** en torno a un eje.

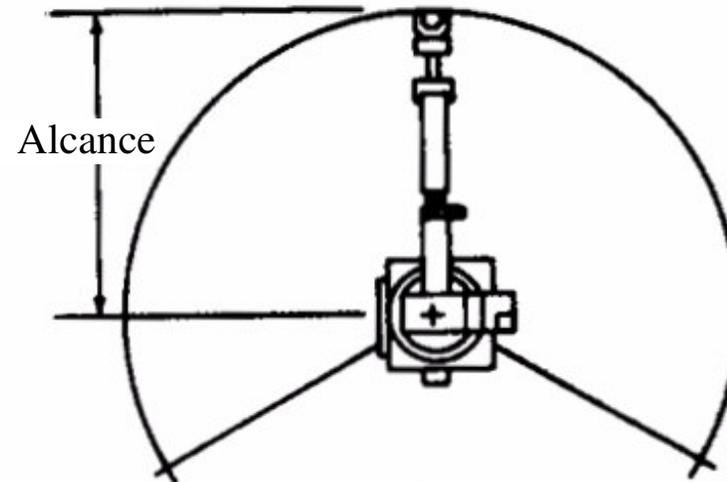


Elementos de un robot industrial: Estructura mecánica

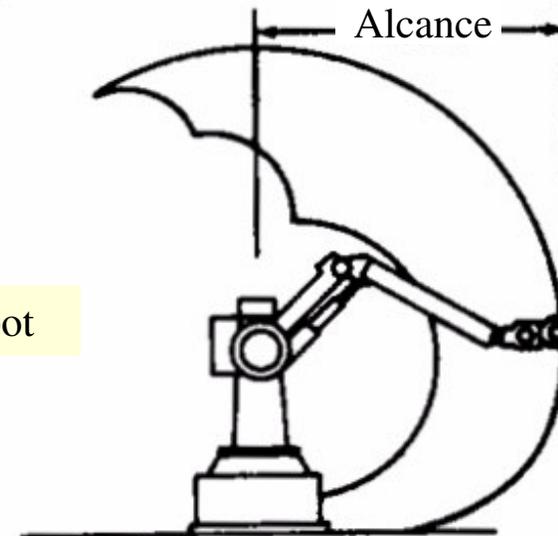
- Campo de acción



Humano



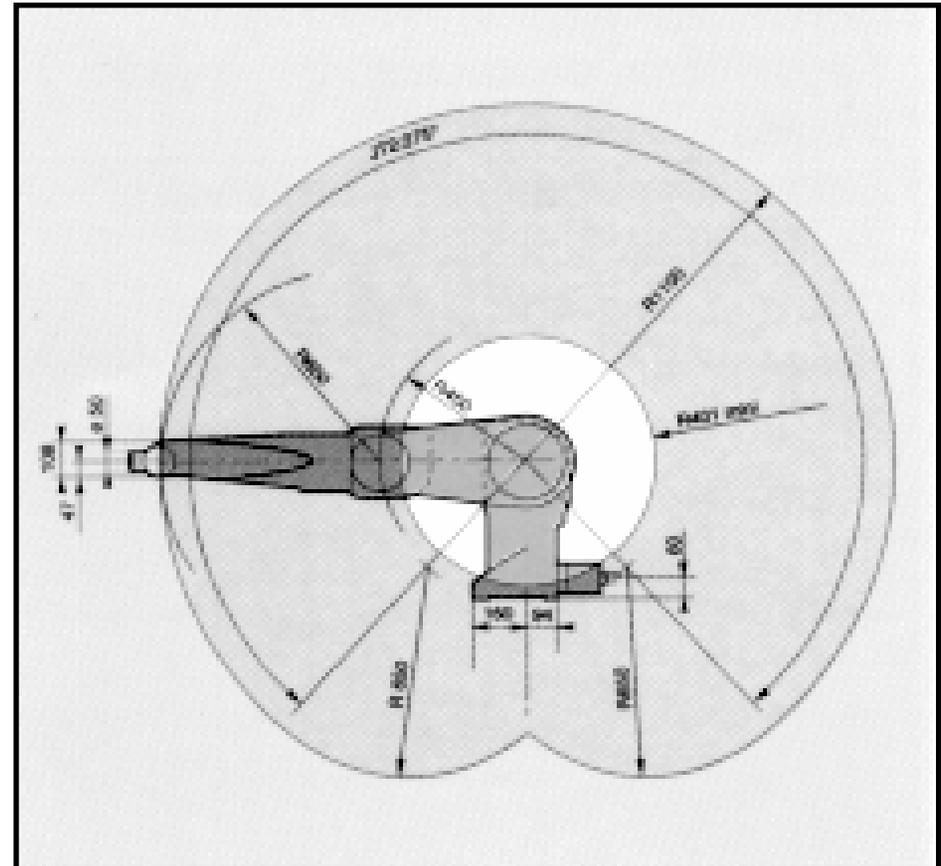
Robot





Elementos de un robot industrial: Estructura mecánica

- Campo de acción



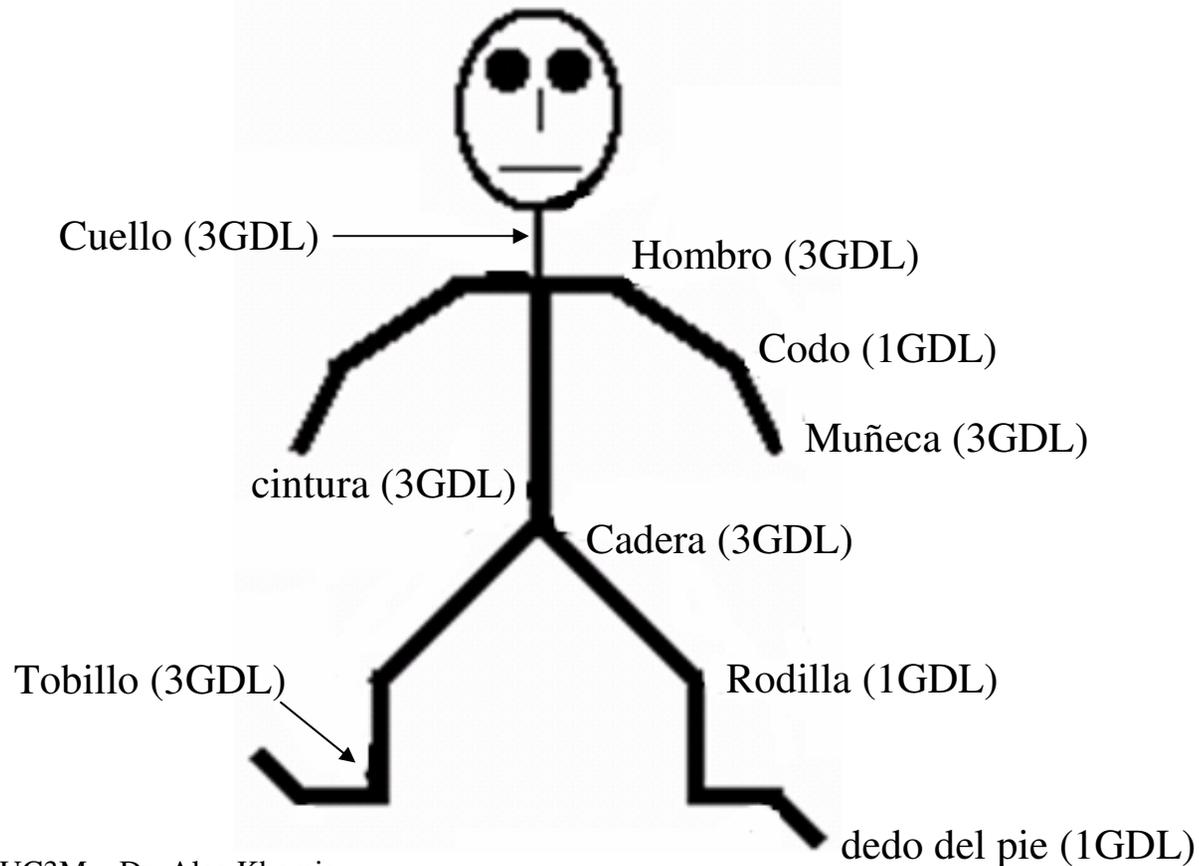
STAUBLI RX 90



Elementos de un robot industrial: Estructura mecánica

● Grados de Libertad (GDL)

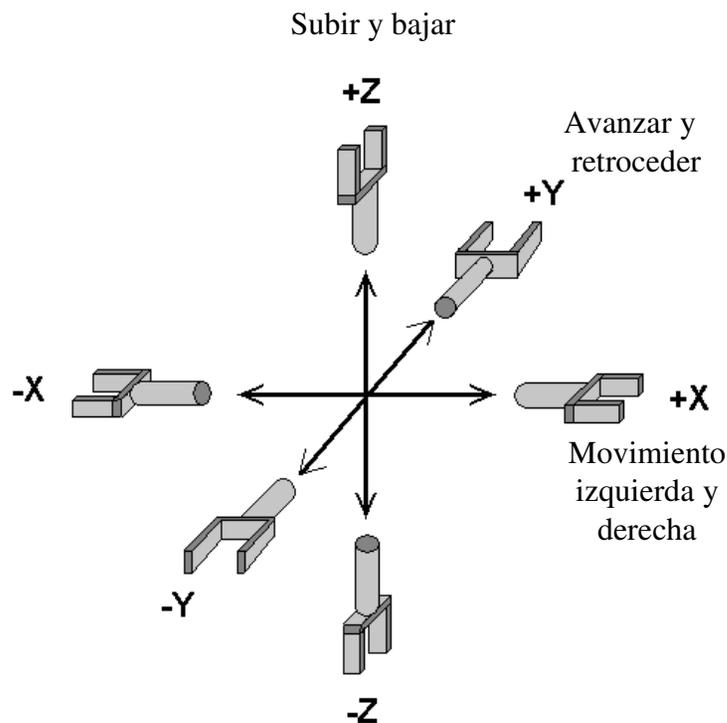
Cada uno de los **movimientos independientes** que puede realizar **cada articulación** con respecto a la anterior, se denomina **grado de libertad (GDL)**.



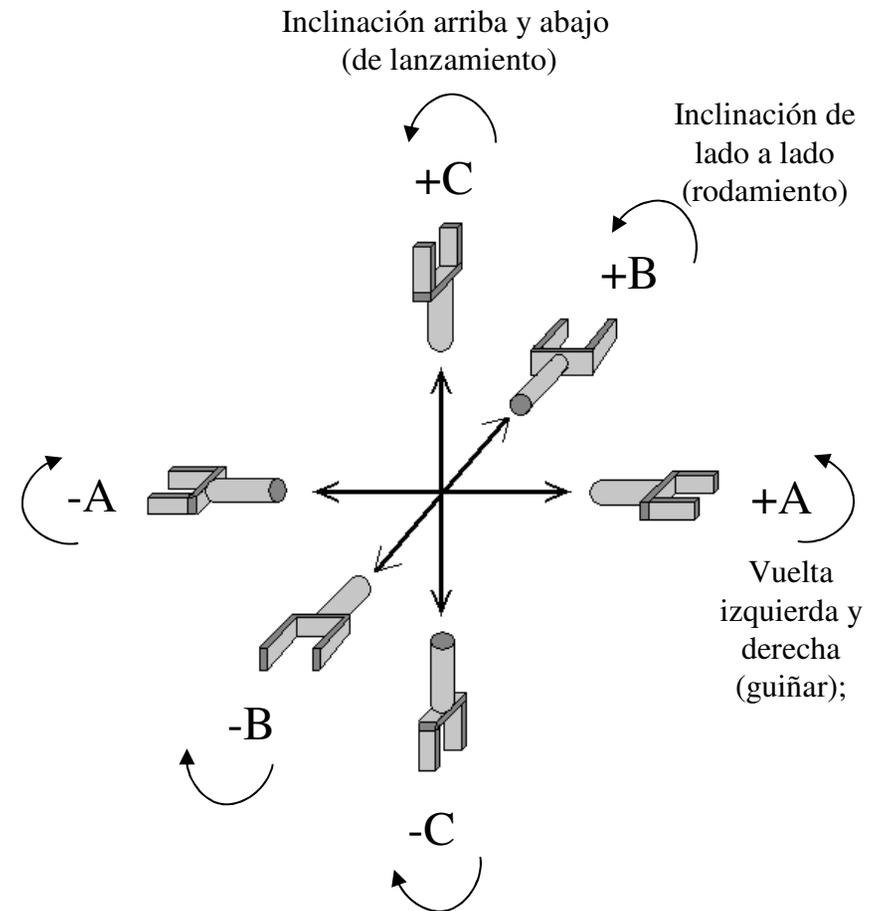


Elementos de un robot industrial: Estructura mecánica

● Grados de Libertad (GDL)



3-GDL (Posición)

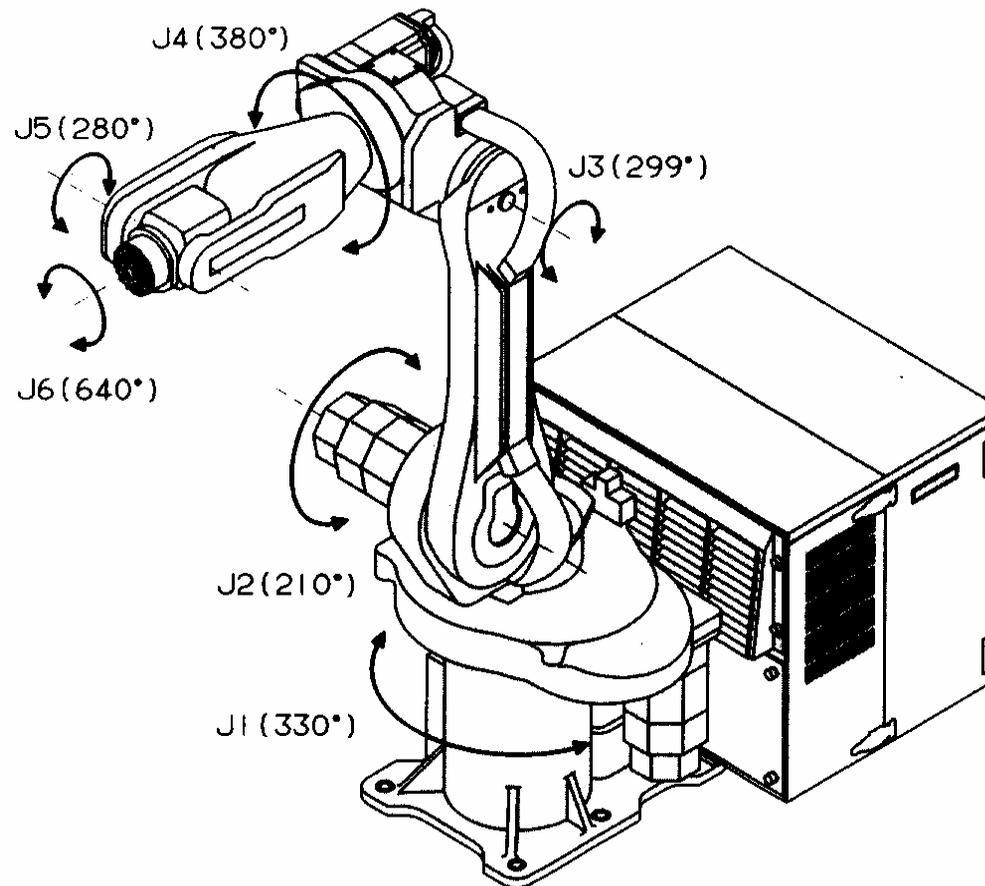


3-GDL (Orientación)



Elementos de un robot industrial: Estructura mecánica

- Grados de Libertad (GDL)



FANUC ARC Mate 100i



Elementos de un robot industrial: Estructura mecánica

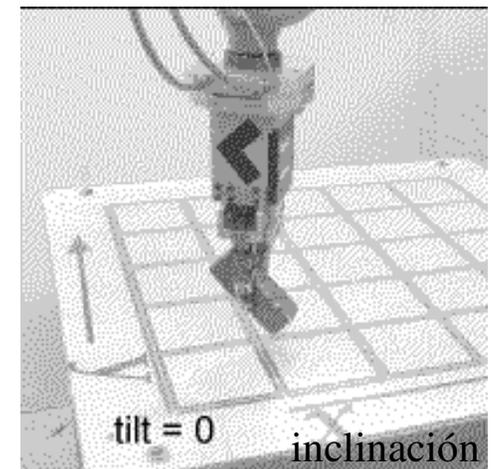
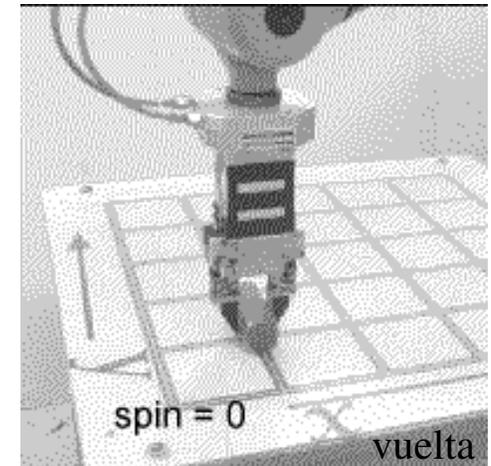
● Grados de Libertad (GDL)

► El número de grados de libertad (GDL) de un robot industrial es el **número de movimientos independientes** que puede realizar el robot respecto a su base:

3 GDL - es el mínimo (según definición robot)

6 GDL - es el máximo (para posicionar y orientar)
(Robot Holonomico)

7+ GDL – redundancia (Robot Redundante)



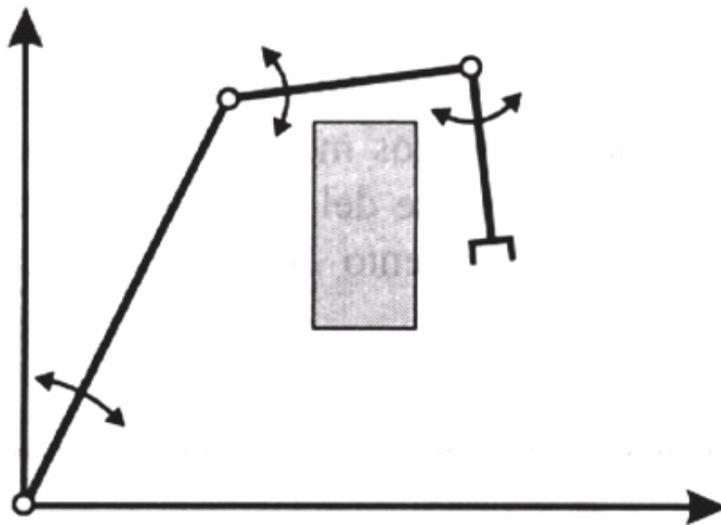


Elementos de un robot industrial: Estructura mecánica

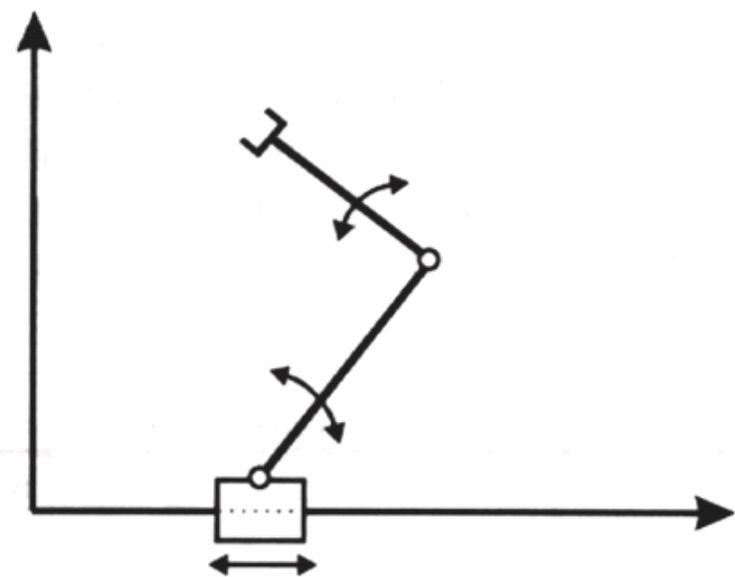
● Grados de Libertad (GDL)

▶ Redundancia

En la práctica, a pesar de ser necesarios los 6 GDL para tener total libertad en el posicionado y orientación del extremo del robot, muchos robots industriales cuentan con sólo 4 o 5 GDL.



Robot plano con 3GDL para aumentar su maniobrabilidad



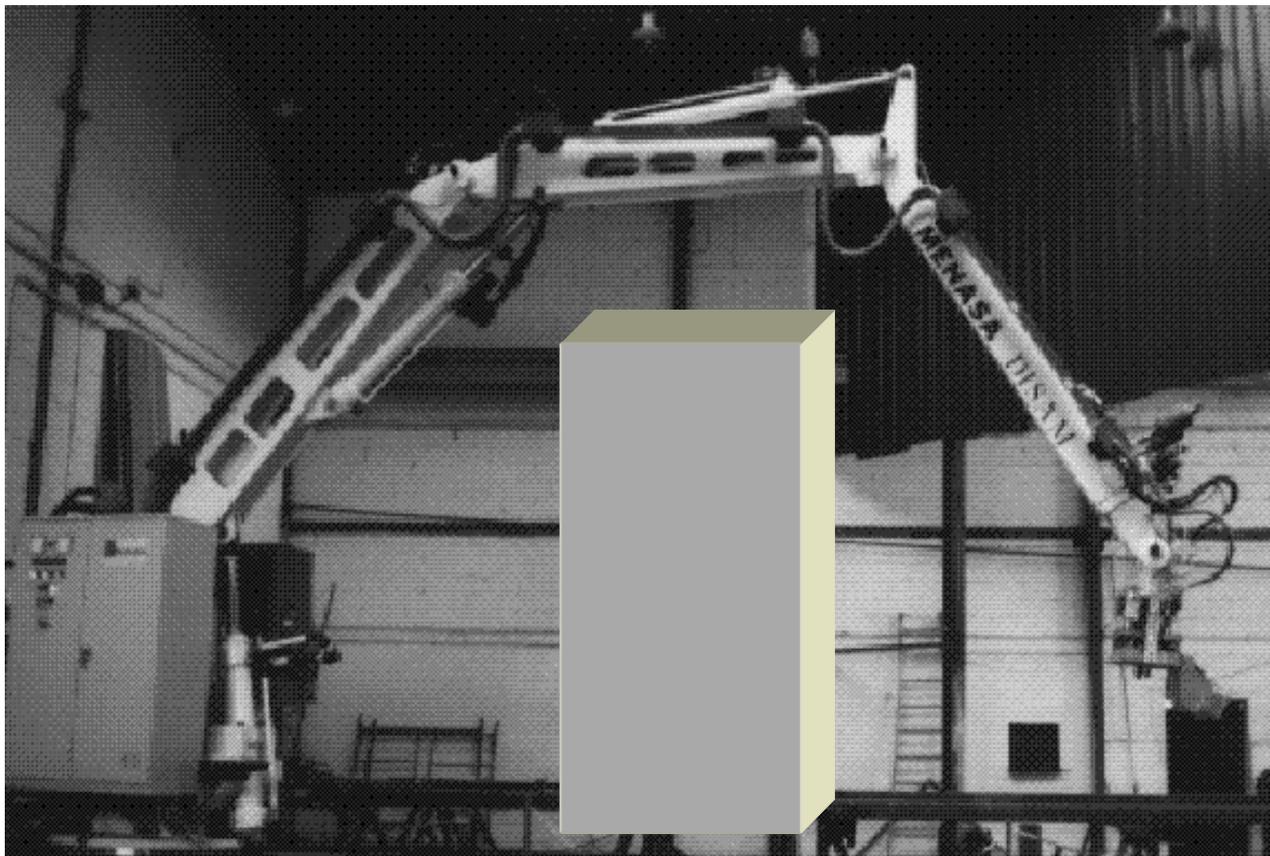
Robot plano con 3GDL para aumentar su campo de acción



Elementos de un robot industrial: Estructura mecánica

- **Grados de Libertad (GDL)**

- ▶ Redundancia: Mayor maniobrabilidad 6GDL (4 posición+2 orientación)



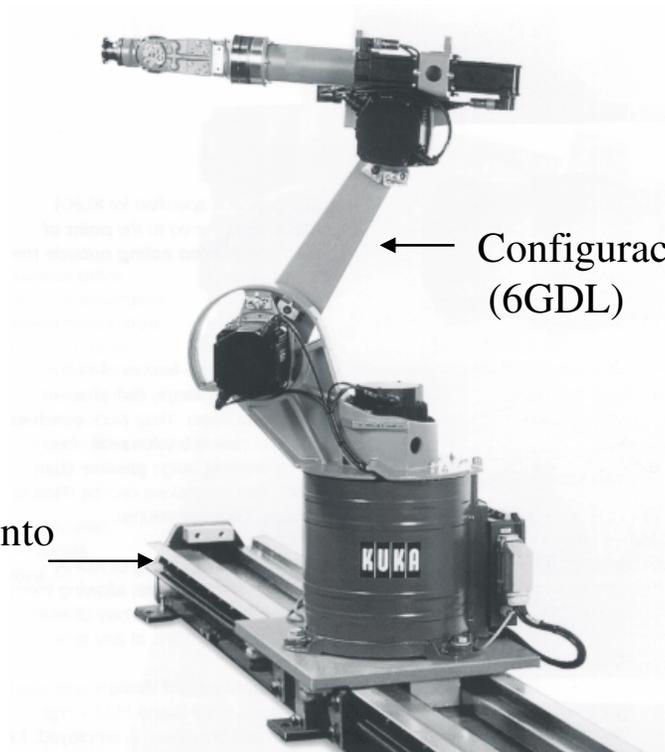
ROCCO



Elementos de un robot industrial: Estructura mecánica

- **Grados de Libertad (GDL)**

- ▶ Redundancia: Aumento del campo de acción (raíl)



KUKA KR 6

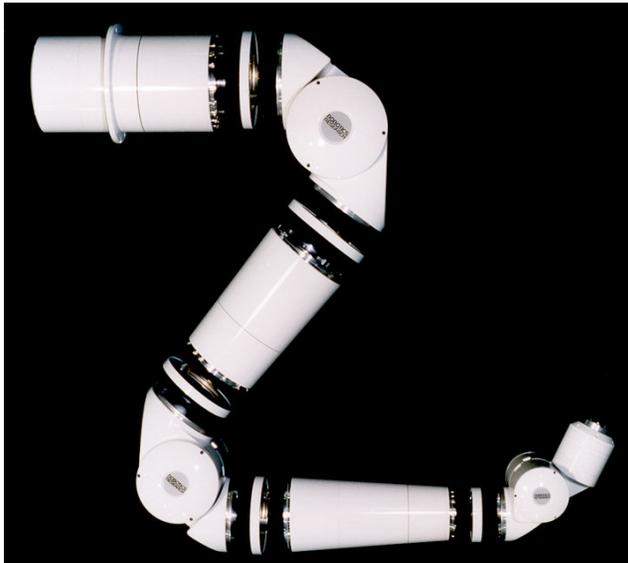


CRS T265



Elementos de un robot industrial: Estructura mecánica

- Grados de Libertad (GDL)



7 GDL



17 GDL



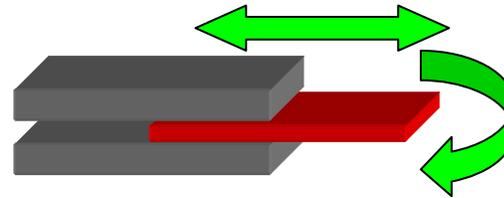
Elementos de un robot industrial: Estructura mecánica

● Articulaciones

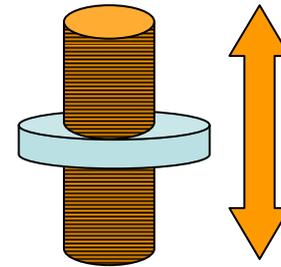
Esférica o Rótula
3 GDL



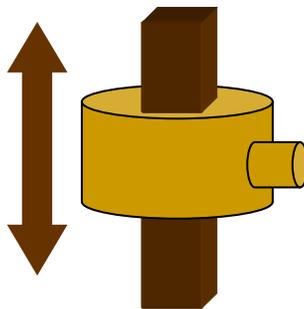
Planar
2 GDL



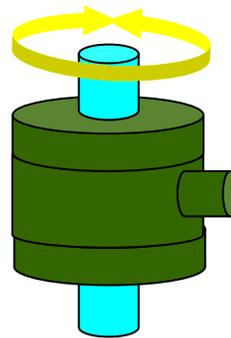
Tornillo
1 GDL



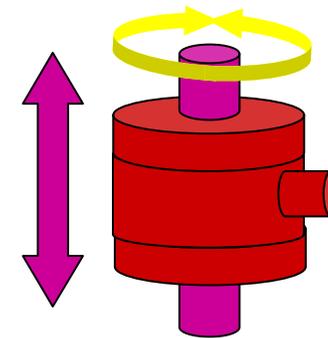
Prismática
1 GDL



Rotación
1 GDL



Cilíndrica
2 GDL



Las más usadas



Elementos de un robot industrial: Estructura mecánica

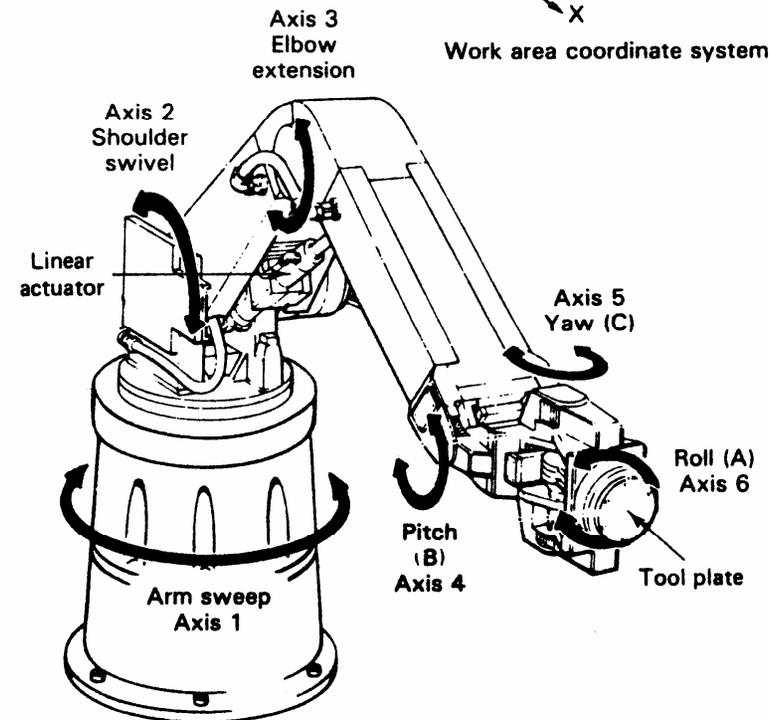
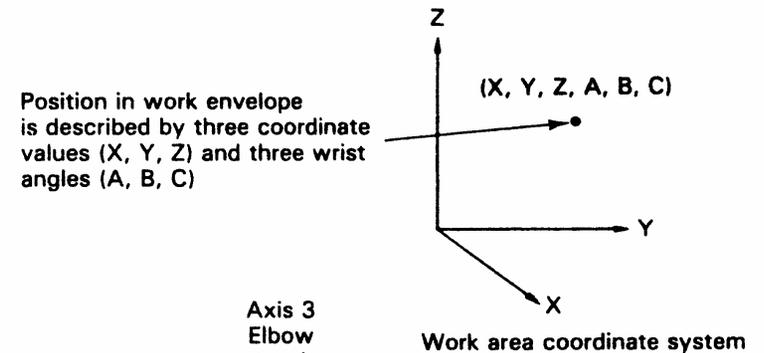
● Sistema de coordenadas

Coordenadas de translación: X, Y, Z

Coordenadas de rotación: A, B, C

● Numeración de ejes

- ▶ Se empieza con la base del robot.
- ▶ El primer eje de movimiento, lo llamamos eje #1.
- ▶ Se incrementa el número sucesivamente hasta llegar al elemento final del manipulador (end-effector).

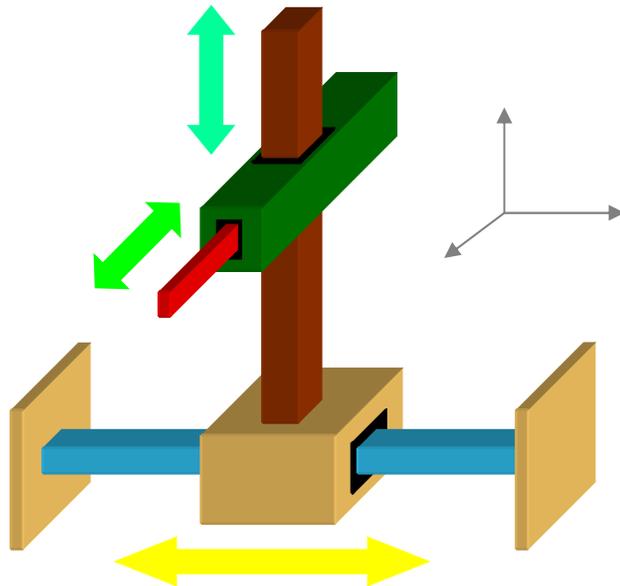




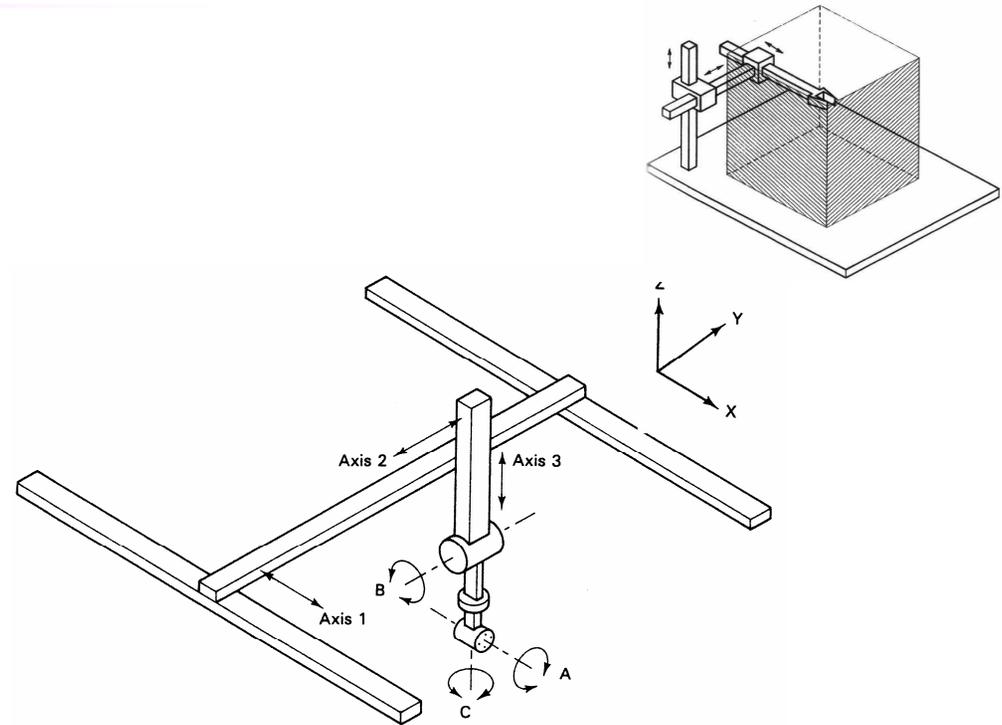
Elementos de un robot industrial: Estructura mecánica

● Configuraciones

☛ Robot cartesiano



Robot Atravesador



Robot Grúa

- ✓ Un campo de acción muy amplio
- ✓ Conserva el espacio
- ✓ Sistema de control simple

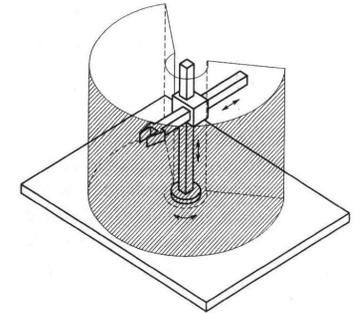
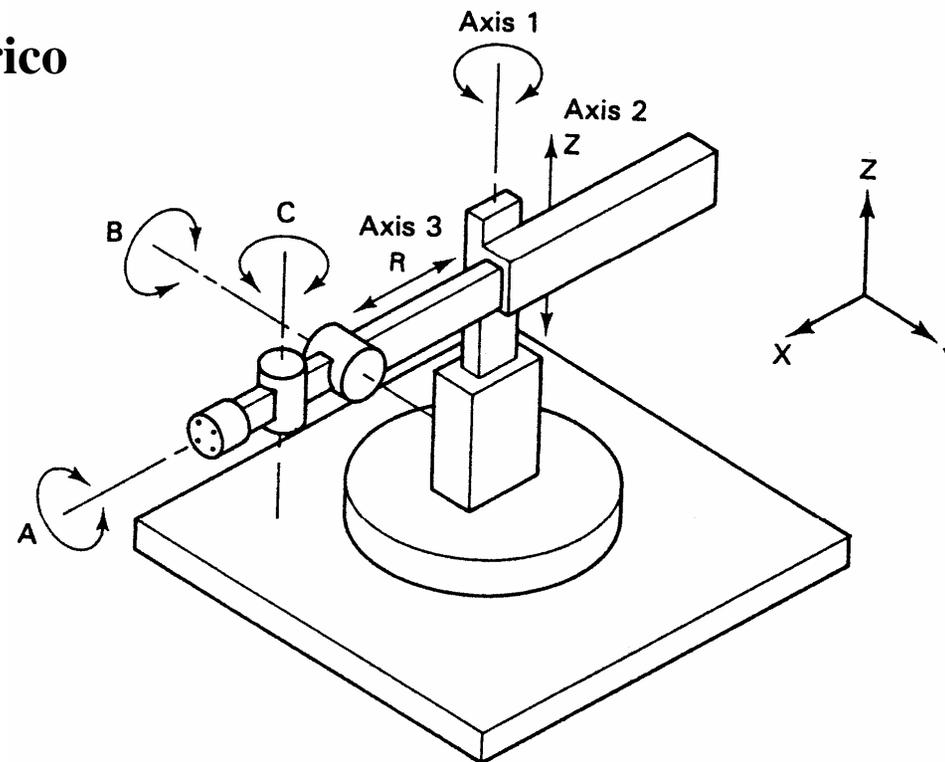
✗ Mantenimiento difícil



Elementos de un robot industrial: Estructura mecánica

● Configuraciones

☛ Robot cilíndrico



- ✓ Alcance horizontal profundo en máquinas de producción
- ✓ La estructura vertical conserva el espacio
- ✓ Cargas grandes y buena repetibilidad

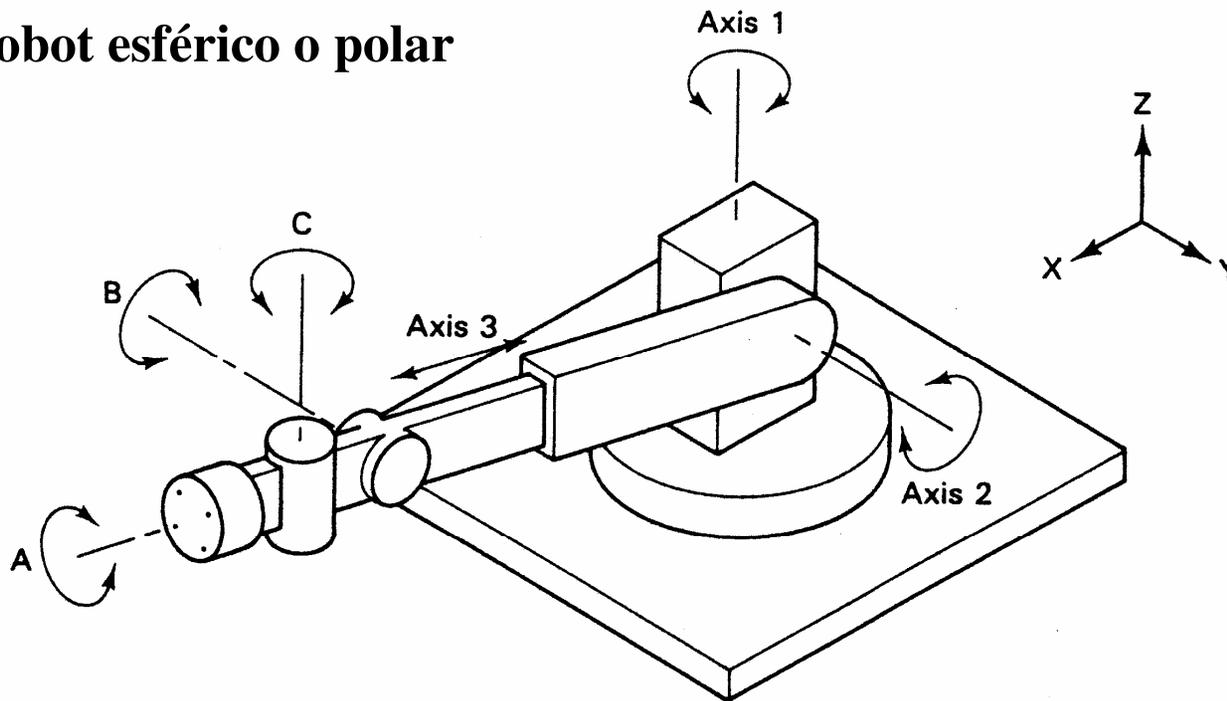
✗ Alcance limitado derecha e izquierda



Elementos de un robot industrial: Estructura mecánica

● Configuraciones

☛ Robot esférico o polar



- ✓ Alcance horizontal profundo en máquinas de producción
- ✓ La estructura vertical conserva el espacio
- ✓ Cargas grandes y buena repetibilidad

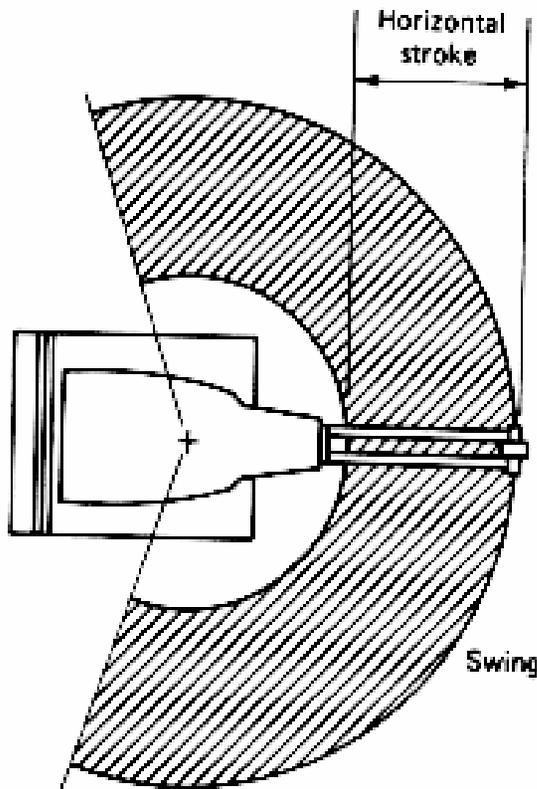
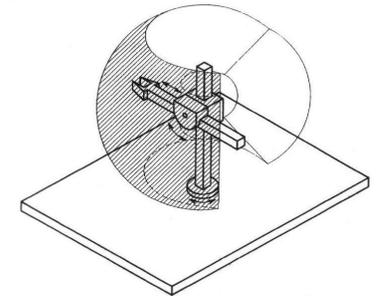
✗ Alcance limitado derecha e izquierda



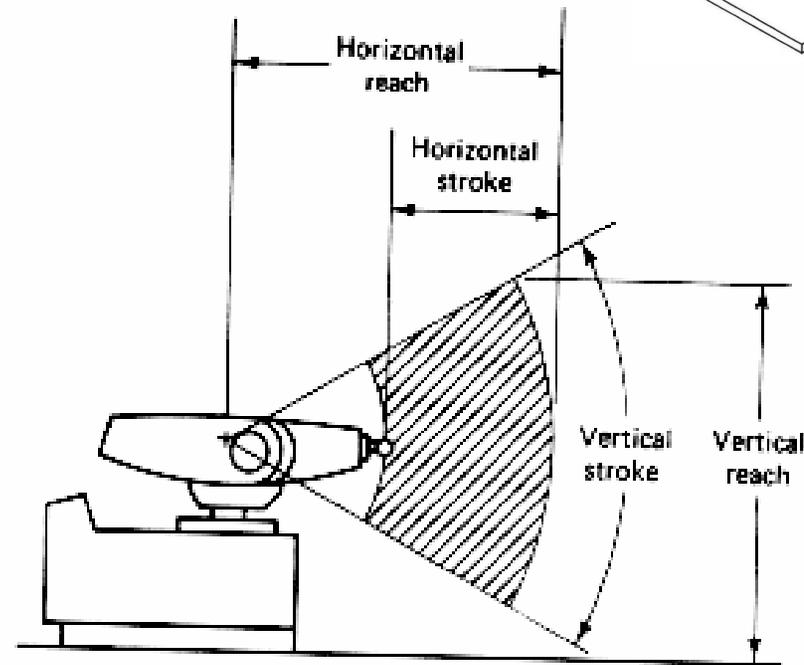
Elementos de un robot industrial: Estructura mecánica

● Configuraciones

■ Robot esférico o polar: Campo de acción



(a) Plan



(b) Elevation

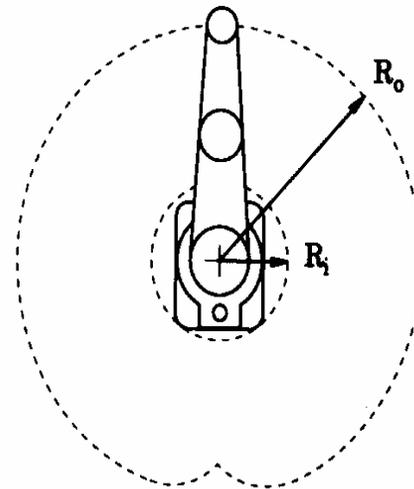
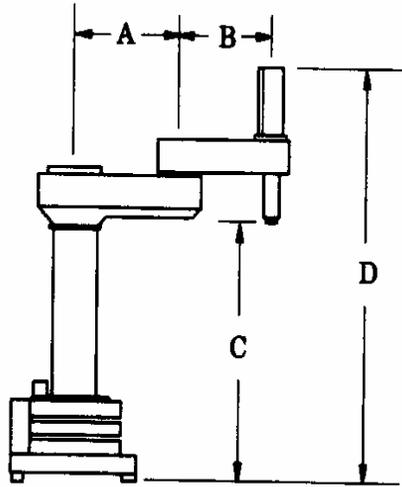
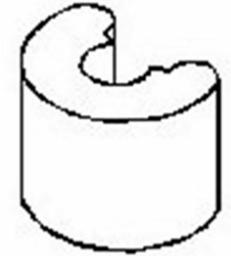
UNIMATION 2000



Elementos de un robot industrial: Estructura mecánica

● Configuraciones

☛ Robot Horizontalmente Articulado



SCARA (Selective Compliance Articulated Robot Arm)

- ✓ Alcance horizontal profundo
- ✓ Buen alcance en comparación con el tamaño
- ✓ Alta movilidad de posicionamiento

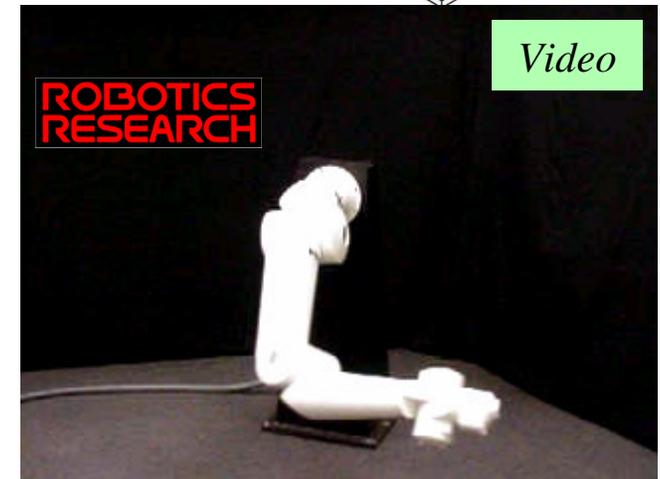
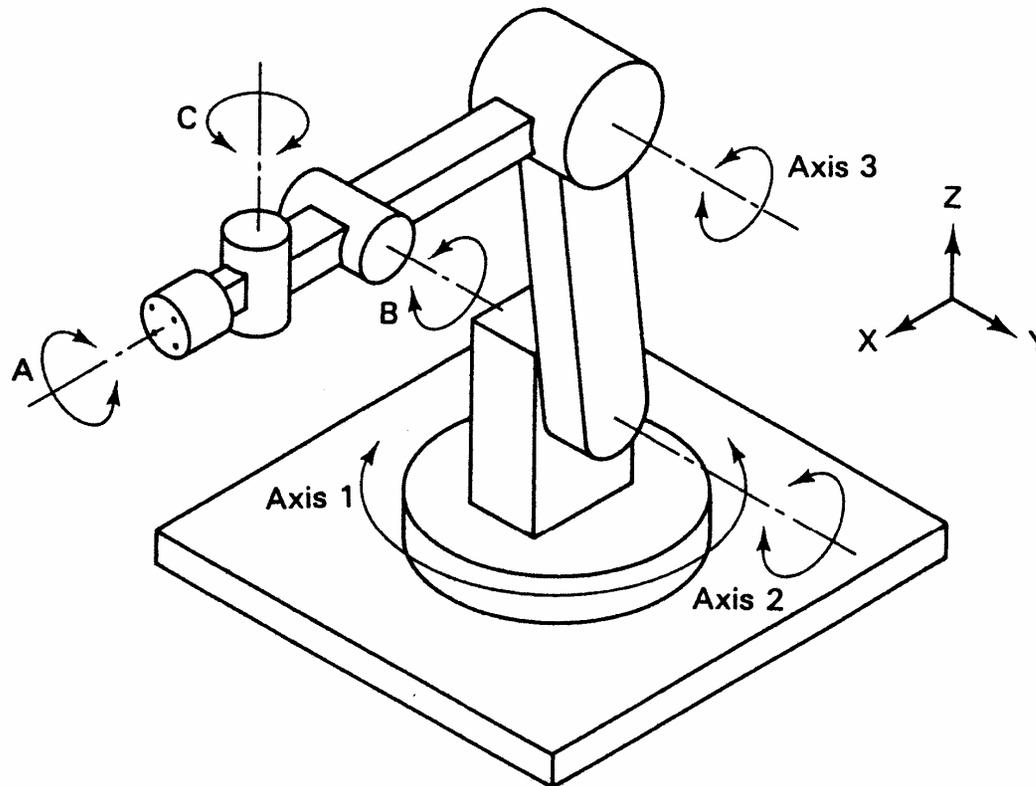
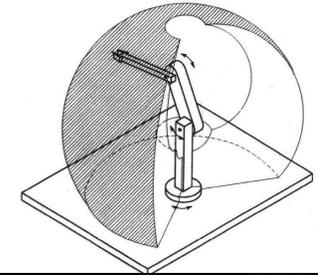
✗ Requiere control sofisticado



Elementos de un robot industrial: Estructura mecánica

- **Configuraciones**

- **Robot Verticalmente Articulado: Antropomórfico**



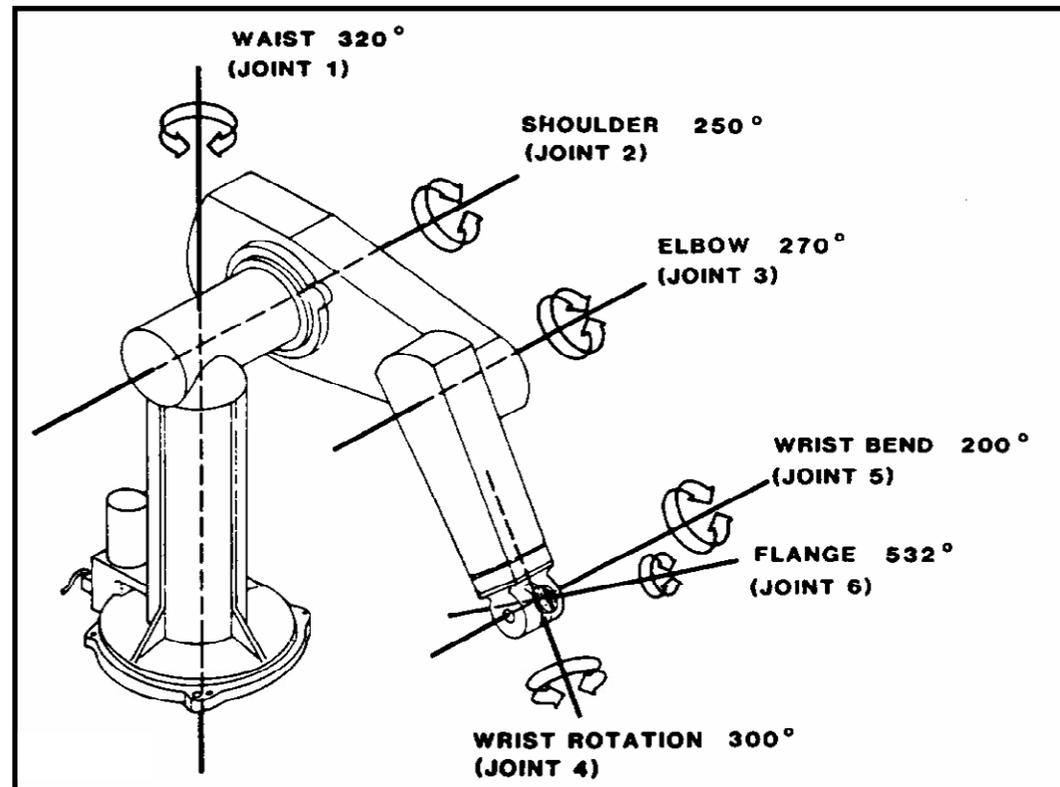
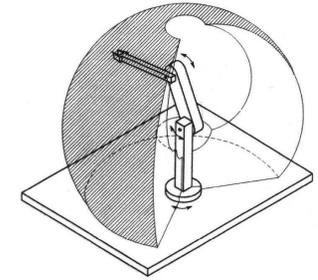
Robot K-1207i
7 GDL



Elementos de un robot industrial: Estructura mecánica

● Configuraciones

☛ Robot Verticalmente Articulado: Antropomórfico

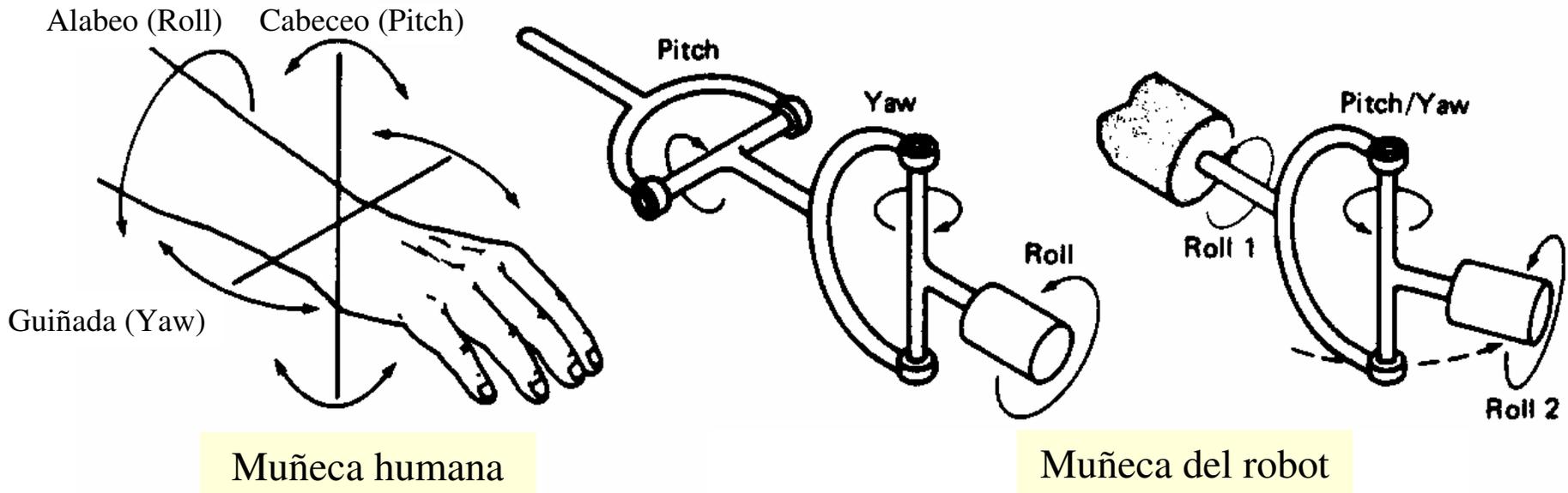




Elementos de un robot industrial: Estructura mecánica

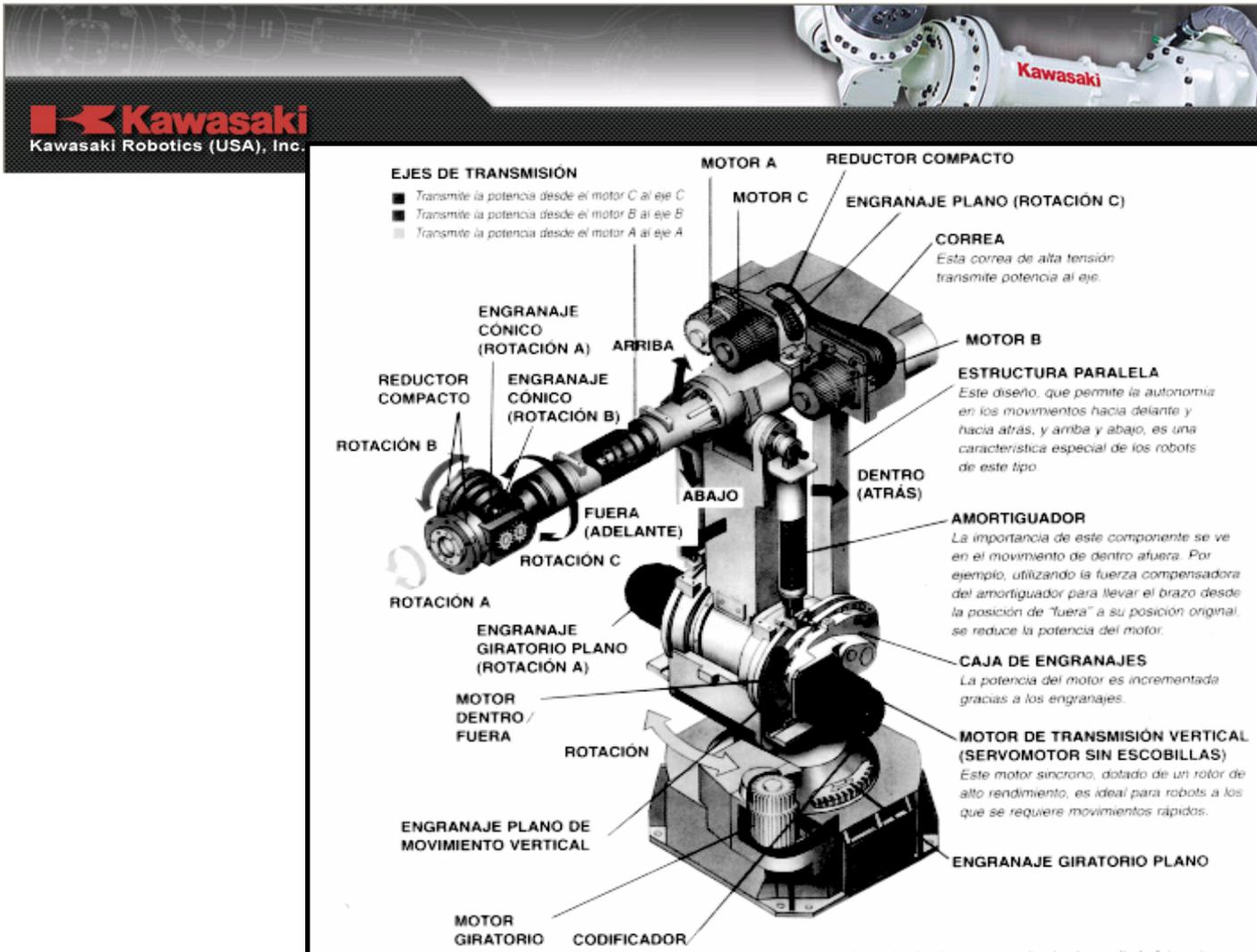
- Configuraciones

- La Muñeca





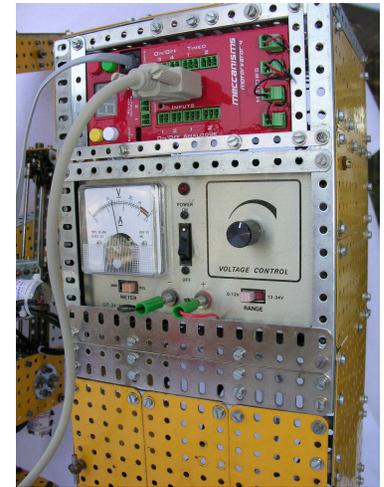
Elementos de un robot industrial: Estructura mecánica





Elementos de un robot industrial: Sistema de accionamiento

- ▶ Equipos electrónicos de potencia, encargados de **suministrar energía a los generadores de movimiento** (motores, electroválvulas hidráulicas o neumáticas).
- ▶ En la actualidad en su mayoría los robots utilizan **motores eléctricos de corriente continua**. Motores AC, comienzan a ser utilizados.
- ▶ Los primeros robots utilizaban accionamiento **hidráulico**, aún se utiliza este tipo de accionamiento cuando se requiere **gran cantidad de energía** (manipulación de grandes cargas).





Elementos de un robot industrial: Sistema de accionamiento

● Actuadores

- ▶ Los actuadores tienen por misión generar el movimiento de los elementos del robot según las órdenes dadas por la unidad de control.
- ▶ Los actuadores utilizados en robótica pueden emplear energía **neumática, hidráulica o eléctrica**.
- ▶ Cada uno de estos sistemas presenta características diferentes tales como:
 - Potencia
 - Controlabilidad
 - Peso y volumen
 - Precisión
 - Velocidad
 - Mantenimiento
 - Coste





Elementos de un robot industrial: Sistema de accionamiento

● **Actuadores Eléctricos**

▶ Por sus características de control, precisión y sencillez son los más utilizados en los robots industriales. Se disponen de tres tipos de actuadores eléctricos

☞ **Motores de corriente continua (DC)**

- Controlados por inducido
- Controlados por excitación



☞ **Motores de corriente alterna (AC)**

- Síncronos
- Asíncronos



☞ **Motores paso a paso**

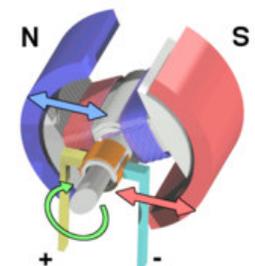
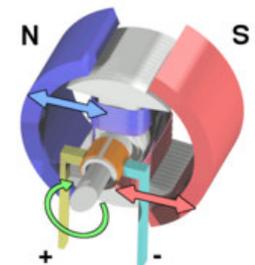
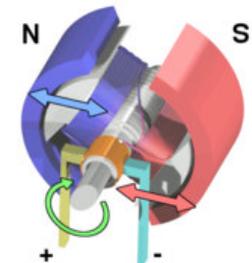




Elementos de un robot industrial: Sistema de accionamiento

● Actuadores Eléctricos: Motores de corriente continua (DC)

- ▶ Dos devanados internos inductor e inducido.
- ▶ El inductor, situado en el estator, crea un campo de excitación de dirección fija.
- ▶ El inducido, situado en el rotor, lo hace girar debido a la fuerza de Lorentz debida al la corriente circulante por él y el campo magnético de excitación.
- ▶ En control por inducido la tensión del inductor se mantiene constante, utilizando la tensión del inducido para controlar la velocidad de giro.
- ▶ Velocidades de **1000 – 3000 r.p.m.**



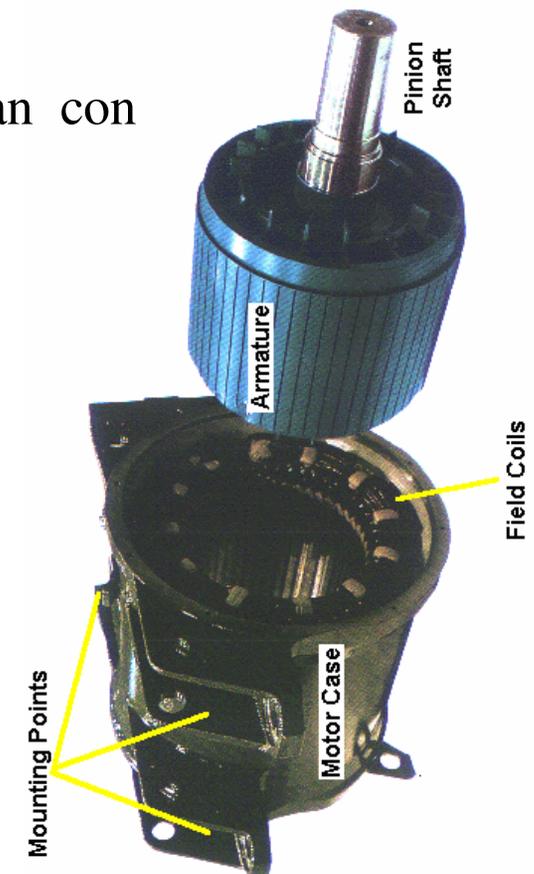


Elementos de un robot industrial: Sistema de accionamiento

- **Actuadores Eléctricos: Motores de corriente alterna (AC)**



- ▶ El inductor situado en el rotor está constituido por imanes permanentes.
- ▶ El inducido ubicado en el estator está formado por tres devanados iguales desfasadas 120° eléctricos y se alimentan con un sistema trifásico de tensiones.
- ▶ El inducido, situado en el rotor, lo hace girar debido a la fuerza de Lorentz debida al la corriente circulante por él y el campo magnético de excitación.
- ▶ El control de la velocidad se realiza mediante convertidores de frecuencia.
- ▶ El motor síncrono autopilotado excitado con imán permanente (motor senoidal) no requiere mantenimiento puesto que no posee escobillas.

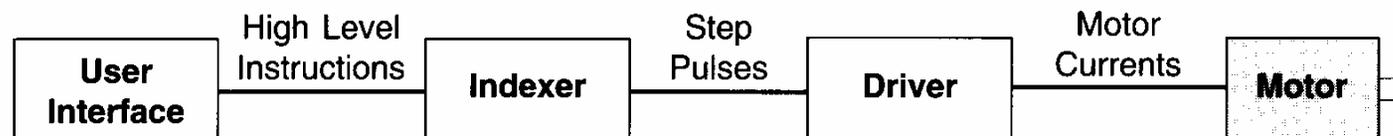
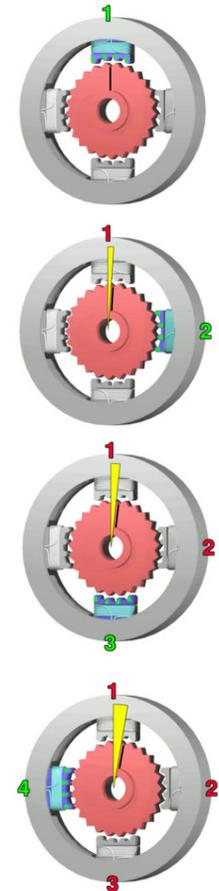




Elementos de un robot industrial: Sistema de accionamiento

● Actuadores Eléctricos: Motores paso a paso

- ▶ La señal de control son trenes de pulsos que van actuado rotativan sobre una serie de imanes dispuestos en el estator.
- ▶ Para lograr el giro del rotor en un determinado número de grados, las bobinas del estator deben ser excitadas **secuencialmente** en una frecuencia que determina la velocidad de giro.
- ▶ Funcionamiento en bajas velocidades **no es suave**.
- ▶ Tienden a **sobrecalentarse**.

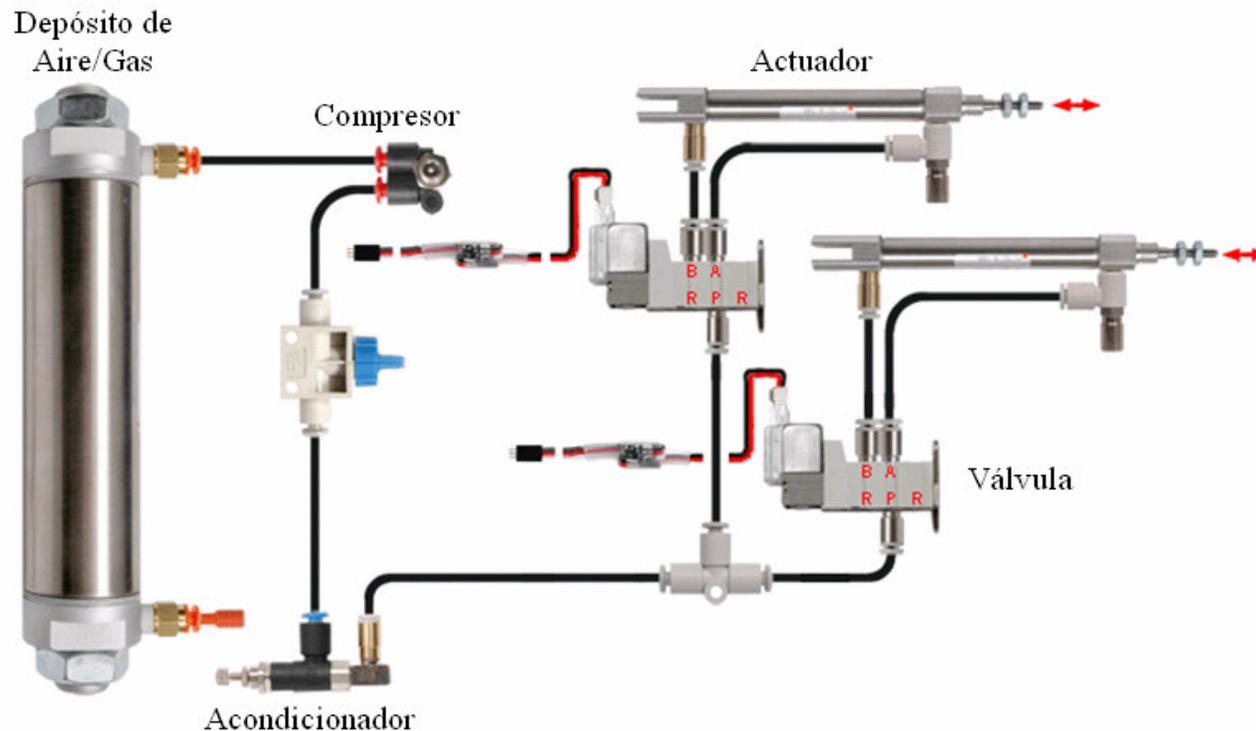




Elementos de un robot industrial: Sistema de accionamiento

● Actuadores Neumáticos

- ▶ La fuente de energía es aire a presión entre 5 y 10 bar.
- ▶ Pueden ser de dos tipos: Cilindros Neumáticos y Motores Neumáticos (De aletas rotativas).





Elementos de un robot industrial: Sistema de accionamiento

● Actuadores Hidráulicos

- ▶ Utilizan aceites minerales a una presión comprendida normalmente entre los 50 y los 100 bar, llegando en ocasiones hasta los 300 bar.
- ▶ Poseen mayor precisión en comparación con los neumáticos.
- ▶ Este tipo de accionamientos suelen utilizarse con frecuencia en robots que deban manejar grandes cargas.
- ▶ Los robots UNIMATE 2000 y 4000 utilizan accionamiento hidráulico con capacidades de carga de 70 y 205 kg. respectivamente.





Elementos de un robot industrial: Sistema de accionamiento

	Neumático	Hidráulico	Eléctrico
Energía	Aire a (5-10 bar)	• Aceite mineral (50-100 bar)	• Corriente eléctrica
Opciones	<ul style="list-style-type: none"> • Cilindros • Motor de Paletas • Motor de Pistón 	<ul style="list-style-type: none"> • Cilindros • Motor de Paletas • Motor de Pistón 	<ul style="list-style-type: none"> • Corriente continua • Corriente alterna • Motor paso a paso
Ventajas	<ul style="list-style-type: none"> • Baratos • Rápidos • Sencillos • Robustos 	<ul style="list-style-type: none"> • Rápidos • Alta relación potencia peso. • Autolubricantes • Alta capacidad de Carga • Estabilidad ante cargas estáticas 	<ul style="list-style-type: none"> • Precisos • Fiables • Fácil control • Sencilla instalación • Silenciosos
Desventajas	<ul style="list-style-type: none"> • Dificultad de Control continuo • Instalación especial • Ruidoso 	<ul style="list-style-type: none"> • Difícil mantenimiento • Instalación especial • Frecuentes fugas • Costosos. 	<ul style="list-style-type: none"> • Potencia limitada.



Elementos de un robot industrial: Sistema de transmisiones

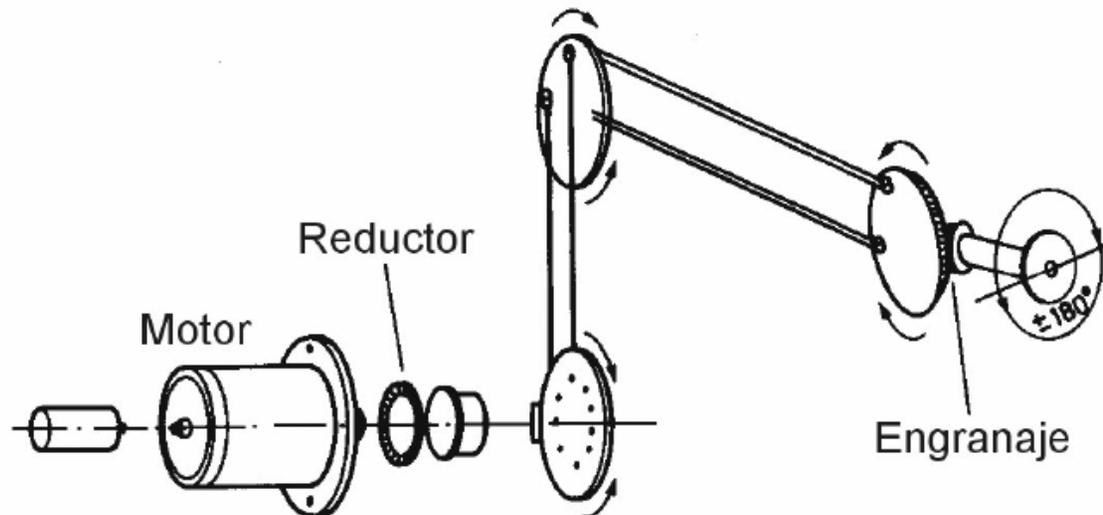
● Problema

▶ Transmitir los movimientos desde los accionadores hacia las articulaciones para disminuir las inercias.

● Sistemas de Transmisiones

▶ Encargados de transferir y canalizar el movimiento de los motores hasta los elementos móviles del robot.

▶ Constan de dos partes sistema de reductores y sistema de transmisión.





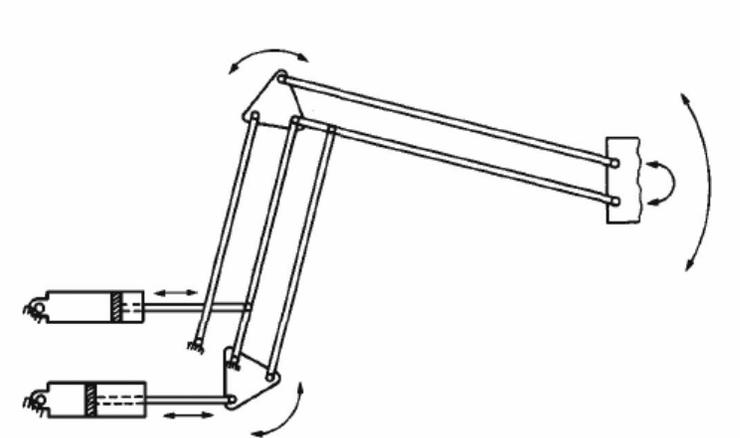
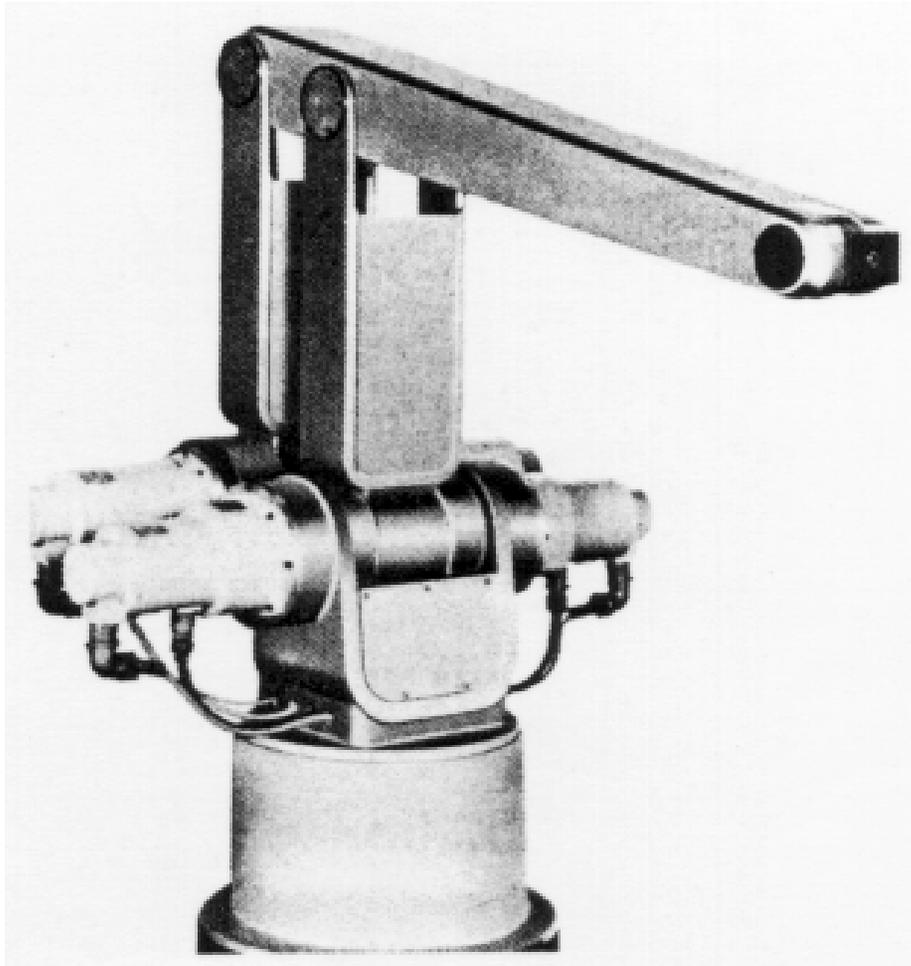
Elementos de un robot industrial: Sistema de transmisiones

- **Sistema de transmisión: Objetivos**
 - ▶ Reducir los pares estáticos.
 - ▶ Reducir los momentos de inercia (esp. en el extremo).
- **Método**
 - ▶ Trasladar el mayor peso posible cerca de la base del robot.
- **Función adicional**
 - ▶ Convertir movimiento circular en lineal o viceversa.
- **Características básicas**
 - ▶ Tamaño y peso reducido.
 - ▶ Juegos y holguras pequeños
 - ▶ Gran rendimiento
 - ▶ Escaso rozamiento
 - ▶ Mínimo desgaste, “maintenance free”
 - ▶ Capaz de soportar funcionamiento continuo a un par elevado.



Elementos de un robot industrial: Sistema de transmisiones

- Tipos de transmisiones: Paralelogramo

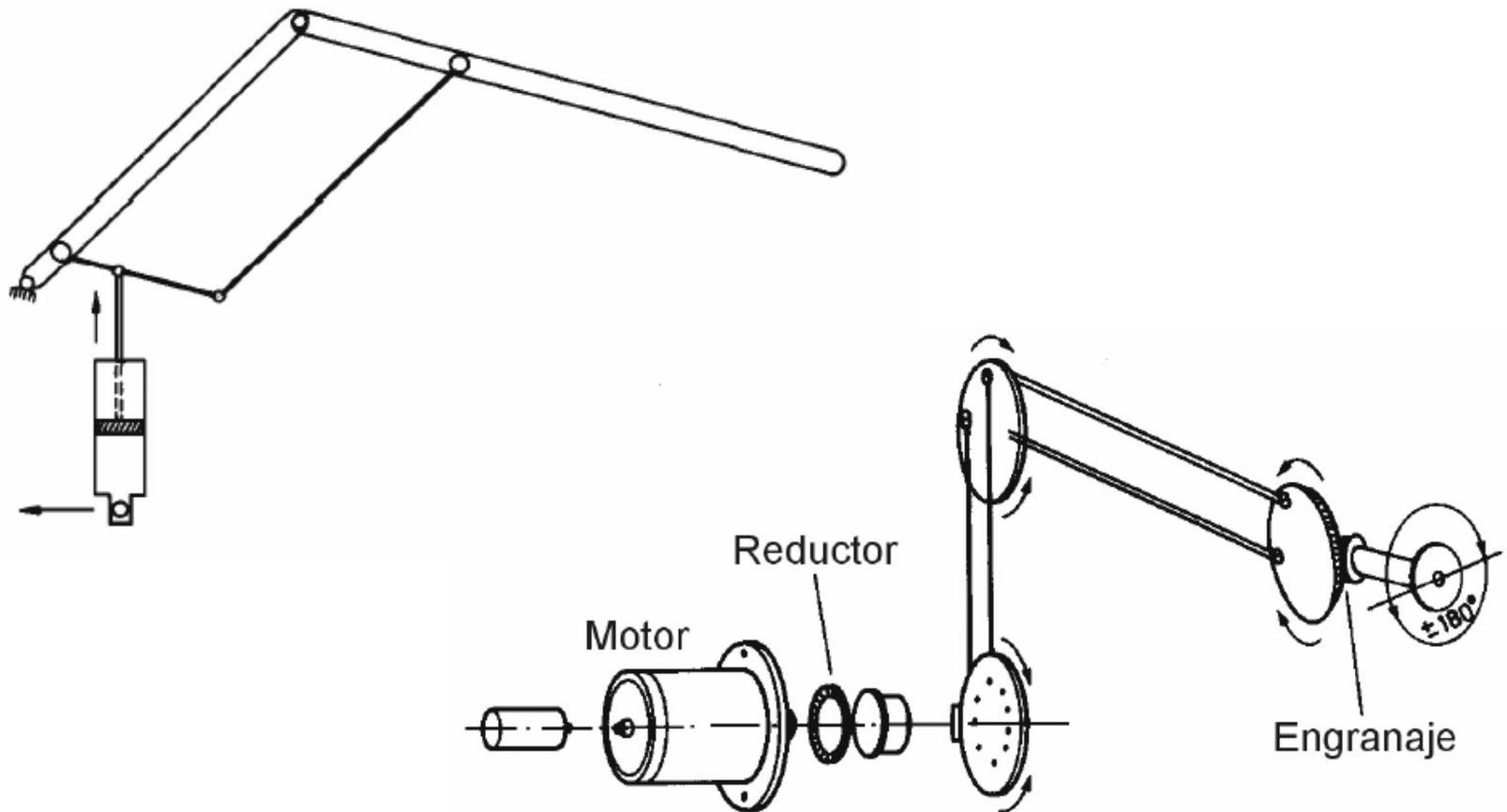


Paralelogramos articulados para
la conversión de movimiento
linear en circular



Elementos de un robot industrial: Sistema de transmisiones

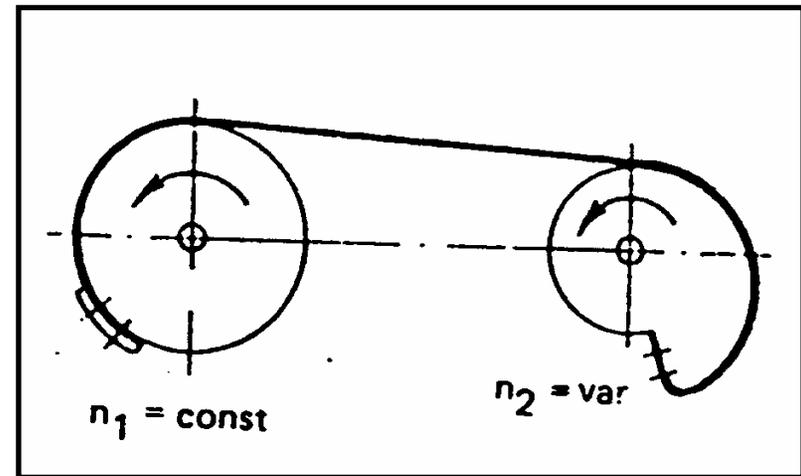
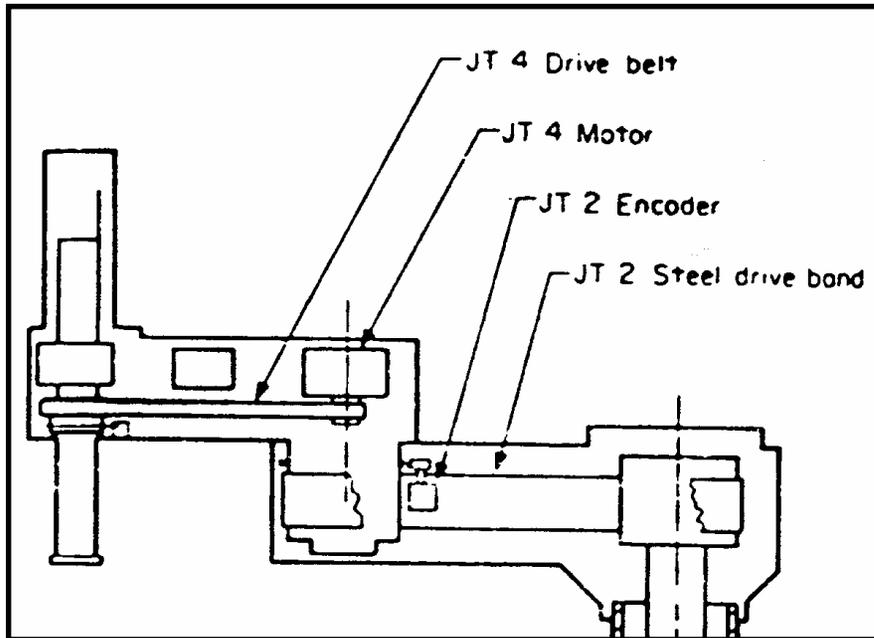
- Tipos de transmisiones: Tirantes





Elementos de un robot industrial: Sistema de transmisiones

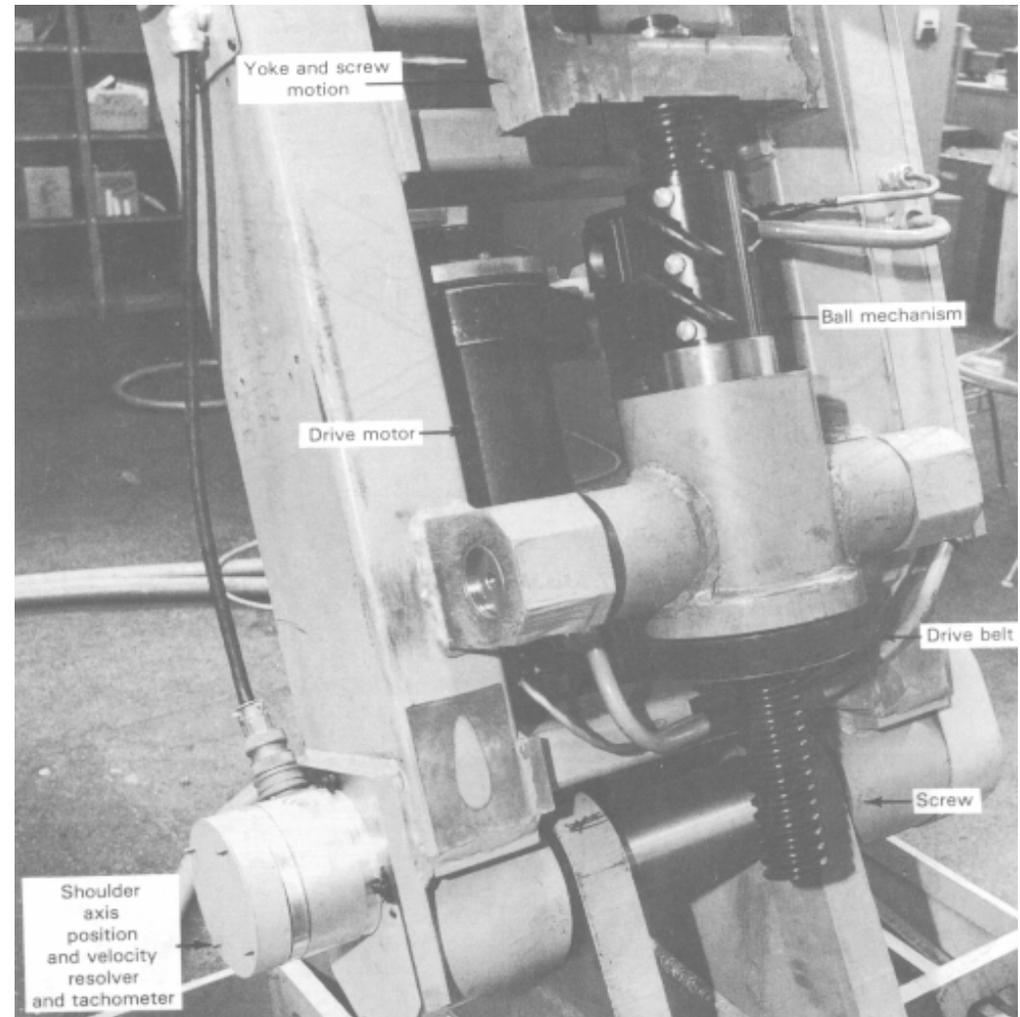
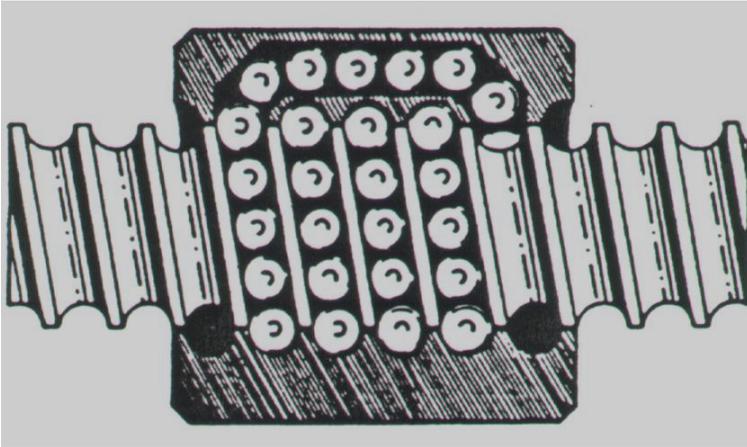
- Tipos de transmisiones: Correas





Elementos de un robot industrial: Sistema de transmisiones

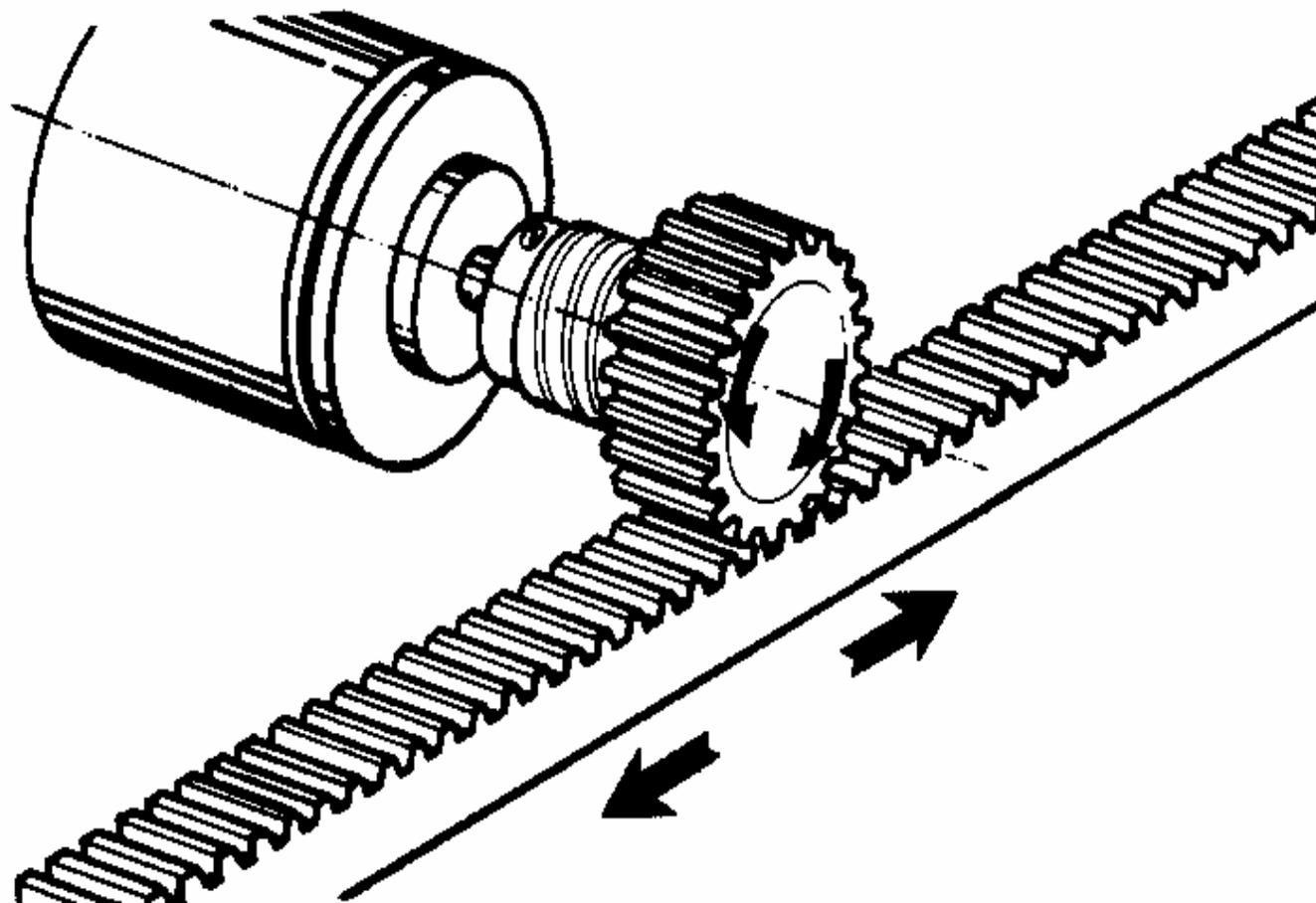
- Tipos de transmisiones: Husillo





Elementos de un robot industrial: Sistema de transmisiones

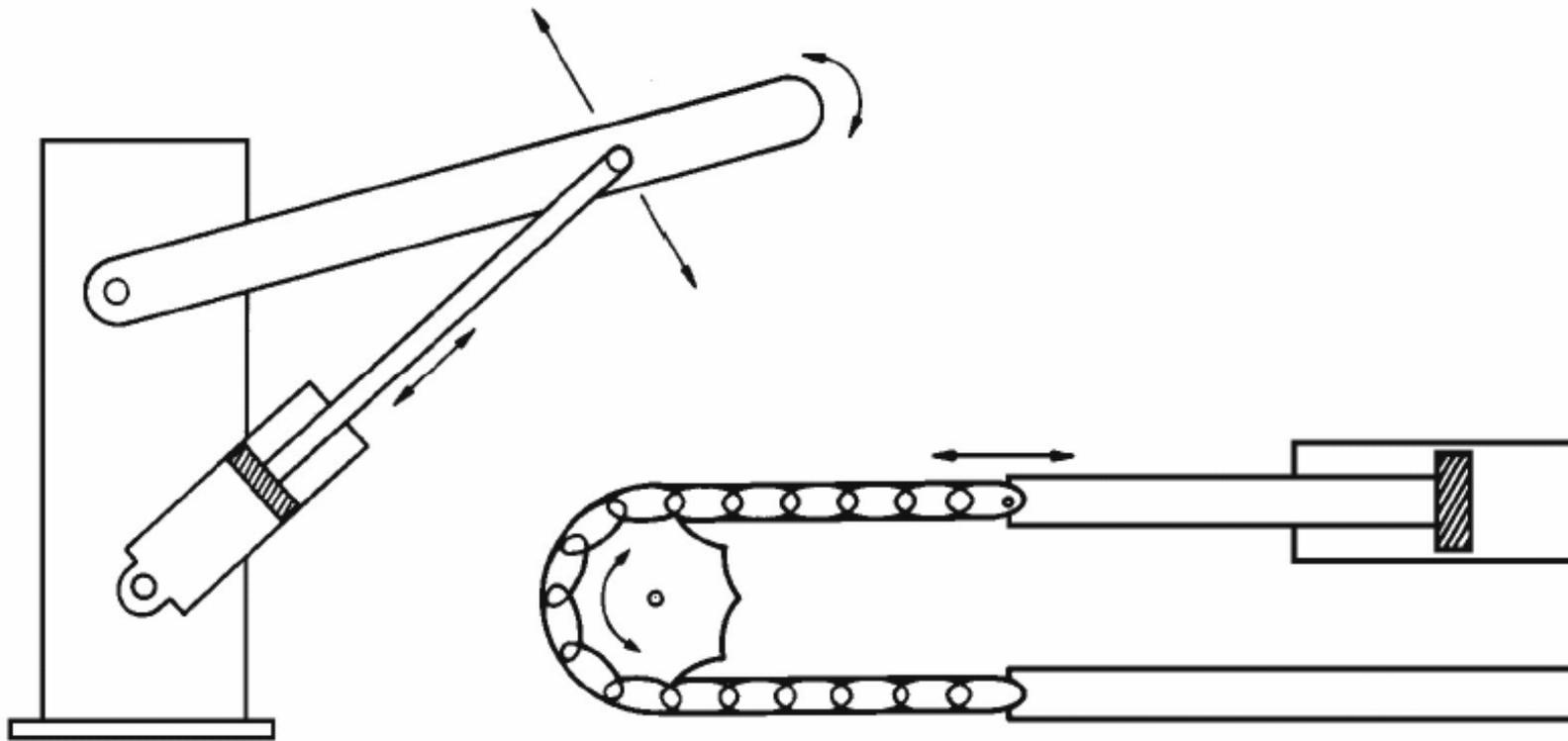
- Tipos de transmisiones: piñón-cremallera





Elementos de un robot industrial: Sistema de transmisiones

- Tipos de transmisiones: Mecanismo articulado



Mecanismos para efectuar un movimiento circular mediante un actuador de desplazamiento lineal



Elementos de un robot industrial: Sistema de transmisiones

- **Tipos de transmisiones**

Entrada-Salida	Denominación	Ventajas	Inconvenientes
Circular - Circular	Engranaje	Pares altos	Holguras
	Correa Dentada	Distancia grande	Desgaste
	Cadena	Distancia grande	Ruido
	Paralelogramo	Pares altos	Giro limitado
	Cable	Poco peso	Deformidad
Circular - Lineal	Tornillo sin fin	Poca holgura	Rozamiento
	Cremallera	Holgura media	Rozamiento
Lineal - Circular	Paral. articulado	Inercia baja	Control difícil
	Cremallera	Holgura media	Rozamiento



Elementos de un robot industrial: Sistema de transmisiones

● Reductores: Objetivos

- ▶ Los reductores disminuyen las altas velocidades de giro de los ejes de los motores aumentando el par.
- ▶ Aumentar el par motor

$$P_{art} = \eta P_{motor} N$$

P_{art} : Par de la articulación (salida)
 P_{motor} : Par del motor (entrada)
 η : Rendimiento

$$N = \omega_{motor} / \omega_{art}$$

ω_{eje} : Velocidad de la articulación (salida)
 ω_{motor} : Velocidad del motor (entrada)



Elementos de un robot industrial: Sistema de transmisiones

● Reductores: Características

Valores típicos

Relación de reducción	50-300
Peso y tamaño	0.1-30 Kg.
Momento de inercia	10^{-4} Kgm ²
Velocidades de entrada máxima	6000 - 7000 r.p.m.
Par de salida nominal	5700 Nm
Par de salida máximo	7900 Nm
Juego angular*	0 - 2''
Rigidez torsional**	100 - 2000 Nm/rad
Rendimiento	85 % ÷ 98 %

* Juego angular (*backlash*) es el ángulo que gira el eje de salida (articulación) cuando se cambia su sentido de giro sin que llegue a girar el eje de entrada (motor)

** Rigidez torsional es el par que hay que aplicar sobre el eje de salida para que, manteniendo bloqueado el de entrada, aqué gire un ángulo unitario.

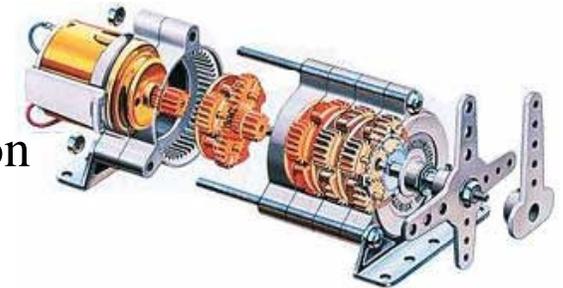


Elementos de un robot industrial: Sistema de transmisiones

● Reductores: Tipos

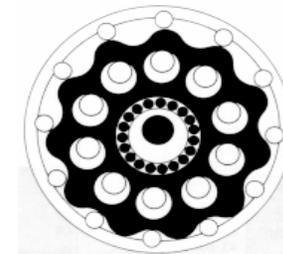
☛ Planetarios

- ✓ Bajo coste, gran variedad, alto par de transmisión
- ✗ Alta inercia, gran peso, grandes juegos



☛ Cyclo

- ✓ Media inercia, pequeño peso, medios juegos
- ✗ Coste medio, bajo par de transmisión



☛ Harmonic Drive (HD)

- ✓ Baja inercia, pequeño peso, pequeños juegos
- ✗ Alto coste, no excesivamente alto par de transmisión

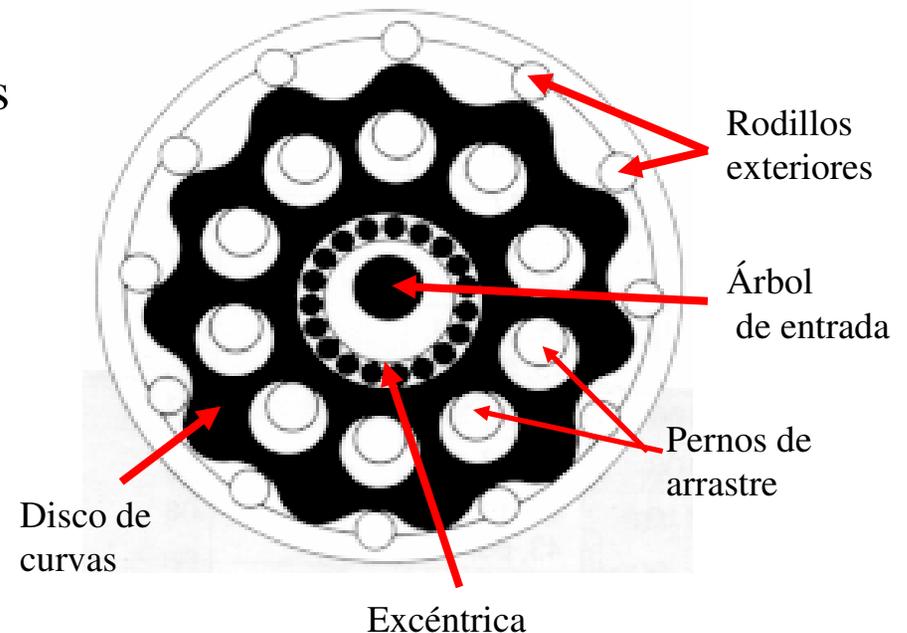




Elementos de un robot industrial: Sistema de transmisiones

● Reductores: Reductor Cyclo

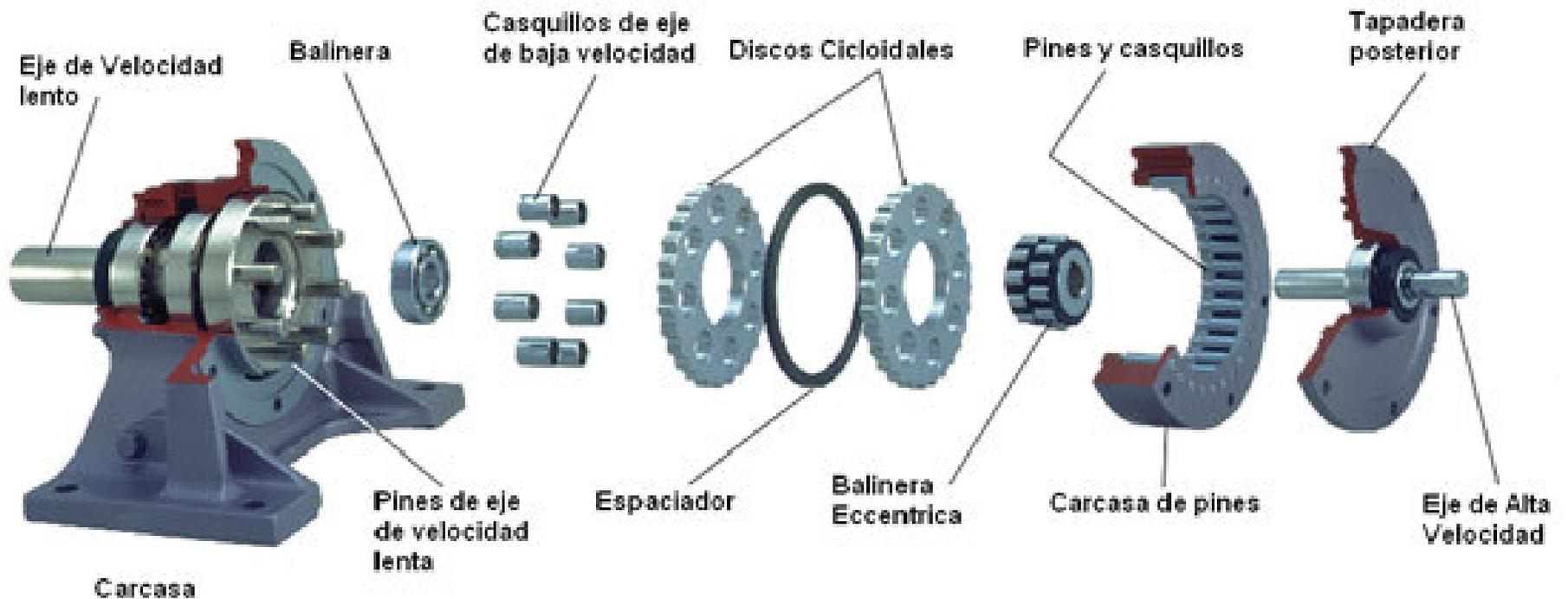
- ▶ Se basa en el **movimiento cicloidal de un disco de curvas** movido por una excéntrica solidaria al árbol de entrada.
- ▶ Por cada revolución de la excéntrica el disco de curvas avanza un saliente rodando sobre los rodillos exteriores.
- ▶ Este avance arrastra a su vez los pernos del árbol de salida.
- ▶ La relación de reducción está determinada por el **número de salientes**.





Elementos de un robot industrial: Sistema de transmisiones

- **Reductores: Reductor Cyclo**



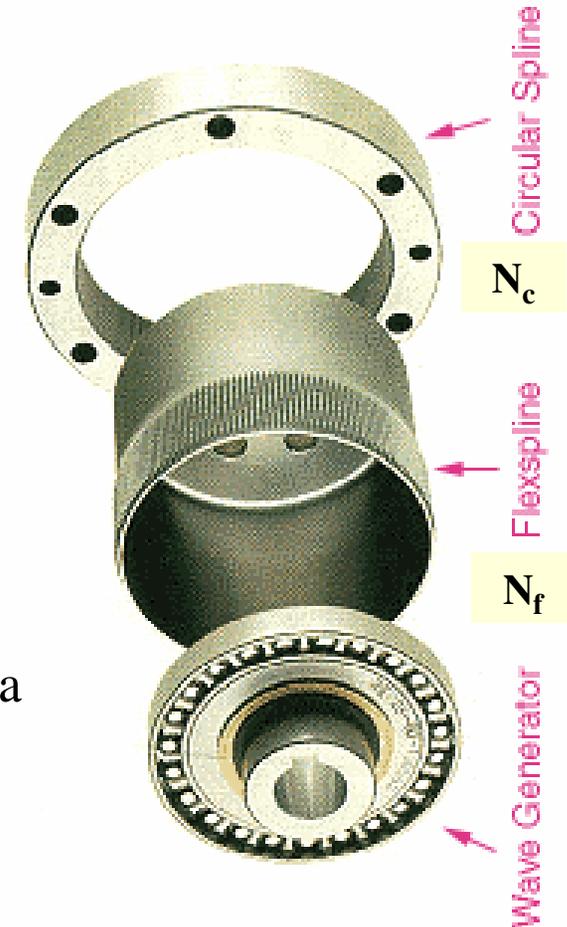
Reductor Cyclo 6000 - SUMITOMO



Elementos de un robot industrial: Sistema de transmisiones

● Reductores: Harmonic Drive (HDUC)

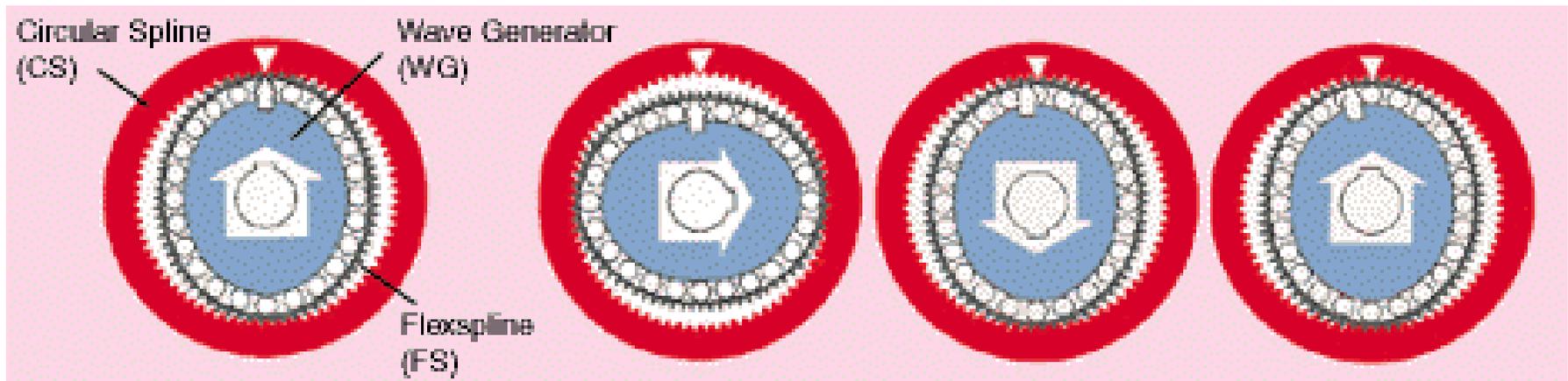
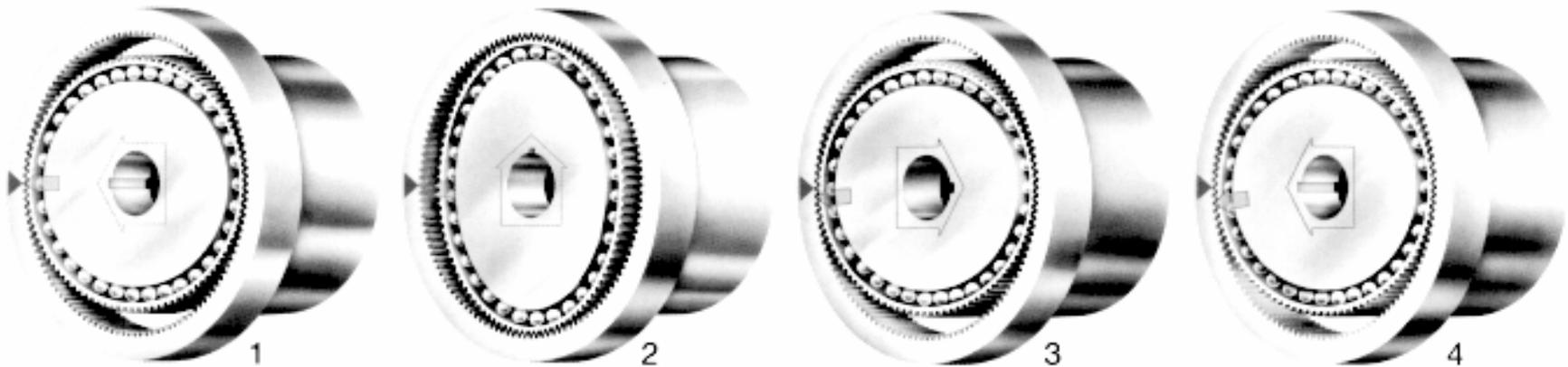
- ▶ Se basa en una corona exterior rígida con dentado interior (circular spline) y un vaso flexible (flexspline) con dentado exterior que engrana en el circular spline.
- ▶ Interior al vaso gira un rodamiento elipsoidal (wave generator) que deforma el vaso, poniendo en contacto la corona exterior con la zona del vaso correspondiente al máximo diámetro de la elipse.
- ▶ Al haber una diferencia de dientes $N_c - N_f$, tras una vuelta del wave generator, el flexible spline solo avanza Z dientes. Relación $N_c / (N_c - N_f)$.
- ▶ Se logran reducciones de hasta **320** y capacidad de transmisión de **5720 Nm**.





Elementos de un robot industrial: Sistema de transmisiones

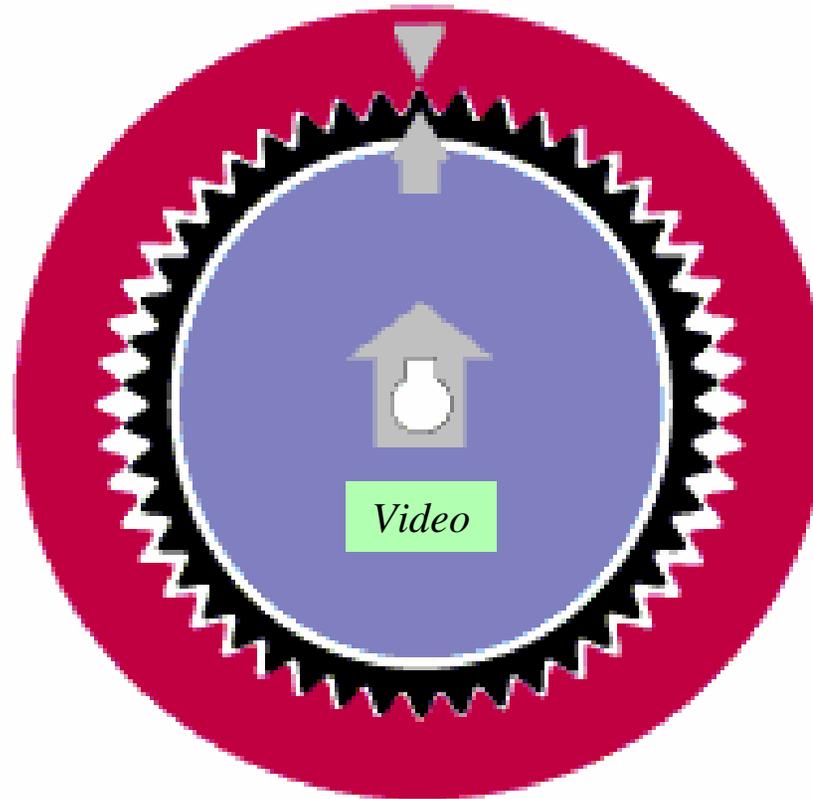
- **Reductores: Harmonic Drive (HDUC)**





Elementos de un robot industrial: Sistema de transmisiones

- **Reductores: Harmonic Drive (HDUC)**

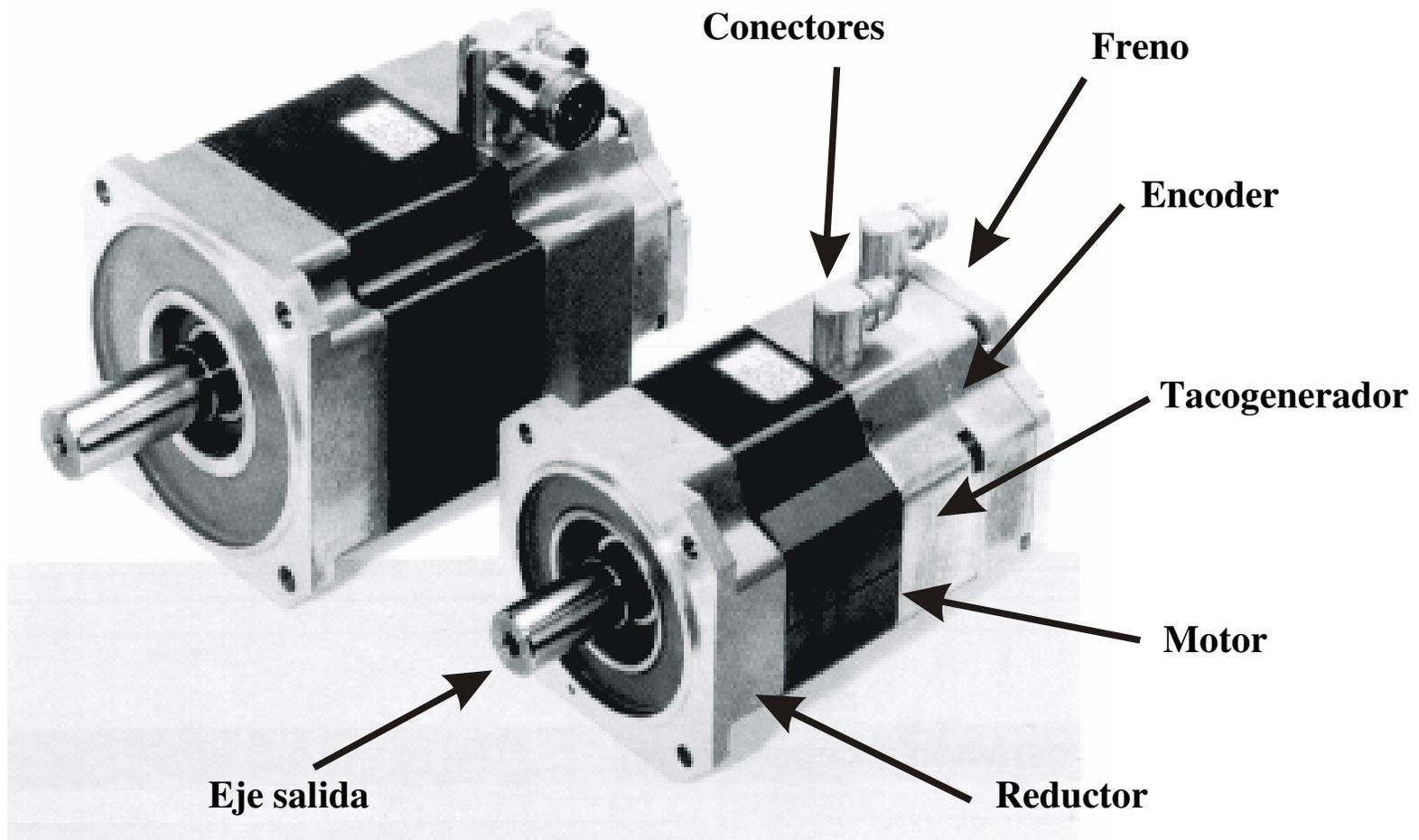


Harmonic Drive



Elementos de un robot industrial

- **Sistemas de Accionamiento: Conjunto Accionador**





Elementos de un robot industrial

- **Sistemas de Accionamiento directo:**

- ▶ El eje del actuador se conecta directamente a la carga o articulación sin la utilización de un reductor intermedio.
- ▶ Aplicaciones que combinan gran precisión y alta velocidad.
- ▶ Los motores deben proporcionar par elevado a bajas revoluciones.
- ▶ Utilizan motores síncronos de continua sin escobillas *brushless*.
- ▶ Posicionamiento rápido y preciso.
- ▶ Aumento de las posibilidades de controlabilidad.
- ▶ Simplificación del sistema mecánico.



AdeptOne-XL



Elementos de un robot industrial: Sistema Sensorial



- **Sensores Internos**

- ▶ Para conseguir que un robot realice su tarea con la adecuada precisión, velocidad e inteligencia, será preciso que tenga conocimiento tanto de su propio estado como del estado de su entorno.
- ▶ La información relacionada con su estado (fundamentalmente la posición de sus articulaciones) la consigue con sus sensores internos.
- ▶ La información que la unidad de control del robot puede obtener sobre el estado de su estructura mecánica es fundamentalmente la relativa a su posición y velocidad.



Elementos de un robot industrial: Sistema Sensorial



● **Sensores Internos**

Presencia

- Inductivo
- Capacitivo
- Efecto Hall
- Célula Reed
- Óptico
- Ultrasonido
- Contacto

Velocidad

- Tacogeneratriz

Posición

• Analógicos

- Potenciómetro
- Resolver
- Sincro
- Inductosyn
- LVDT

• Digitales

- Encoders absolutos
- Encoders incrementales
- Regla óptica



Elementos de un robot industrial: Sistema Sensorial

- **Sensores Externos: Sensores de presencia**

- ▶ Este tipo de sensor es capaz de detectar la presencia de un objeto dentro de un radio de acción determinado. Esta detección se puede realizar con o sin contacto con el objeto.
- ▶ Los detectores de presencia en robótica se utilizan principalmente como auxiliares de los detectores de posición, para indicar los límites de movimiento de las articulaciones y permitir localizar la posición de referencia de cero de éstos en el caso de ser incrementales.



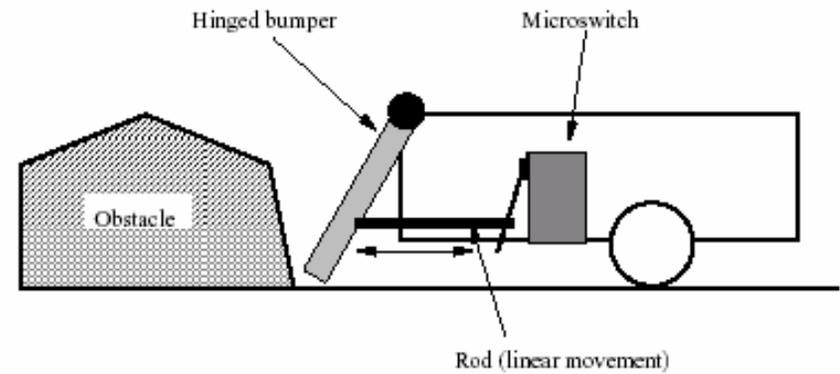
Elementos de un robot industrial: Sistema Sensorial

● Sensores Externos: Sensores de presencia - con contacto

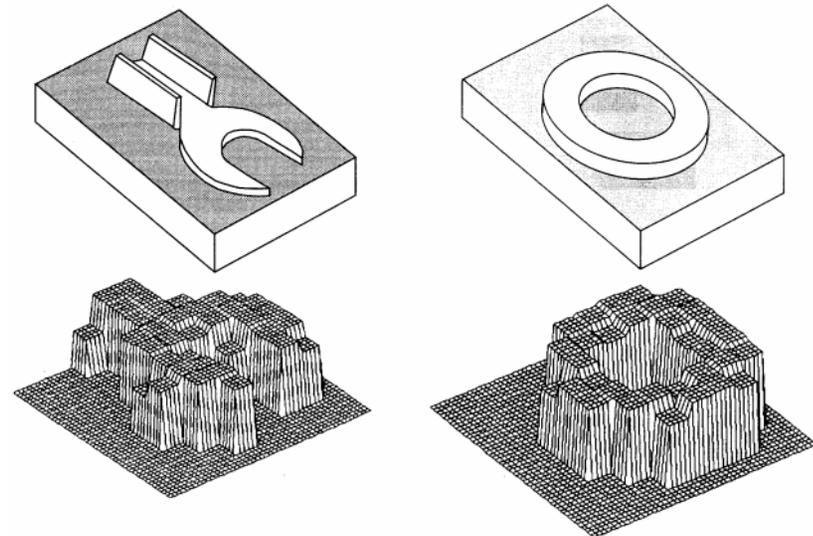
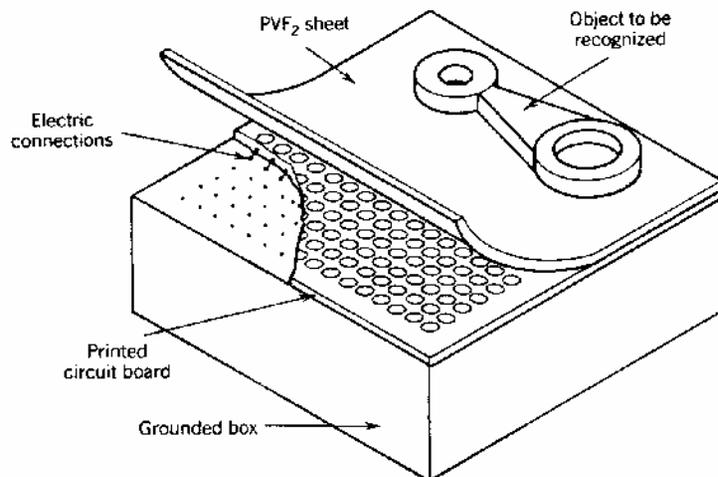
▶ Botones (interruptores)



▶ Finales de carrera



▶ Sensores táctiles



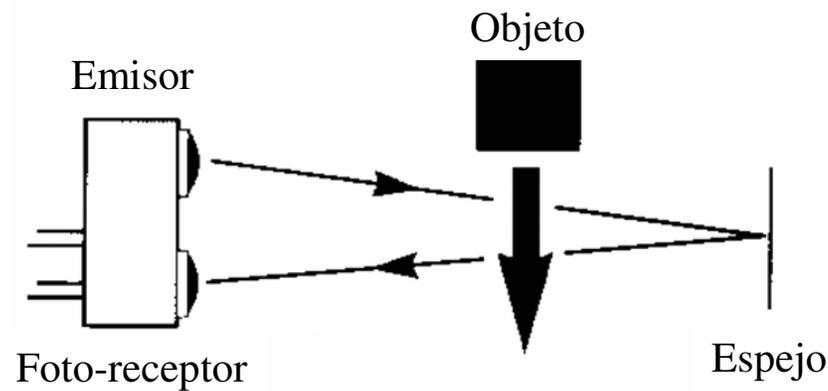


Elementos de un robot industrial: Sistema Sensorial

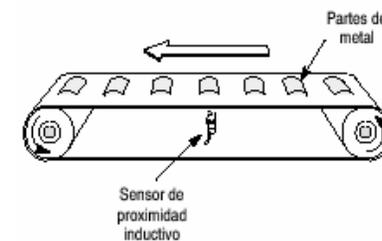
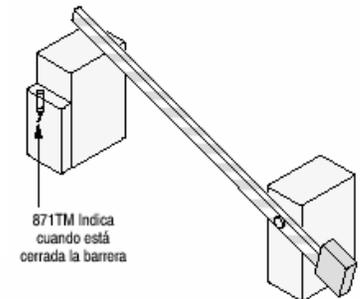
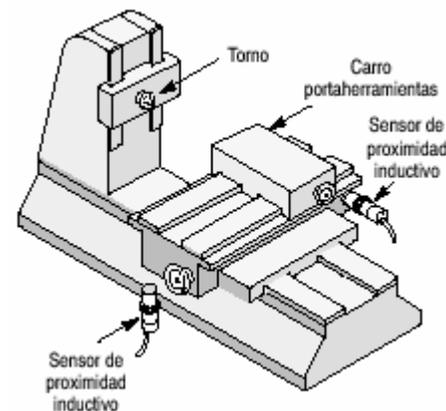
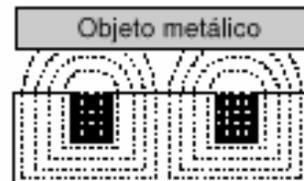
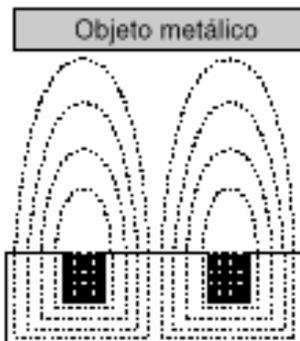
- **Sensores de presencia - sin contacto (Sensores de proximidad)**



- ▶ **Sensores ópticos**



- ▶ **Sensores Inductivos**



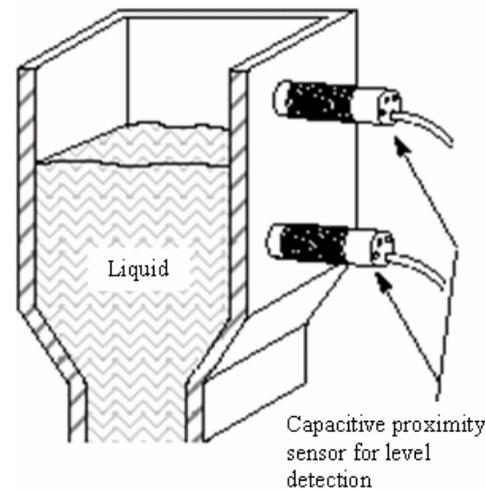
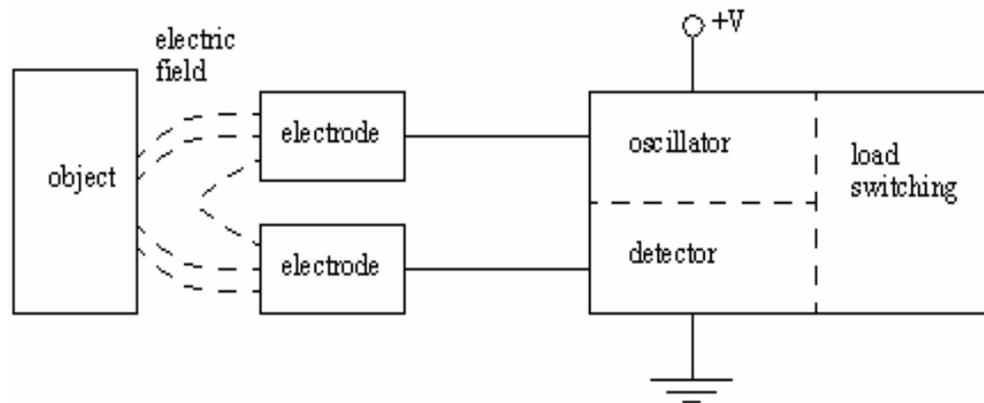


Elementos de un robot industrial: Sistema Sensorial



- **Sensores de presencia - sin contacto (Sensores de proximidad)**

- ▶ **Sensores Capacitivos**



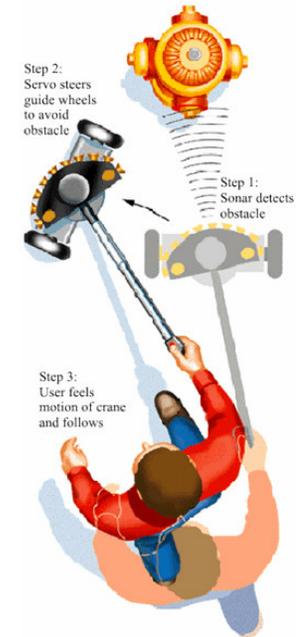
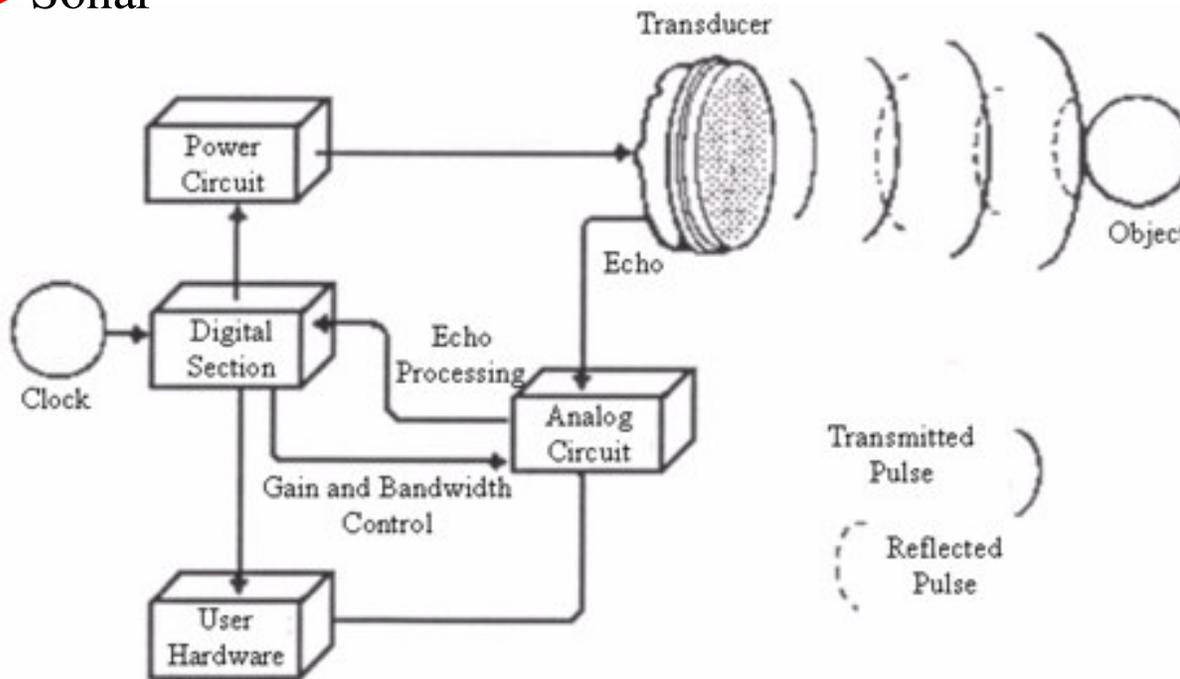


Elementos de un robot industrial: Sistema Sensorial

- **Sensores de presencia - sin contacto (Sensores de proximidad)**



- ▶ **Sonar**



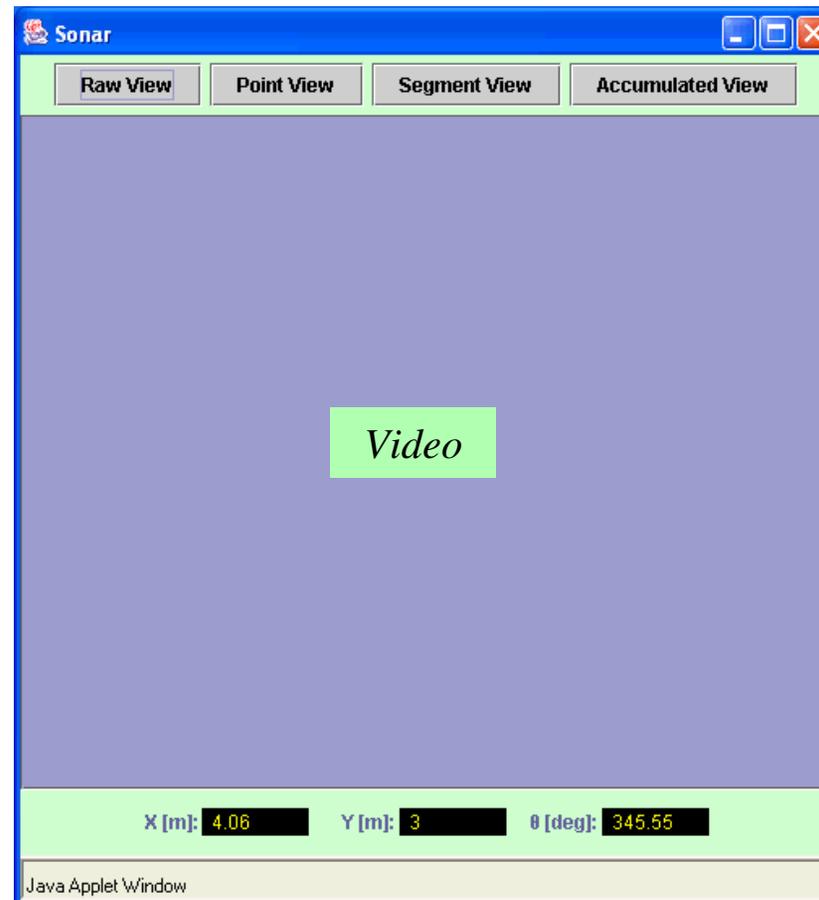


Elementos de un robot industrial: Sistema Sensorial

- **Sensores de presencia - sin contacto (Sensores de proximidad)**



- ▶ **Sonar**

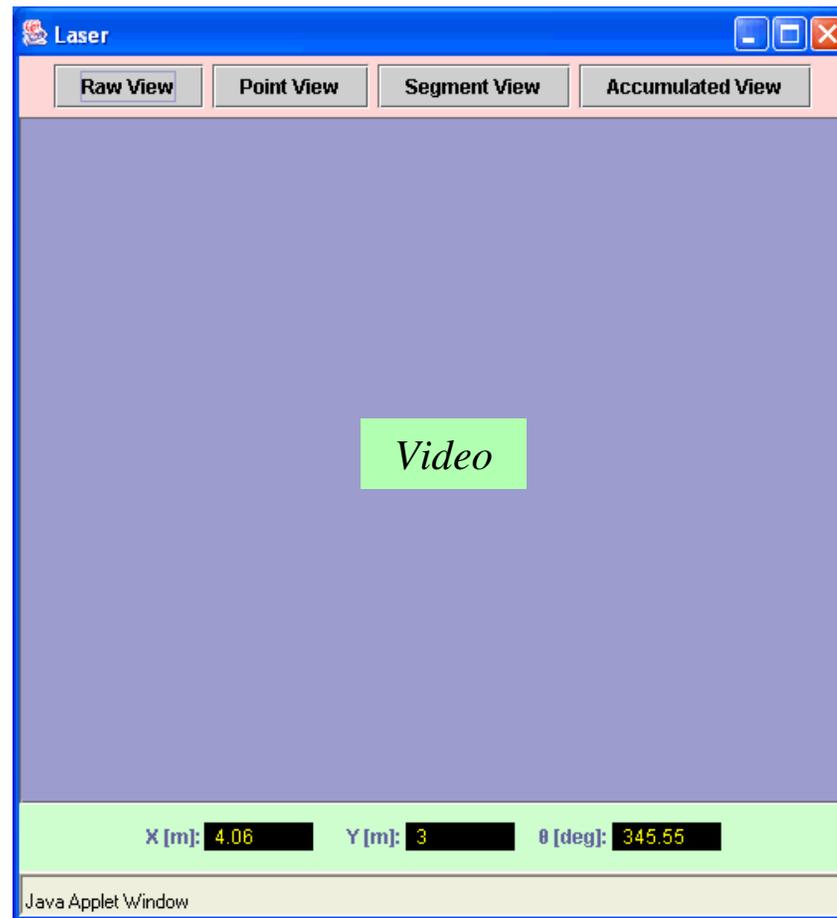
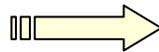




Elementos de un robot industrial: Sistema Sensorial

- **Sensores de presencia - sin contacto (Sensores de proximidad)**

- ▶ Láser





Elementos de un robot industrial: Sistema Sensorial

- **Sensores Internos: Sensores de posición**

- ▶ Para el control de posición angular se emplean fundamentalmente los denominados **encoders** y **resolvers**

- ▶ Los potenciómetros dan bajas prestaciones por lo que no se emplean salvo en contadas ocasiones (robots educacionales, ejes de poca importancia).

- **Sensores Internos: Codificadores angulares de posición (Encoders)**

- ▶ Los codificadores ópticos o **encoders** incrementales constan, en su forma más simple, de un disco transparente con una serie de marcas opacas colocadas radialmente y equidistantes entre si; de un sistema de iluminación (led emisor) y de un elemento foto-receptor.

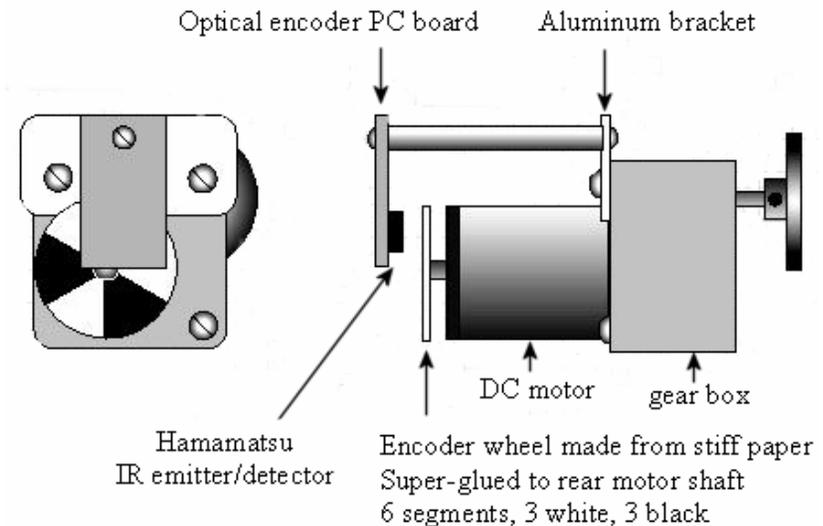
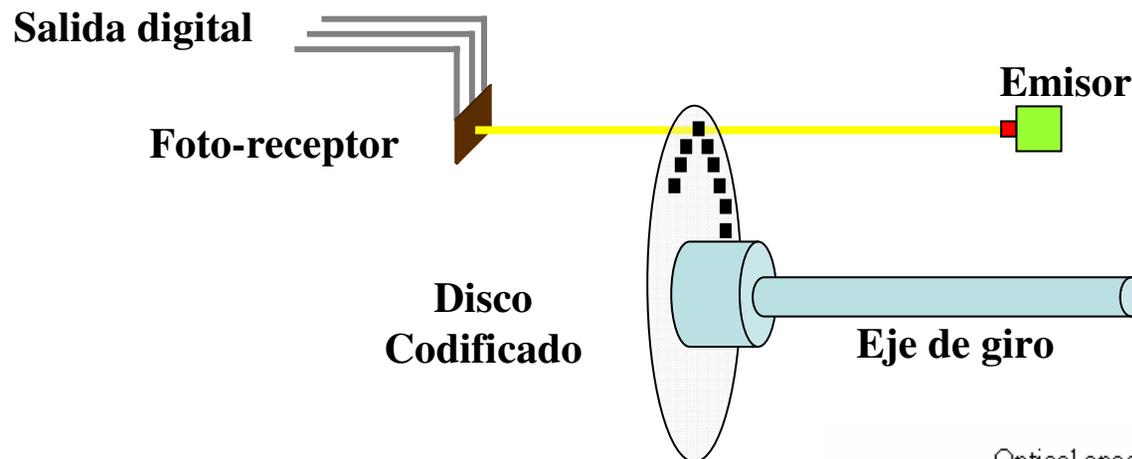
- ▶ El eje cuya posición se desea medir va acoplado al disco transparente.

- ▶ Suministran información digital.



Elementos de un robot industrial: Sistema Sensorial

- **Sensores Internos: Codificadores angulares de posición (Encoders)**



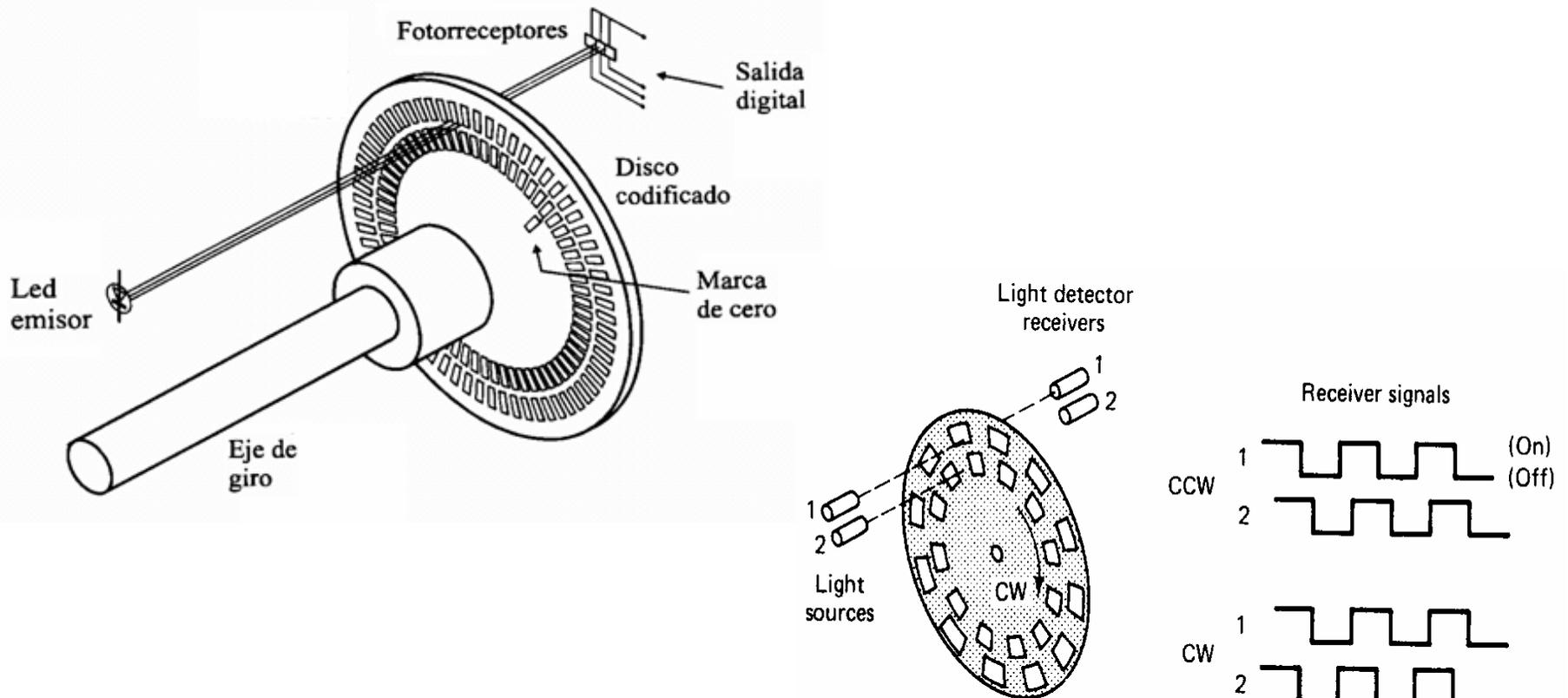


Elementos de un robot industrial: Sistema Sensorial

- **Sensores Internos: Codificadores angulares de posición (Encoders)**

- ▶ Tipos: Encoder Relativo, Encoder Absoluto y Regla Óptica

- ▶ Encoder Relativo:

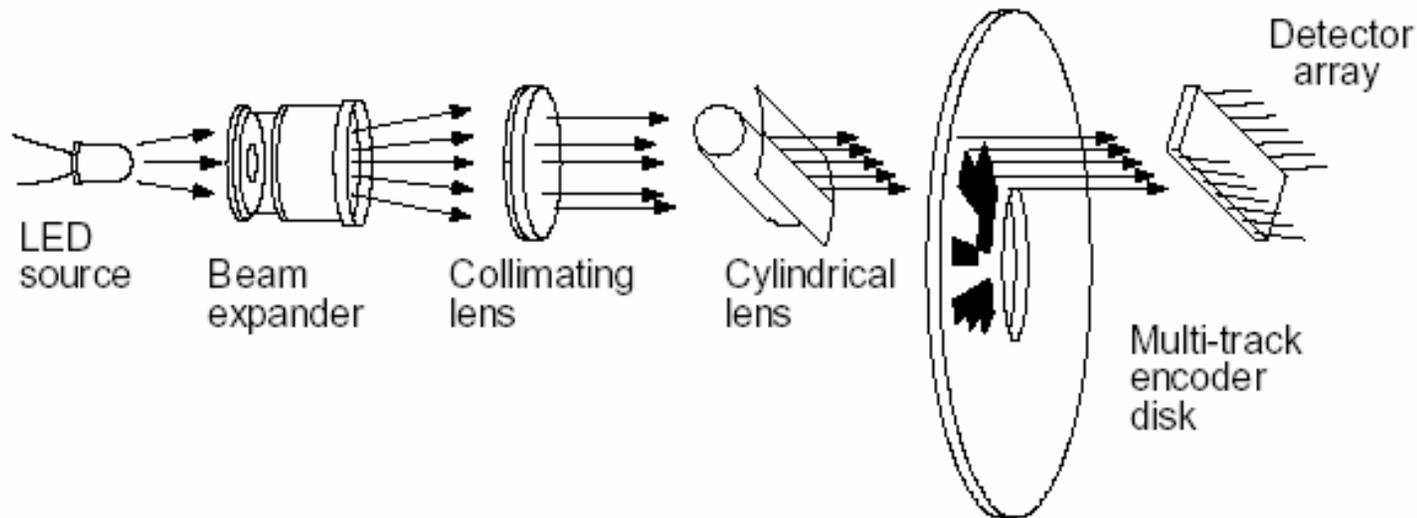




Elementos de un robot industrial: Sistema Sensorial

- **Sensores Internos: Codificadores angulares de posición (Encoders)**

- ▶ Encoder Absoluto:

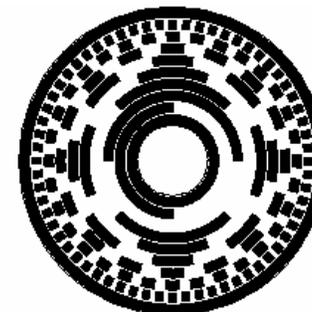


Código Gray

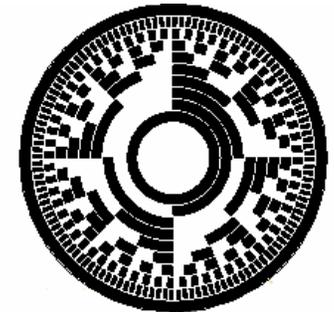
b_2	b_1	b_0
0	0	0
0	0	1
0	1	1
0	1	0
1	1	0
1	1	1
1	0	1
1	0	0

Código Binario

b_2	b_1	b_0
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1



Código Gray



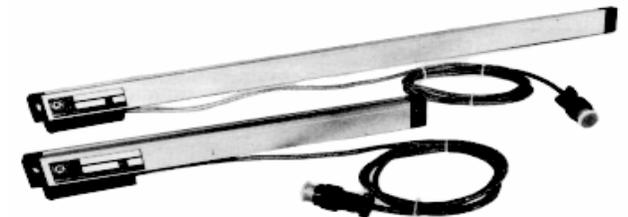
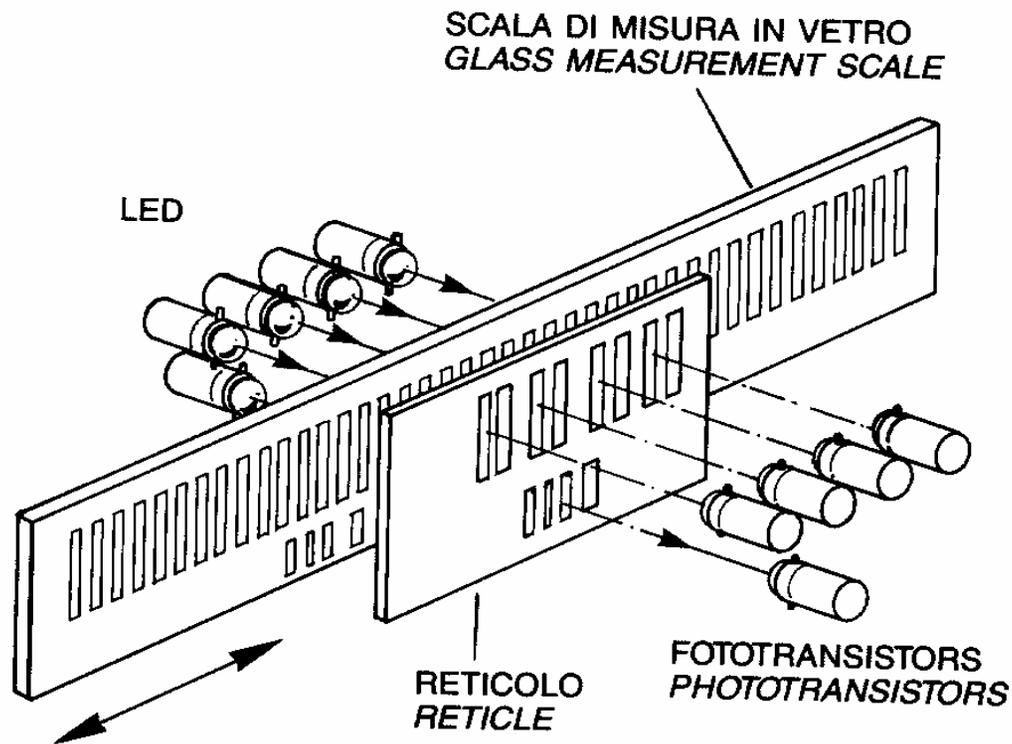
Código Binario



Elementos de un robot industrial: Sistema Sensorial

- **Sensores Internos: Codificadores lineales de posición (Encoders)**

- ▶ Regla óptica:





Elementos de un robot industrial: Sistema Sensorial

● Sensores Internos: Captadores angulares de posición (Resolvers)

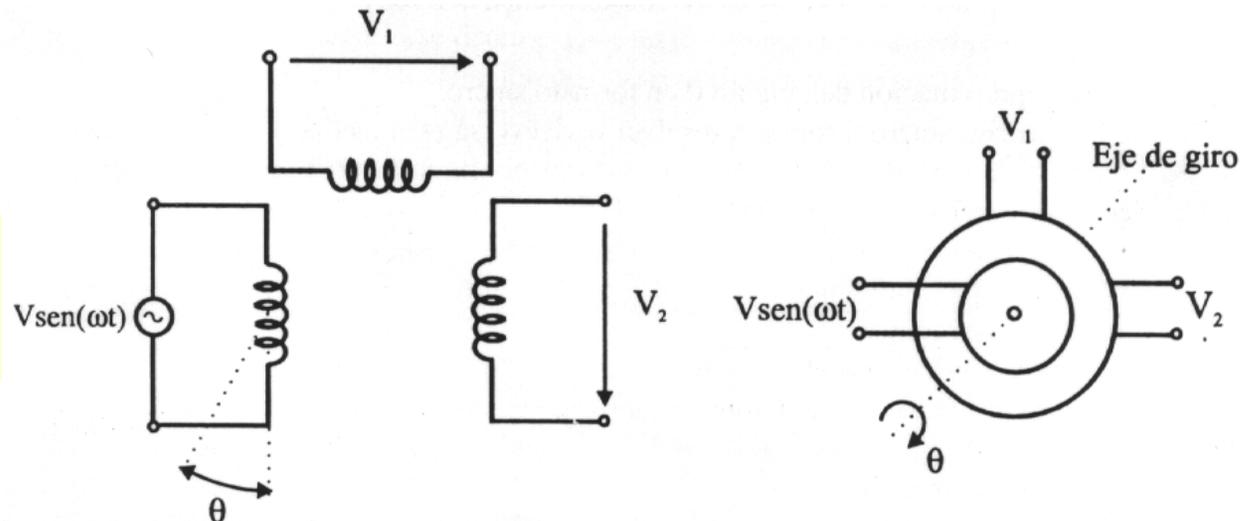
- ▶ Una bobina móvil conectada al eje de giro, y varias bobinas fijas.
- ▶ Al excitar la bobina móvil con una señal senoidal ($\cong 400$ Hz), en las bobinas fijas se inducen tensiones que dependen del ángulo girado.
- ▶ Tecnología analógica.
- ▶ Bajo momento de inercia.
- ▶ Resolución infinita.

👉 Resolver

Dos bobinas fijas desfasadas 90°

$$V_1 = V \sin(\omega t) \sin \theta$$

$$V_2 = V \sin(\omega t) \cos \theta$$





Elementos de un robot industrial: Sistema Sensorial

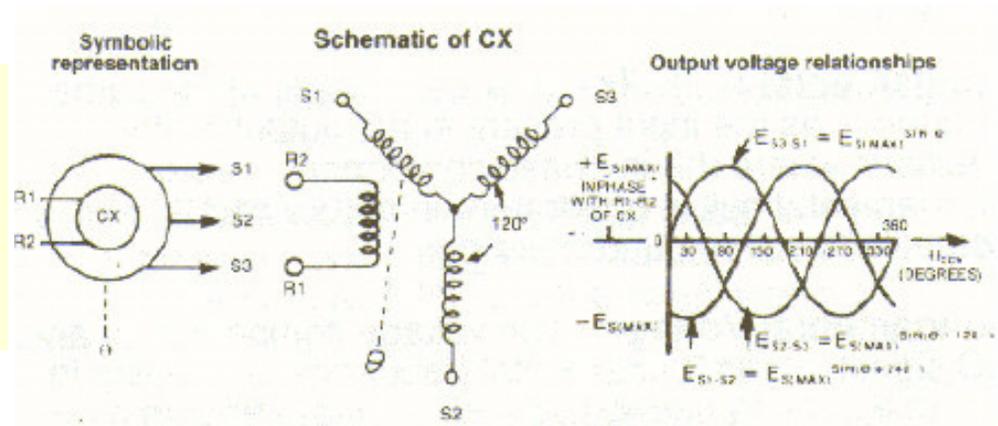
● **Sensores Internos: Sincroresolvers**

Las bobinas fijas forman un triángulo trifásico en estrella (tres bobinas fijas desfasadas 120°).

$$V_{13} = \sqrt{3} V \cos(\omega t) \sin \theta$$

$$V_{32} = \sqrt{3} V \cos(\omega t) \cos(\theta + 120^\circ)$$

$$V_{21} = \sqrt{3} V \cos(\omega t) \cos(\theta + 240^\circ)$$



👉 **Comparación**

	Robustez Mecánica	Rango Dinámico	Resolución	Estabilidad Térmica
Encoder	Mala	Media	Buena	Buena
Resolver	Buena	Buena	Buena	Buena
Potenciómetro	regular	Mala	Mala	Mala



Elementos de un robot industrial: Sistema Sensorial

● **Sensores Internos: Sensores de velocidad**

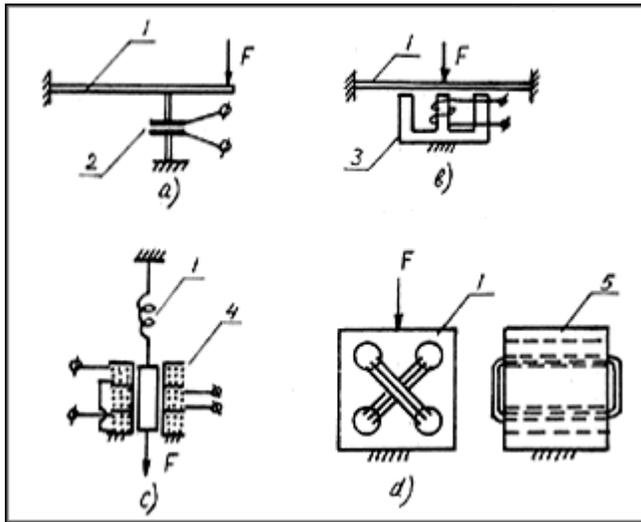
- ▶ La captación de la velocidad se hace necesaria para mejorar el comportamiento dinámico de los actuadores del robot.
- ▶ La información de la velocidad de movimiento de cada actuador se realimenta normalmente a un bucle de control analógico implementado en el propio accionador del elemento motor.
- ▶ Normalmente, y debido a que el bucle de control de velocidad es analógico, el captador utilizado es un *tacogeneratriz* que proporciona una tensión proporcional a la velocidad de giro de su eje.



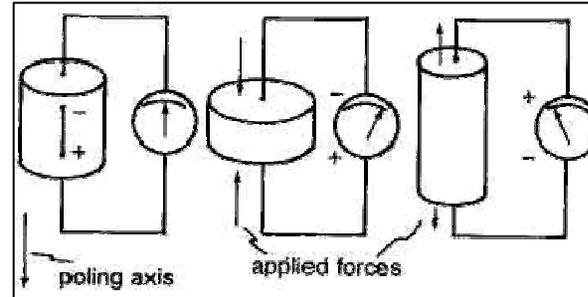


Elementos de un robot industrial: Sistema Sensorial

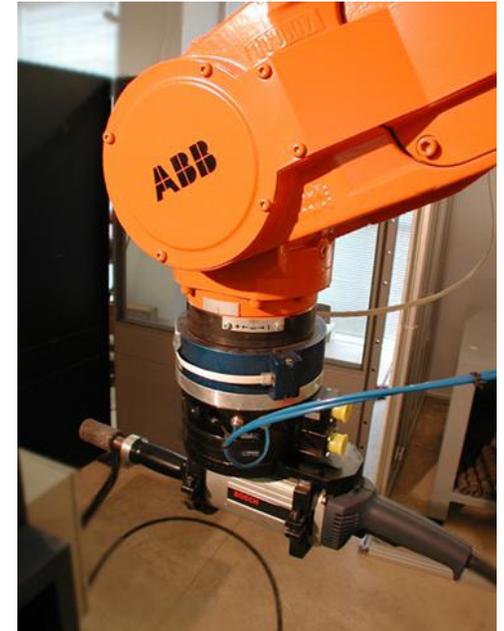
- **Sensores Internos: Sensores de fuerza y par**



Reactancia variable



Piezoeléctrico



JR3



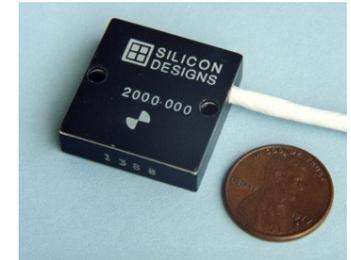
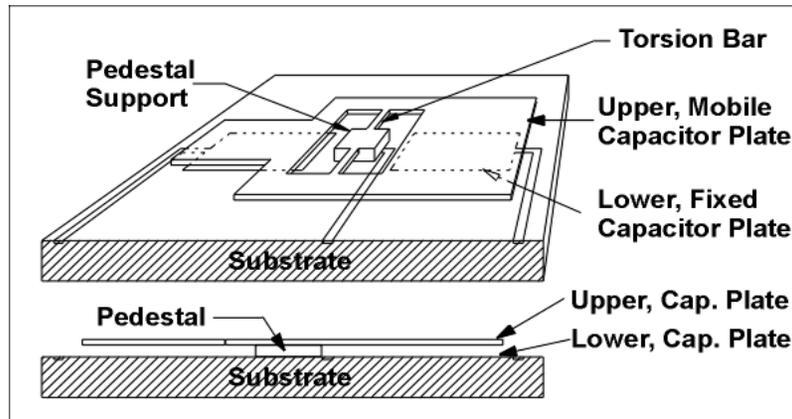
Más información: http://robotics.dem.uc.pt/norberto/jr3pci/ft_sensors.htm



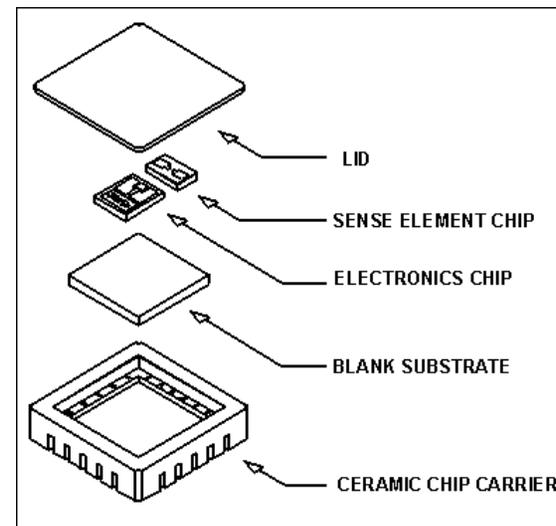
Elementos de un robot industrial: Sistema Sensorial

● Sensores Internos: Sensores de aceleración

Microsistemas electromecánicos (MEMS)

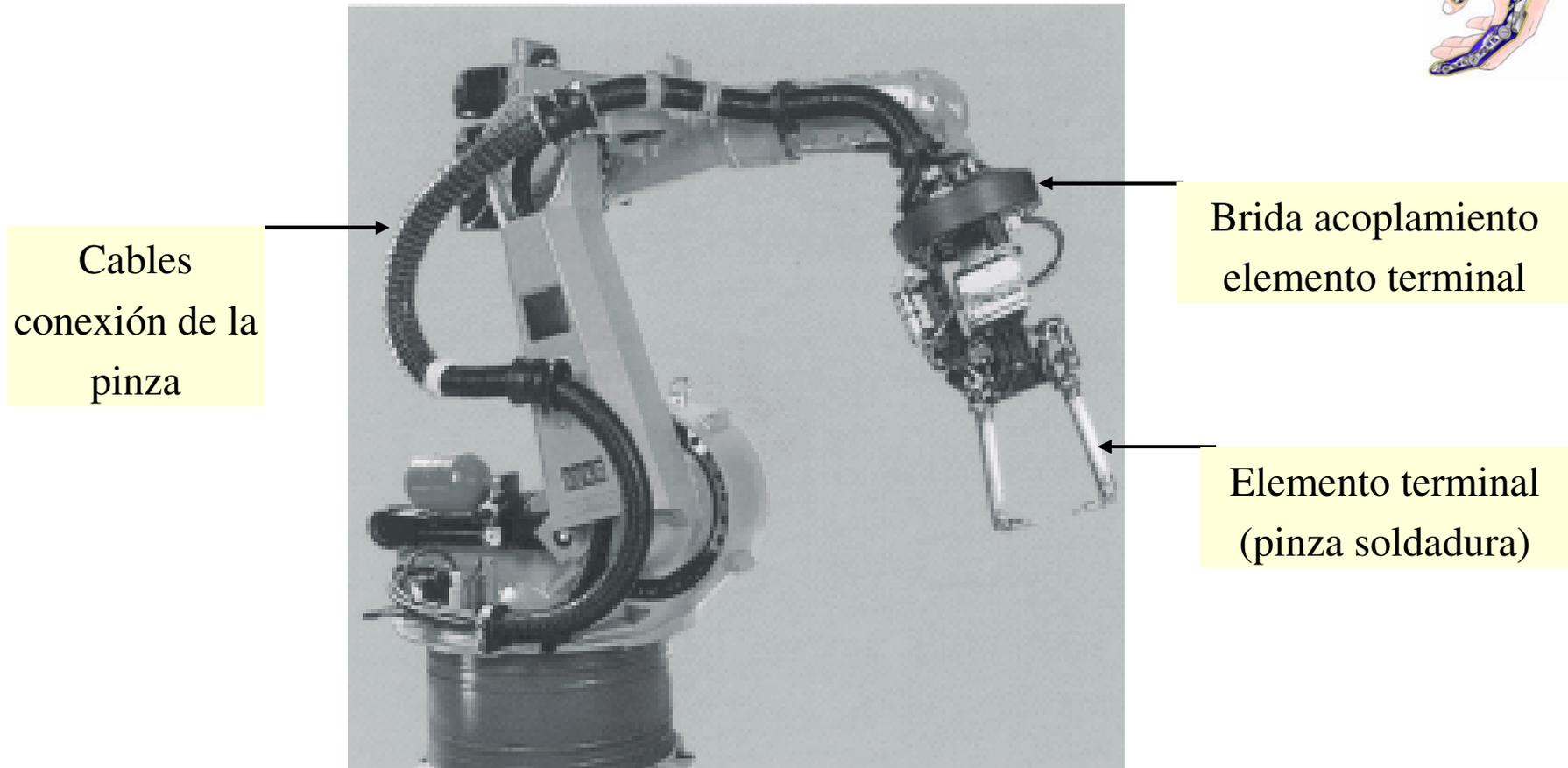


SILICON DESIGNS. Inc.





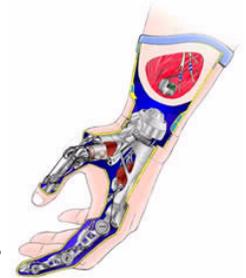
Elementos de un robot industrial: Elementos terminales





Elementos de un robot industrial: Elementos terminales

- **Elemento final o la garra (End-effector)**



- ▶ Los elementos finales también llamados efectores finales (end effector), son los encargados de interactuar directamente con el entorno del robot (agarrar o manipular objetos).

- ▶ Pueden ser tanto elementos de aprehensión como herramientas.

- ▶ En muchos casos diseñados específicamente para cada tipo de trabajo.

- ▶ **Clasificación**

- Elementos terminales de sujeción

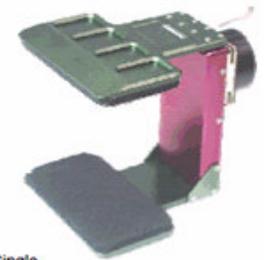
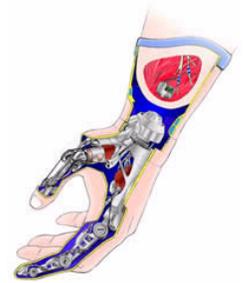
- Elementos terminales herramienta



Elementos de un robot industrial: Elementos terminales

● Elementos terminales de sujeción

- ▶ Se utilizan para agarrar y sostener los objetos y se suelen denominar pinzas.
- ▶ En la elección la pinza se deben considerar varios factores:
 - El peso, forma y tamaño del objeto a manipular
 - la fuerza a ejercer necesaria para sujetarlo
- ▶ El accionamiento neumático suele ser el más utilizado por ofrecer mayores ventajas en simplicidad, precio y fiabilidad, aunque presenta problemas de control en posiciones intermedias.



Single-Opening Hand



Double-Opening Hand



WitchHand®



Elementos de un robot industrial: Elementos terminales

- **Elementos terminales de sujeción**

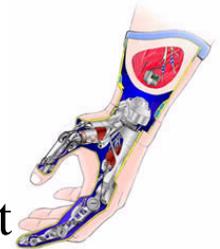


Tipo de sujeción	Accionamiento	Uso
Pinza de presión desp. Lineal desp. angular	Neumático Eléctrico	Transporte y manipulación de piezas sobre las que no importe presionar.
Pinza de enganche	Neumático Eléctrico	Piezas de grandes dimensiones o sobre las que no se pueda presionar.
Ventosas de vacío	Neumático	Cuerpos con superficies lisas poco porosas (cristal, plástico, etc.)
Electroimán	Eléctrico	Piezas ferromagnéticas.



Elementos de un robot industrial: Elementos terminales

- **Elementos terminales: herramientas**



▶ Normalmente, la herramienta está fijada rígidamente al extremo del robot aunque en ocasiones se dota a éste de un dispositivo de cambio automático, que permite al robot utilizar diferentes herramientas durante su tarea.

Tipo de Herramienta

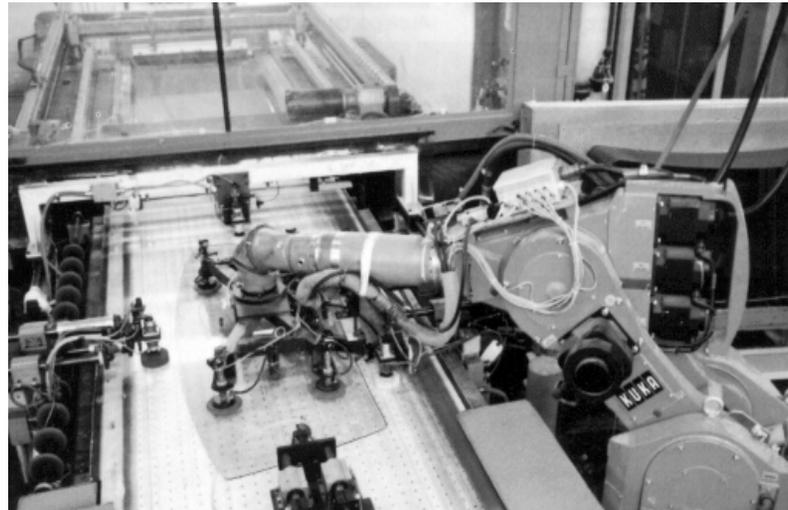
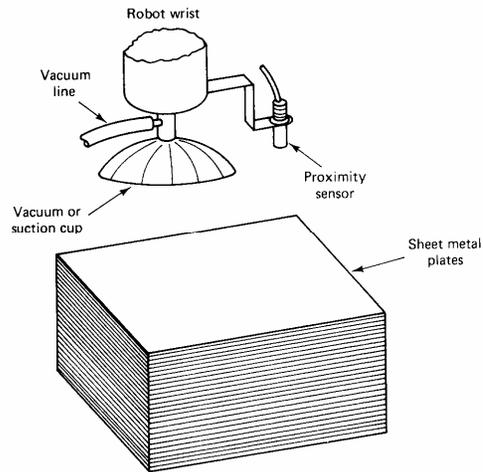
Comentario

Tipo de Herramienta	Comentario
Pinza soldadura por puntos	Dos electrodos que se cierran sobre la pinza al soldar
Soplete soldadura arco	Aportan el flujo del electrodo que se funde
Cucharón para colada	Para trabajos de fundición
Atornillador	Suelen incluir la alimentación de tornillos
Fresa - Lija	Para perfilar, eliminar rebabas, pulir, etc.
Pistola pintura	Por pulverización de la pintura
Cañón Láser	Para corte de materiales, soldadura o inspección
Cañón chorro de agua	Para corte de materiales



Elementos de un robot industrial: Elementos terminales

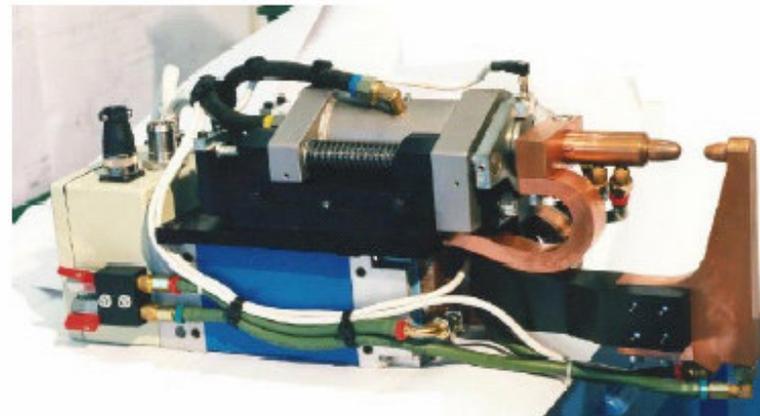
- **Herramientas: Ventosas**





Elementos de un robot industrial: Elementos terminales

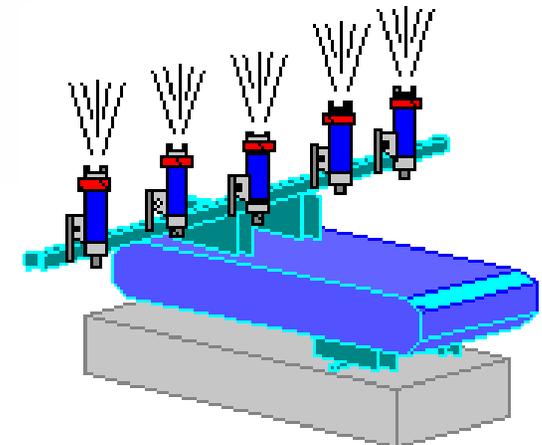
- Herramientas: Pistola de soldadura





Elementos de un robot industrial: Elementos terminales

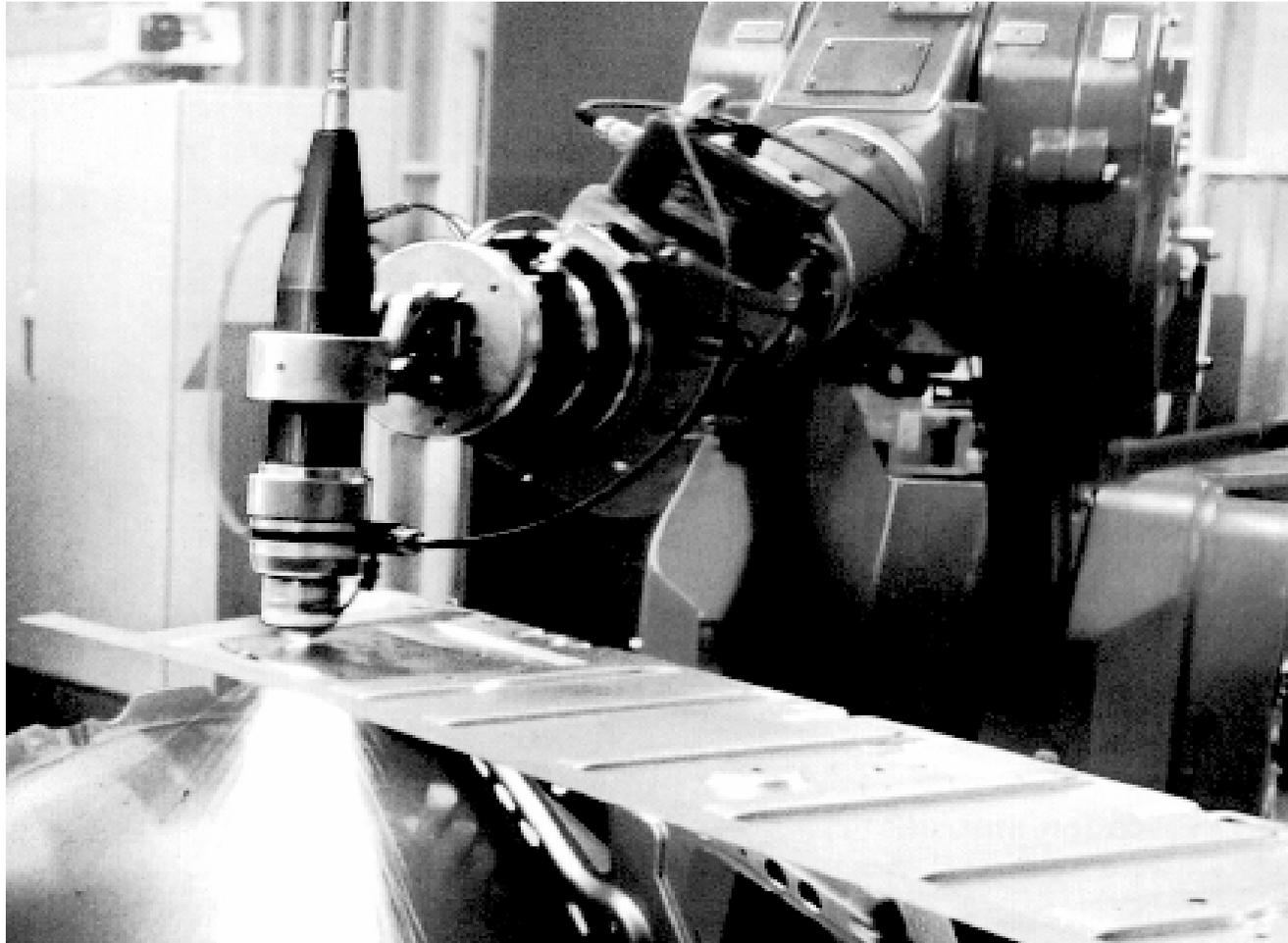
- **Herramientas: Pintura**





Elementos de un robot industrial: Elementos terminales

- Herramientas: Láser de corte





Elementos de un robot industrial: Elementos terminales

● Herramientas: Cambiadores de herramientas



- ▶ Permiten cambiar rápidamente la herramienta terminal del robot.
- ▶ Constan de un plato principal conectado rígidamente a la muñeca del robot, y de varios platos secundarios que portan distintas herramientas.
- ▶ Existen diversos sistemas de acoplamiento entre los platos.
- ▶ Es necesario transportar diversas señales (eléctricas, neumáticas, hidráulicas) entre los platos.





Elementos de un robot industrial: Unidad de control

- ▶ Uno o varios microprocesadores, memorias (RAM, ROM), convertidores (D/A, A/D), unidades de Entrada/Salida, puertos (serie, paralelo), contadores-temporizadores, coprocesadores especializados, etc.
- ▶ Recibe las ordenes del usuario y las transmite a los elementos actuadores (accionadores de motores).
- ▶ Supervisa el correcto funcionamiento de los actuadores mediante realimentación.

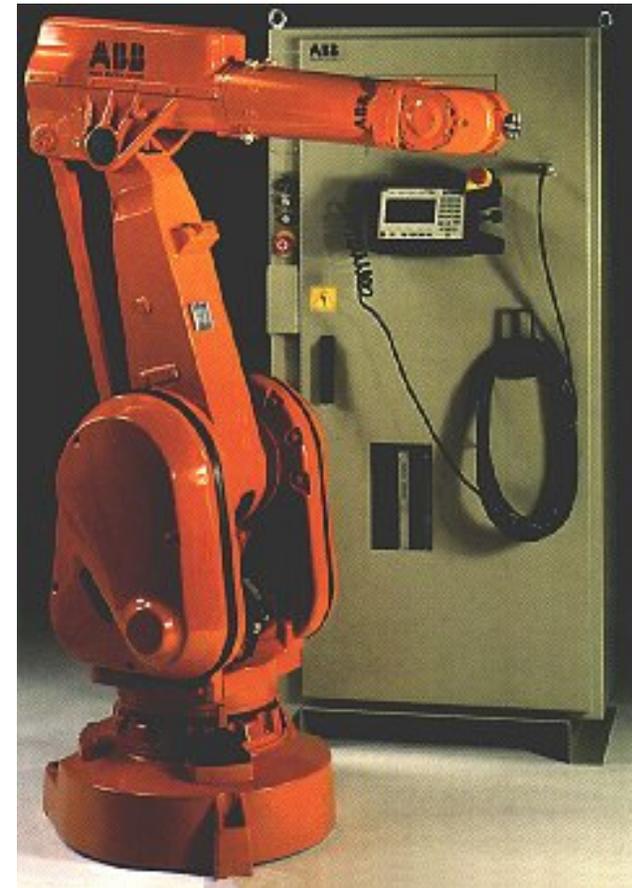




Robots Industriales Comerciales

ABB IRB2400

Fabricante	ABB
Modelo	IRB2400
Aplicación	Uso General
Configuración	Angular
G.D.L.	6
Alcance Horiz.	1542 mm
Carga Max.	10 Kg.
Veloc. Max.	4000 mm/seg
Repetibilidad	0,08 mm.
Accionamiento	Eléctrico c.a.





Robots Industriales Comerciales

ABB IR5002

Fabricante	ABB
Modelo	TR5002
Aplicación	Pintura
Configuración	Angular
G.D.L.	6
Alcance Horiz.	2574 mm
Carga Max.	5 Kg.
Veloc. Max.	2000 mm/seg
Repetibilidad	1,0 mm.
Accionamiento	Eléctrico c.a.





Robots Industriales Comerciales

Adept Three

Fabricante	Adept
Modelo	Three
Aplicación	Uso General
Configuración	SCARA
G.D.L.	4
Alcance Horiz.	1070 mm
Carga Max.	25 Kg.
Veloc. Max.	11000 mm/seg
Repetibilidad	0,025 mm.
Accionamiento	Directo c.c.

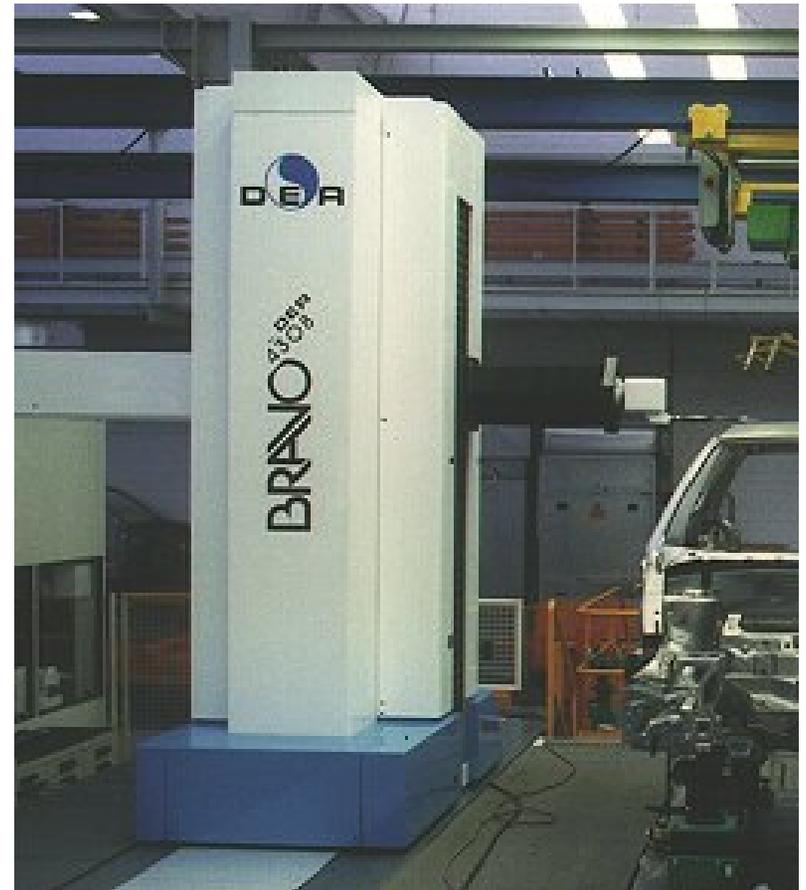




Robots Industriales Comerciales

DEA Bravo 2205

Fabricante	DEA
Modelo	Bravo
Aplicación	Medición
Configuración	Cartesiana
G.D.L.	3
Alcance Horiz.	X:2525 Y:1185 mm
Carga Max.	-
Veloc. Max.	500 mm/seg
Repetibilidad	0,3 mm.
Accionamiento	Eléctrico c.c.





Robots Industriales Comerciales

Kawasaki JS10

Fabricante	Kawasaki
Modelo	JS10
Aplicación	Corte Chorro Agua
Configuración	Angular
G.D.L.	6
Alcance Horiz.	1475 mm
Carga Max.	10 Kg.
Veloc. Max.	1500 mm/seg
Repetibilidad	0.1 mm.
Accionamiento	Brushless.





Robots Industriales Comerciales

Kawasaki JS10

Fabricante	Kawasaki
Modelo	JS10
Aplicación	Corte Chorro Agua
Configuración	Angular
G.D.L.	6
Alcance Horiz.	1475 mm
Carga Max.	10 Kg.
Veloc. Max.	1500 mm/seg
Repetibilidad	0.1 mm.
Accionamiento	Brushless.





Robots Industriales Comerciales

Kawasaki ARCJS

Fabricante	Kawasaki
Modelo	ARCJS
Aplicación	Soldadura por Arco
Configuración	Angular
G.D.L.	6
Alcance Horiz.	1475 mm
Carga Max.	6 Kg.
Veloc. Max.	-
Repetibilidad	0.1 mm.
Accionamiento	-

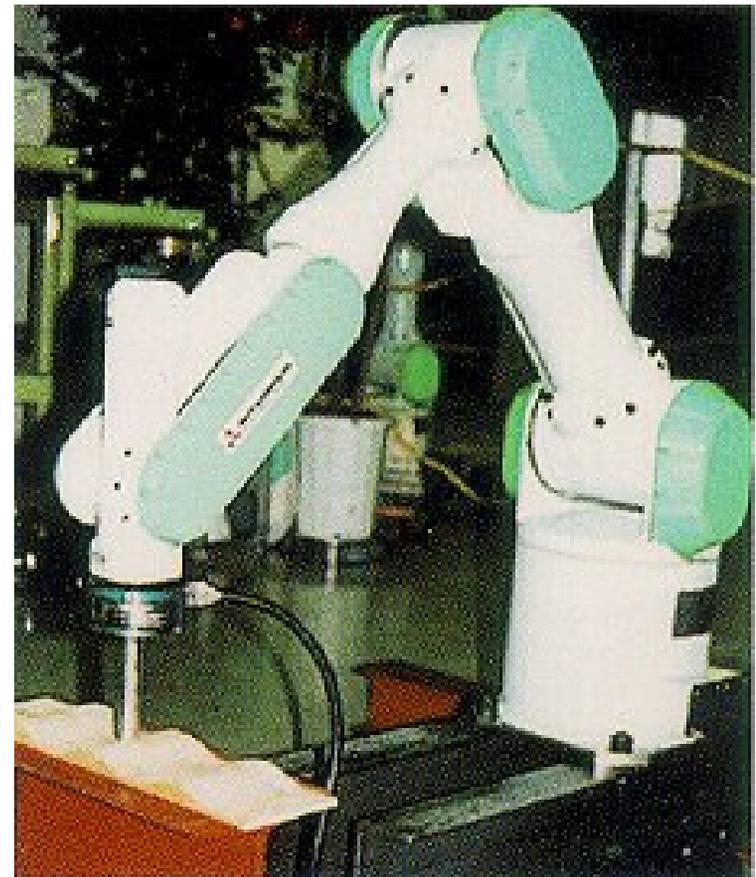




Robots Industriales Comerciales

Mitshubisi PA-10

Fabricante	Mitshubisi
Modelo	PA-10
Aplicación	Uso General
Configuración	Angular
G.D.L.	7
Alcance Horiz.	1030 mm
Carga Max.	10 Kg.
Veloc. Max.	1550 mm/seg
Repetibilidadd	0.1 mm.
Accionamiento	Eléctrica c.a.





Resumen:

- Un **robot industrial** está formado por los siguientes elementos: estructura mecánica, transmisiones, sistema de accionamiento, sistema sensorial, sistema de control y elementos terminales.
- Mecánicamente, un **robot industrial** está formado por una serie de elementos o eslabones unidos mediante articulaciones que permiten un movimiento relativo entre cada dos eslabones consecutivos.
- El tipo de movimiento relativo permitido por las articulaciones que unen los eslabones de la estructura mecánica del robot definen su configuración.
- El número de grados de libertad (**GDL**) de un robot industrial es el **número de movimientos independientes** que puede realizar el robot respecto a su base:
- Los **elementos finales** también llamados efectores finales (end effector), son los encargados de interactuar directamente con el entorno del robot.



Automatización Industrial - II

56 – 10569 2° Cuatrimestre 2006

Clase 3 – Miércoles 26 de Abril 2006

Morfología del robot



Automatización Industrial - II

56 – 10569 2° Cuatrimestre 2006

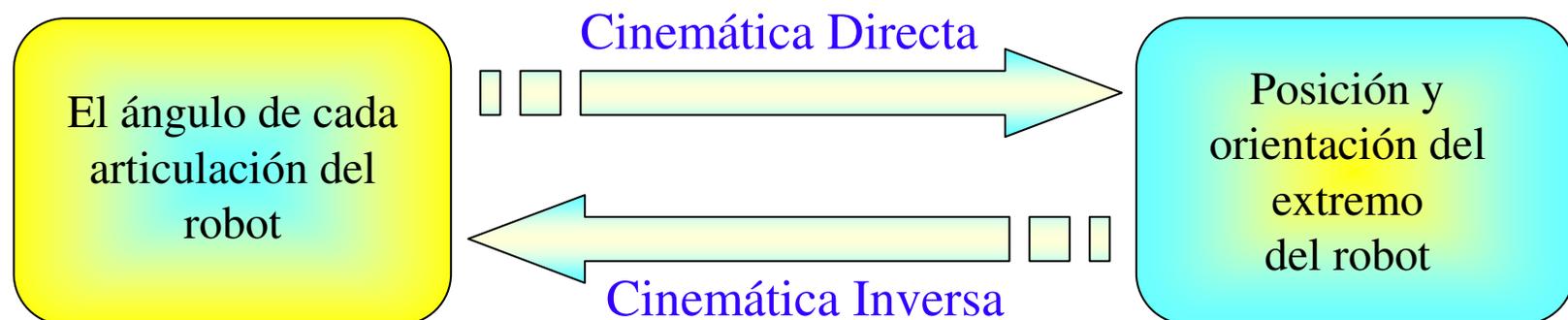
Clase 5 – Miércoles 3 de Mayo 2006

Cinemática - I



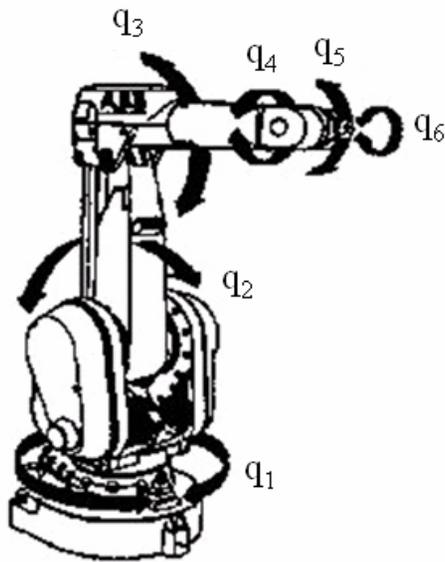
Problemas cinemáticos

- ▶ La cinemática del robot estudia el movimiento del mismo, especialmente de su extremo, con respecto a un sistema base **sin tener en cuenta las fuerzas o pares** que generan dichos movimientos.
- ▶ La cinemática se interesa por la descripción analítica del movimiento espacial del robot como una función del tiempo y en particular por las relaciones entre la **posición y la orientación** del extremo final del robot con los valores que toman sus coordenadas articulares.





Problemas cinemáticos

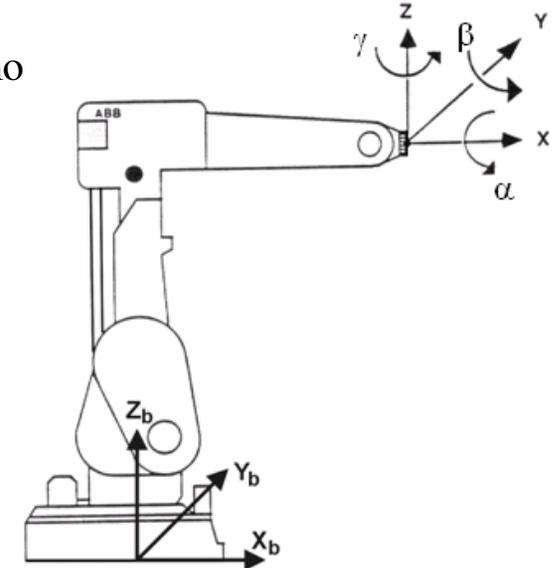


Dado: Largos y configuración de cada eslabón.
El ángulo de cada articulación

Determinar: Posición y orientación del extremo del robot $(x, y, z, \alpha, \beta, \gamma)$

Cinemática Directa

Cinemática Inversa

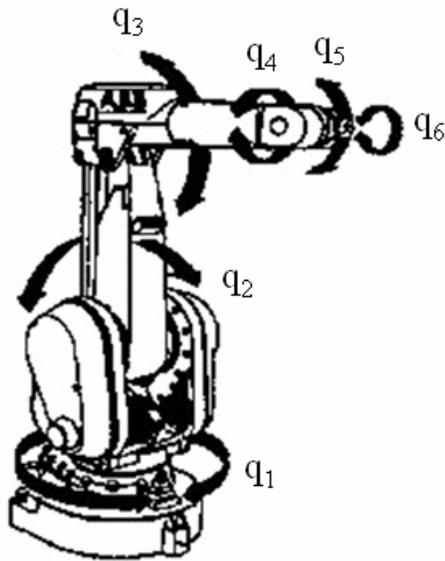


Dado: Largos y configuración de cada eslabón.
Posición y orientación del extremo del robot.

Determinar: Los ángulos de cada articulación necesarios para determinar aquella posición $(q_1, q_2, q_3, q_4, q_5, q_6)$.



Problemas cinemáticos

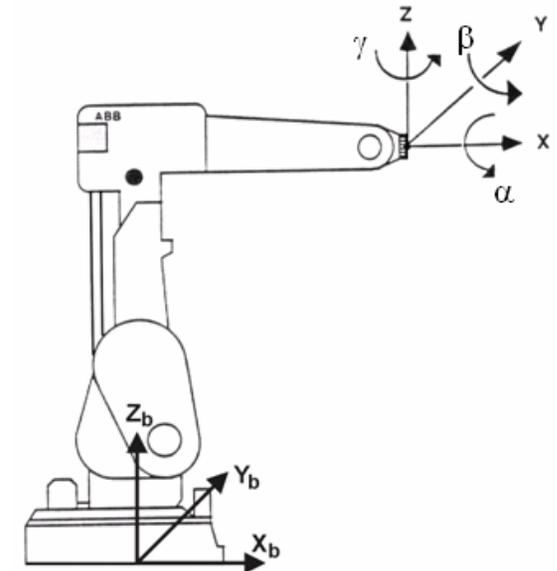


$$q_k = f_k(x, y, z, \alpha, \beta, \gamma)$$

$K = 1, \dots, n$ (n es GDL = 6)

Cinemática Directa

Cinemática Inversa



$$x = f_x(q_1, q_2, q_3, q_4, q_5, q_6)$$

$$y = f_y(q_1, q_2, q_3, q_4, q_5, q_6)$$

$$z = f_z(q_1, q_2, q_3, q_4, q_5, q_6)$$

$$\alpha = f_\alpha(q_1, q_2, q_3, q_4, q_5, q_6)$$

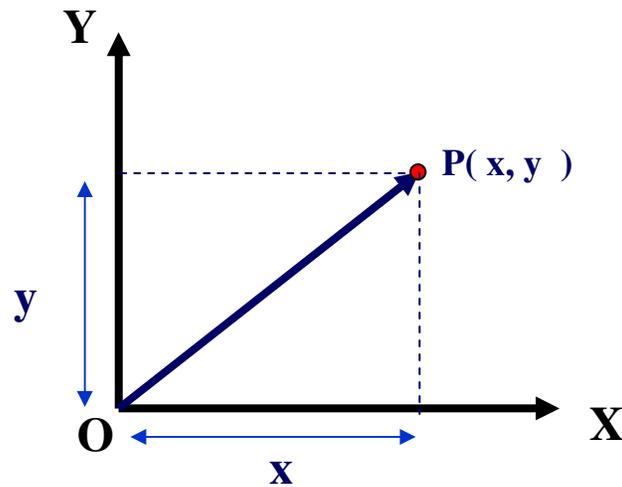
$$\beta = f_\beta(q_1, q_2, q_3, q_4, q_5, q_6)$$

$$\gamma = f_\gamma(q_1, q_2, q_3, q_4, q_5, q_6)$$

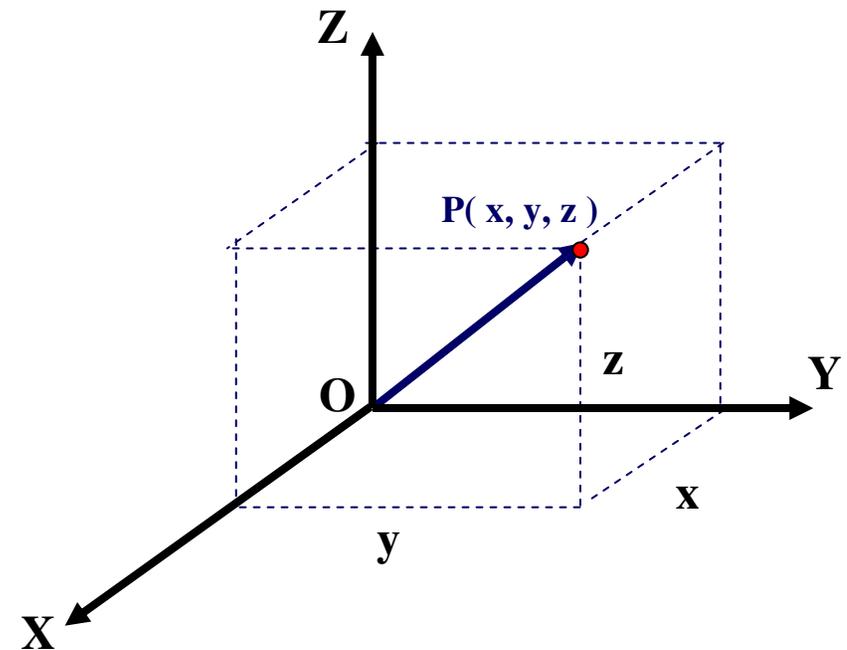


Representación de un punto

- **Coordenadas Cartesianas**



Coordenadas Cartesianas 2D

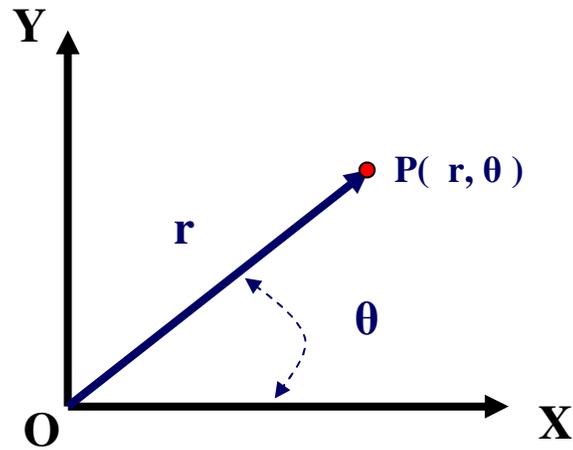


Coordenadas Cartesianas 3D

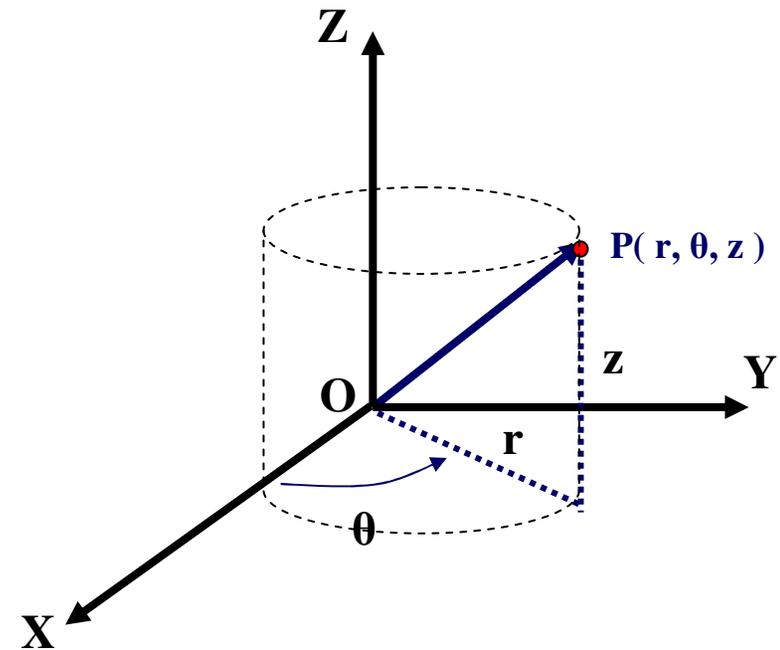


Representación de un punto

- **Coordenadas Polares**



Coordenadas Polares 2D

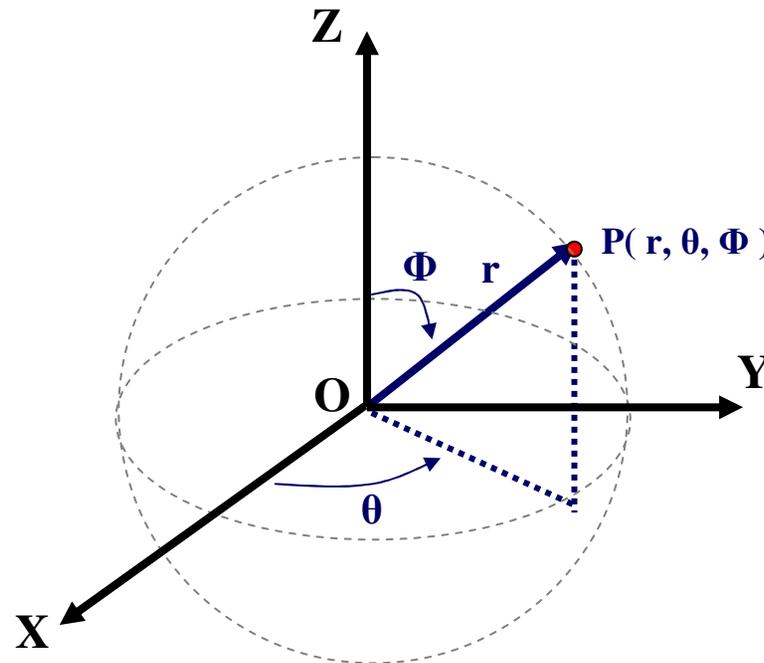


Coordenadas Cilíndricas



Representación de un punto

- **Coordenadas Esféricas**





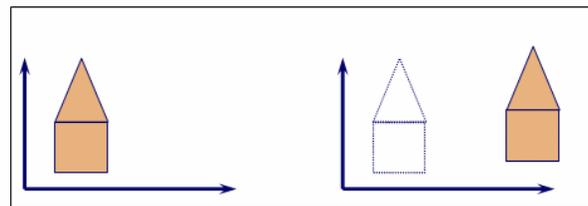
Localización espacial de un cuerpo regido

- ▶ Un punto queda totalmente definido en el espacio a través de los datos de su posición.
- ▶ En el caso de un **objeto sólido**, es necesario además definir cuál es su **orientación** con respecto a un sistema de referencia.

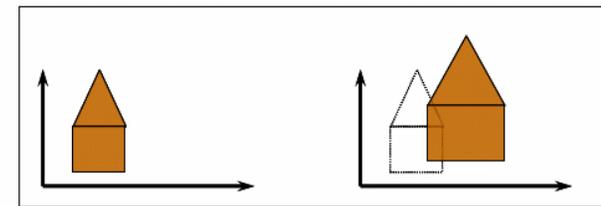
☛ Transformaciones

Para transformar un objeto transformamos cada uno de sus vértices.

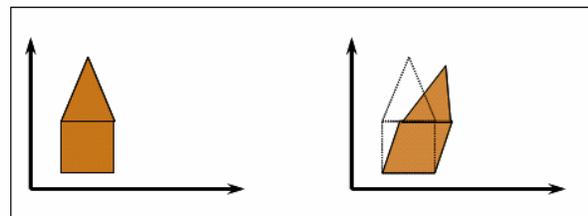
Traslación



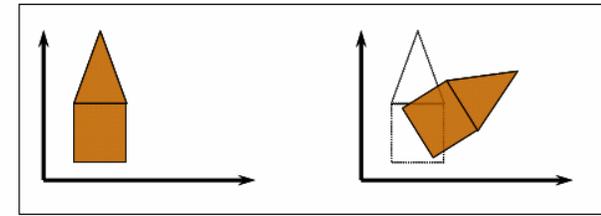
Escalado



Deformación



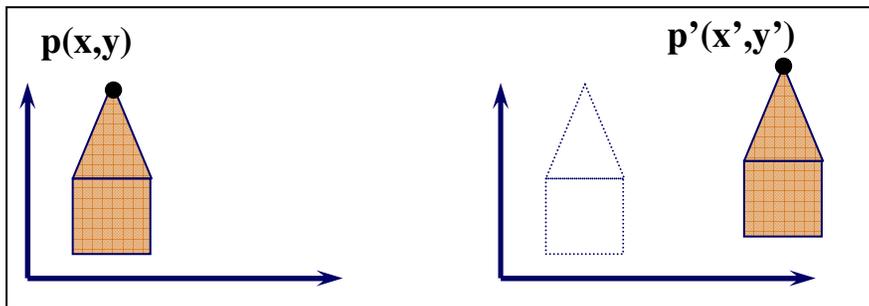
Rotación





Localización espacial de un cuerpo rígido

- Transformaciones: Traslación 2D



$$\begin{aligned}x' &= x + t_x \\ y' &= y + t_y\end{aligned}$$

Notación matricial

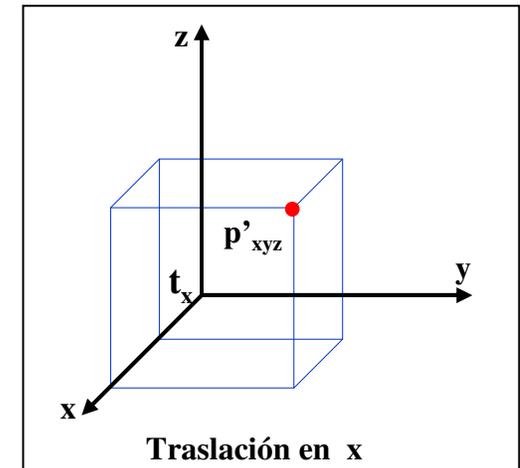
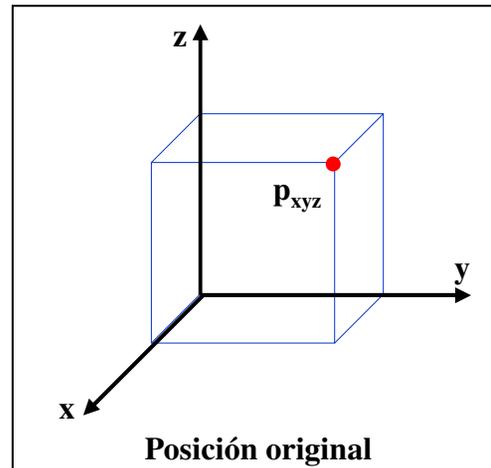
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



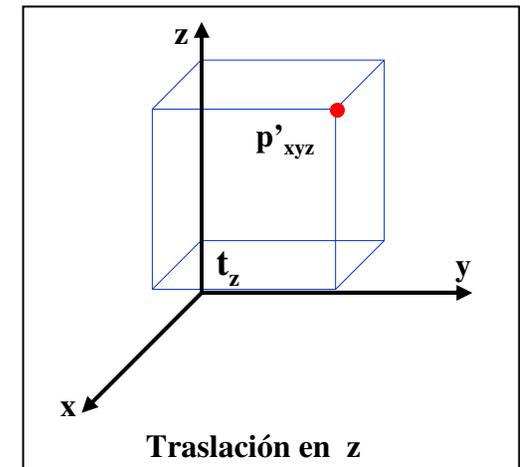
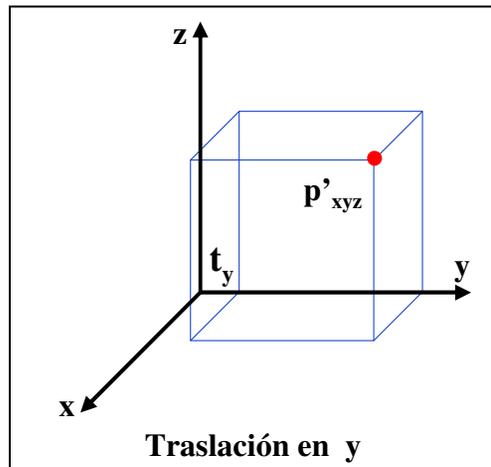
Localización espacial de un cuerpo regido

● Transformaciones: Traslación 3D

$$\begin{aligned} x' &= x + t_x \\ y' &= y + t_y \\ z' &= z + t_z \end{aligned}$$



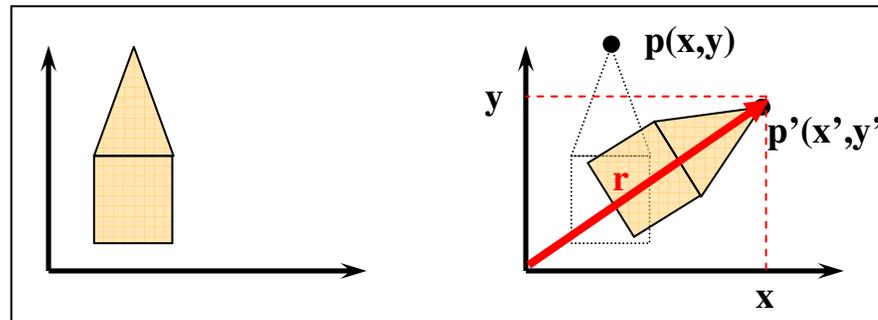
$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$





Localización espacial de un cuerpo regido

● Transformaciones: Rotación 2D



Relaciones básicas

$$x = r \cos a$$

$$y = r \sin a$$

$$r = \sqrt{(x^2 + y^2)}$$

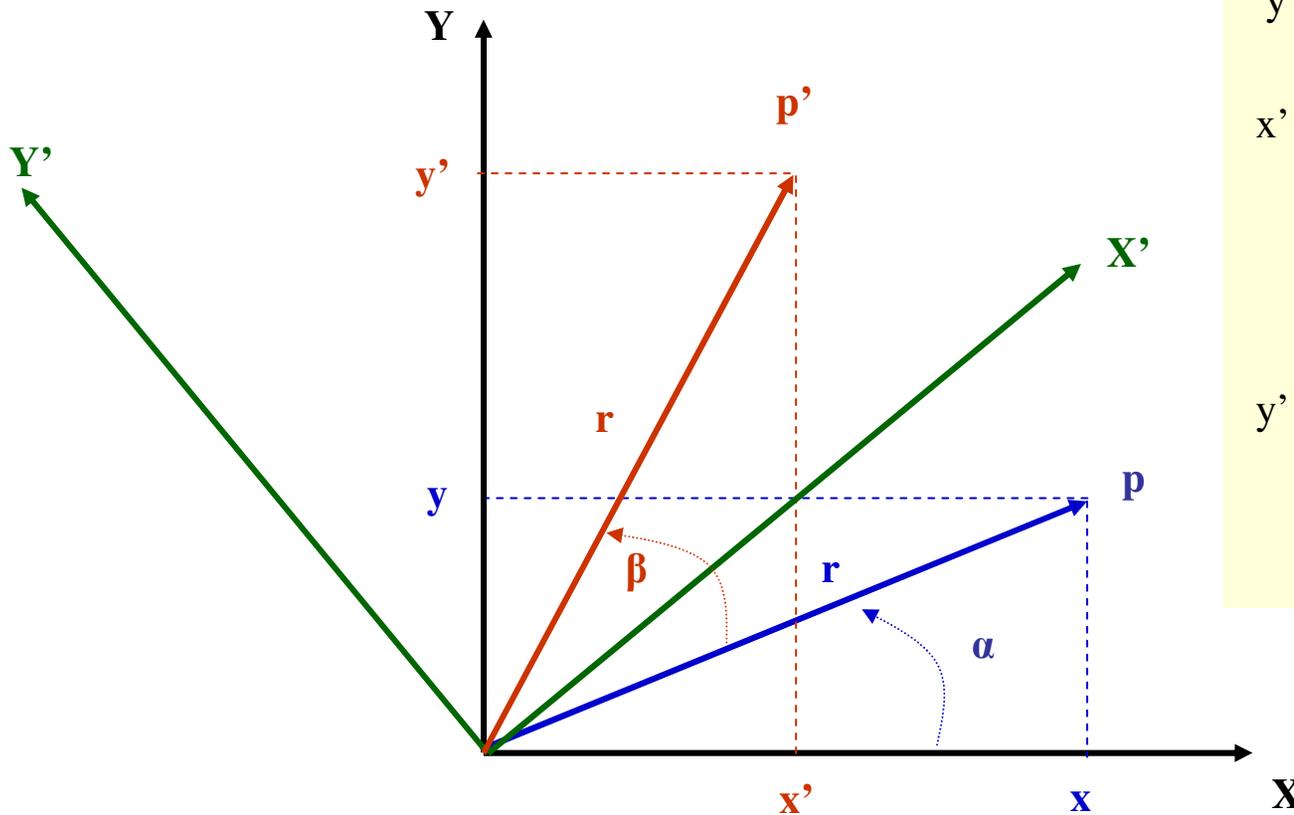
Relaciones trigonométricas

1	$\sin (a - b)$	$\sin a \cos b - \cos a \sin b$
2	$\sin (a + b)$	$\sin a \cos b + \cos a \sin b$
3	$\cos (a - b)$	$\cos a \cos b + \sin a \sin b$
4	$\cos (a + b)$	$\cos a \cos b - \sin a \sin b$



Localización espacial de un cuerpo regido

● Transformaciones: Rotación 2D



$$x = r \cos(\alpha)$$

$$y = r \sin(\alpha)$$

$$\begin{aligned}x' &= r \cos(\alpha + \beta) \\ &= r (\cos \alpha \cos \beta - \sin \alpha \sin \beta) \\ &= x \cos \beta - y \sin \beta\end{aligned}$$

$$\begin{aligned}y' &= r \sin(\alpha + \beta) \\ &= r (\cos \alpha \sin \beta + \sin \alpha \cos \beta) \\ &= x \sin \beta + y \cos \beta\end{aligned}$$



Localización espacial de un cuerpo rígido

● Transformaciones: Rotación 2D

Representación Matricial

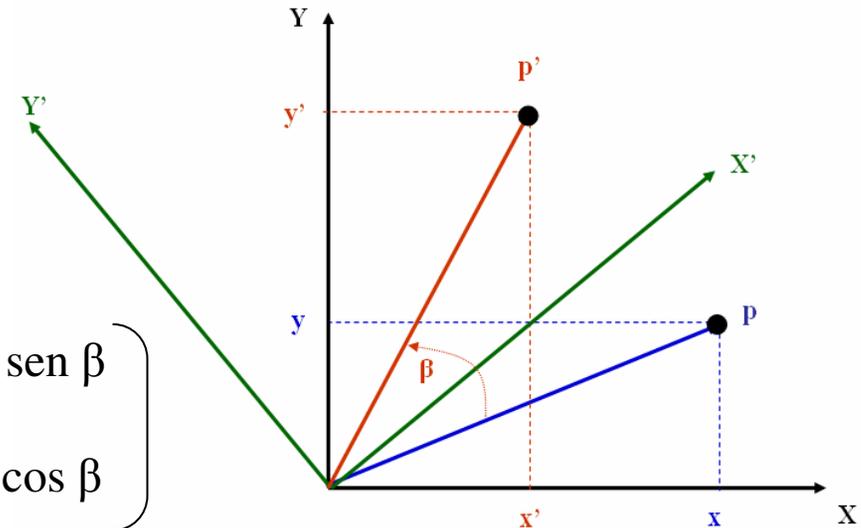
$$\begin{pmatrix} x' \\ y' \end{pmatrix} = R \begin{pmatrix} x \\ y \end{pmatrix}$$

$$R = \begin{pmatrix} \cos \beta & -\operatorname{sen} \beta \\ \operatorname{sen} \beta & \cos \beta \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{x}' \\ \mathbf{y}' \end{pmatrix} = \begin{pmatrix} \cos \beta & -\operatorname{sen} \beta \\ \operatorname{sen} \beta & \cos \beta \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}$$

$$x' = x \cos \beta - y \operatorname{sen} \beta$$

$$y' = x \operatorname{sen} \beta + y \cos \beta$$



R es una matriz de rotación orto-normal $R^T = R^{-1}$

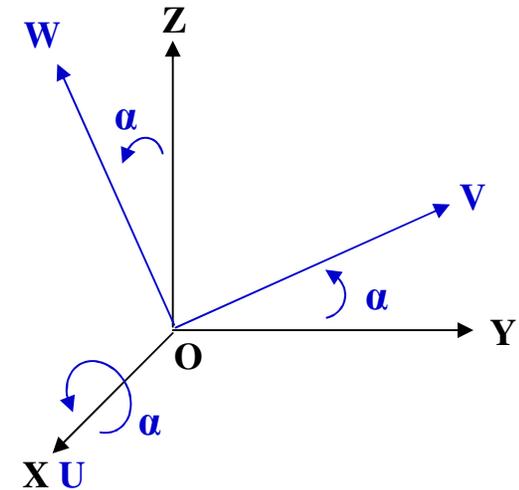


Localización espacial de un cuerpo regido

● Transformaciones: Rotación 3D

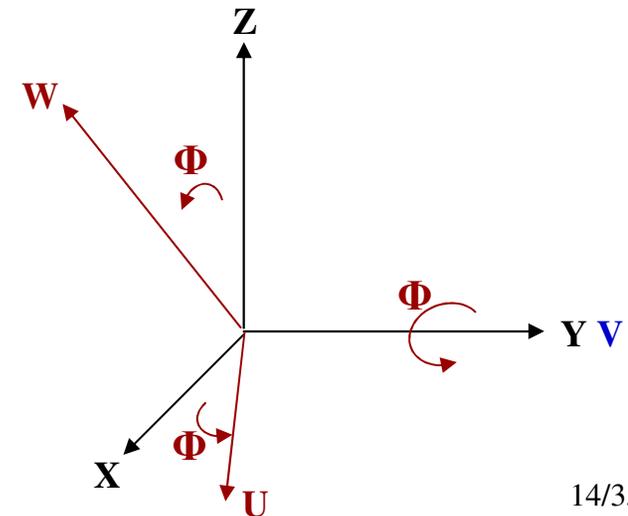
Matriz de rotación Giro X

$$R(x,\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\text{sen } \alpha \\ 0 & \text{sen } \alpha & \cos \alpha \end{pmatrix}$$



Matriz de rotación Giro Y

$$R(y,\Phi) = \begin{pmatrix} \cos \Phi & 0 & \text{sen } \Phi \\ 0 & 1 & 0 \\ -\text{sen } \Phi & 0 & \cos \Phi \end{pmatrix}$$



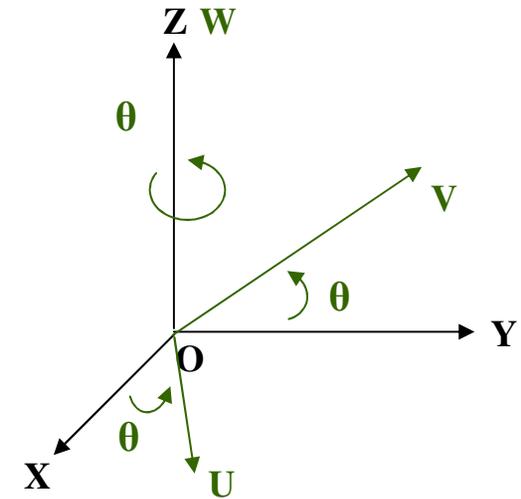


Localización espacial de un cuerpo rígido

- Transformaciones: Rotación 3D

Matriz de rotación Giro Z

$$R(z,\theta) = \begin{pmatrix} \cos \theta & -\text{sen } \theta & 0 \\ \text{sen } \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$





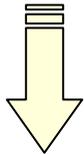
Localización espacial de un cuerpo rígido

● Transformaciones: Rotación 3D - Composición de rotaciones

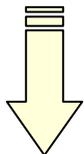
Las matrices de rotación pueden componerse para expresar la aplicación continua de varias rotaciones.

Si al sistema **OUVW** se le aplica :

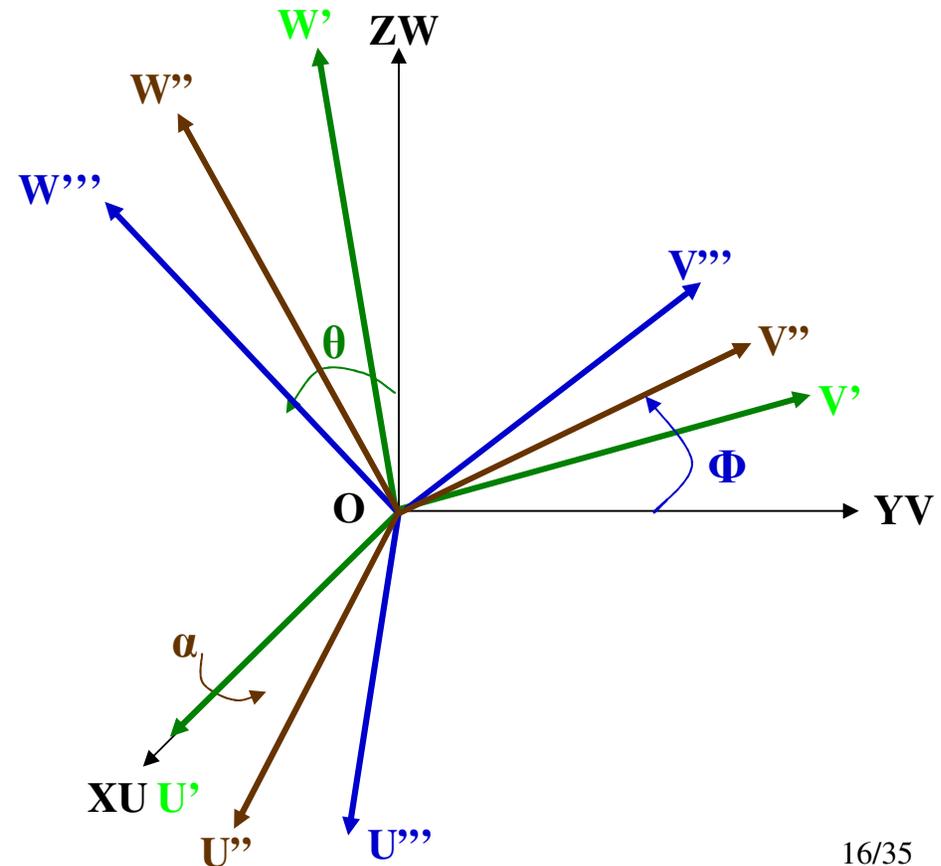
una rotación un ángulo α sobre **OX**



una rotación un ángulo Φ sobre **OY**



una rotación un ángulo θ sobre **OZ**

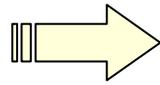




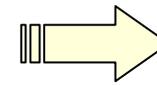
Localización espacial de un cuerpo rígido

● Transformaciones: Rotación 3D - Composición de rotaciones

una rotación un
ángulo α sobre **OX**



una rotación un
ángulo Φ sobre **OY**



una rotación un
ángulo θ sobre **OZ**

La rotación global será:

$$T = R(z,\theta)R(y,\Phi)R(x,\alpha) = \begin{pmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C\Phi & 0 & S\Phi \\ 0 & 1 & 0 \\ -S\Phi & 0 & C\Phi \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & C\alpha & -S\alpha \\ 0 & S\alpha & C\alpha \end{pmatrix}$$

$$T = R(z,\theta)R(y,\Phi)R(x,\alpha) = \begin{pmatrix} C\theta C\Phi & -S\theta C\alpha + C\theta S\Phi S\alpha & S\theta S\alpha + C\theta S\Phi C\alpha \\ S\theta C\Phi & C\theta C\alpha + S\theta S\Phi S\alpha & -C\theta S\alpha + S\theta S\Phi C\alpha \\ -S\Phi & C\Phi S\alpha & C\Phi C\alpha \end{pmatrix}$$

 El producto de matrices **no es conmutativo**, así se debe considerar el orden en que se realizan las rotaciones.



Localización espacial de un cuerpo regido

● Transformación Homogénea: Coordenadas Homogéneas

- ▶ La presentación mediante coordenadas homogéneas de la localización de sólidos en un espacio **n-dimensional** se realiza a través de coordenadas de un espacio **(n+1)-dimensional**.

$$p = (x, y, z) \rightarrow p_H = (wa, wb, wc, w)$$

w: es un factor de escala

$$P = \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = \begin{pmatrix} wa \\ wb \\ wc \\ w \end{pmatrix} = \begin{pmatrix} a \\ b \\ c \\ 1 \end{pmatrix}$$

Ejemplo:

$$2i + 3j + 4k \rightarrow [2, 3, 4, 1]^T$$

$$[4, 6, 8, 2]^T$$

$$[-6, -9, -12, -3]^T$$



Localización espacial de un cuerpo rígido

● Matriz de transformación Homogénea T

- ▶ Es una matriz 4x4 que representa la transformación de un vector en coordenadas homogéneas de un sistema a otro.

$$T = \begin{pmatrix} R_{3 \times 3} & P_{3 \times 1} \\ F_{1 \times 3} & W_{1 \times 1} \end{pmatrix} \quad T = \begin{pmatrix} \text{Rotación} & \text{Traslación} \\ \text{Perspectiva} & \text{Escalado} \end{pmatrix}$$

- ▶ En robótica suelen ser de interés sólo las sub-matrices $R_{3 \times 3}$ y $P_{3 \times 1}$, considerándose $F_{1 \times 3}$ nulo y $W_{1 \times 1}$ la unidad, utilizando los vectores en coordenadas homogéneas.

$$T = \begin{pmatrix} R_{3 \times 3} & P_{3 \times 1} \\ 0 & 1 \end{pmatrix} \quad T = \begin{pmatrix} \text{Rotación} & \text{Traslación} \\ 0 & 1 \end{pmatrix}$$



Localización espacial de un cuerpo rígido

● Matriz de transformación Homogénea T

► Representa la orientación y posición de un sistema O'UVW rotado y trasladado con respecto al sistema de referencia OXYZ.

► La matriz sirve para conocer las coordenadas (r_x, r_y, r_z) del vector r en el sistema OXYZ a partir de sus coordenadas (r_u, r_v, r_w) en el sistema O'XYZ.

$$\begin{pmatrix} r_x \\ r_y \\ r_z \\ 1 \end{pmatrix} = T \begin{pmatrix} r_u \\ r_v \\ r_w \\ 1 \end{pmatrix}$$

► También para expresar la rotación R3x3 y traslación P3x3 de un vector respecto de un sistema fijo OXYZ como $r'xyz$.

$$\begin{pmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{pmatrix} = T \begin{pmatrix} r_x \\ r_y \\ r_z \\ 1 \end{pmatrix}$$

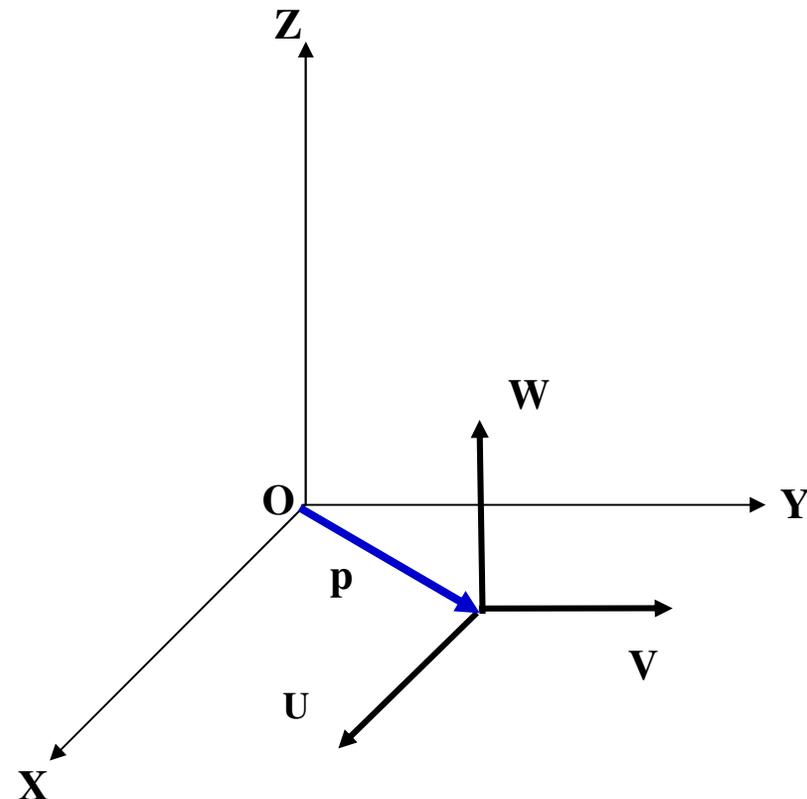


Localización espacial de un cuerpo rígido

● Matriz de transformación Homogénea T: Traslación

- ▶ Matriz de traslación para el sistema O'UVW trasladado el vector p respecto a OXYZ.

$$T = \begin{pmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

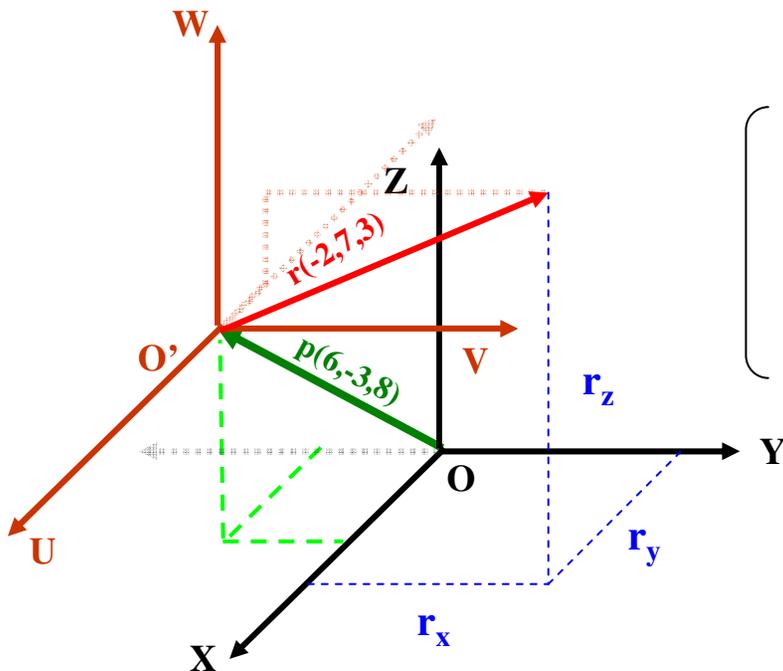




Localización espacial de un cuerpo regido

● Matriz de transformación Homogénea T: Traslación

- ▶ **Ejemplo 1:** Un sistema $O'UVW$ está trasladado un vector $p(6,-3,8)$ con respecto a $OXYZ$, Calcular las coordenadas $p(r_x, r_y, r_z)$ del vector r cuyas coordenadas con respecto al sistema $O'UVW$ son $r_{uvw}(-2,7,3)$.



$$\begin{pmatrix} r_x \\ r_y \\ r_z \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 6 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 8 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} -2 \\ 7 \\ 3 \\ 1 \end{pmatrix} = \begin{pmatrix} 4 \\ 4 \\ 11 \\ 1 \end{pmatrix}$$

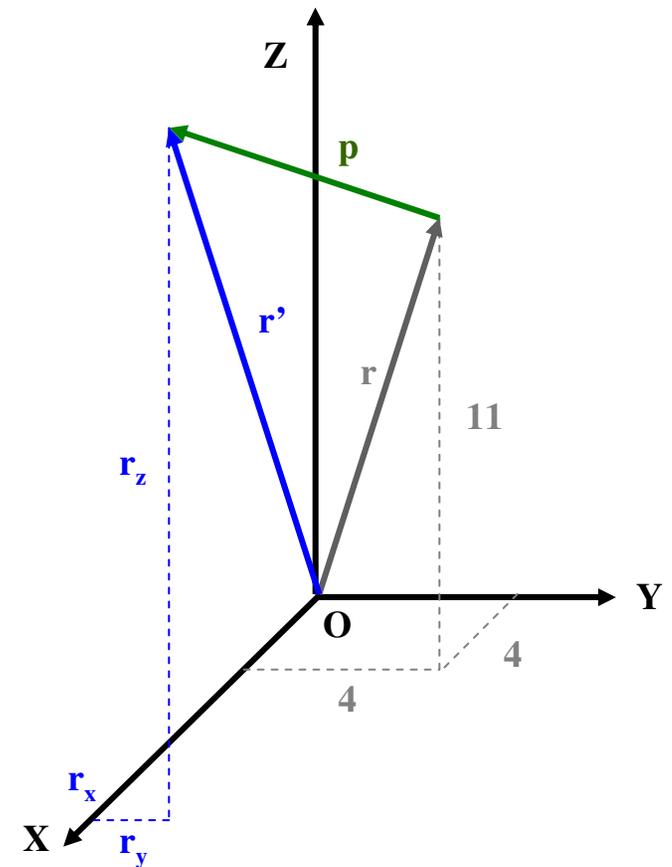


Localización espacial de un cuerpo rígido

● Matriz de transformación Homogénea T: Traslación

- ▶ **Ejemplo 2:** Calcular el vector r'_{xyz} resultante de trasladar al vector $r_{xyz}(4,4,11)$ según la transformación $T(p)$ con $p(6,-3,8)$.

$$\begin{pmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 6 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 8 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 \\ 4 \\ 11 \\ 1 \end{pmatrix} = \begin{pmatrix} 10 \\ 1 \\ 19 \\ 1 \end{pmatrix}$$

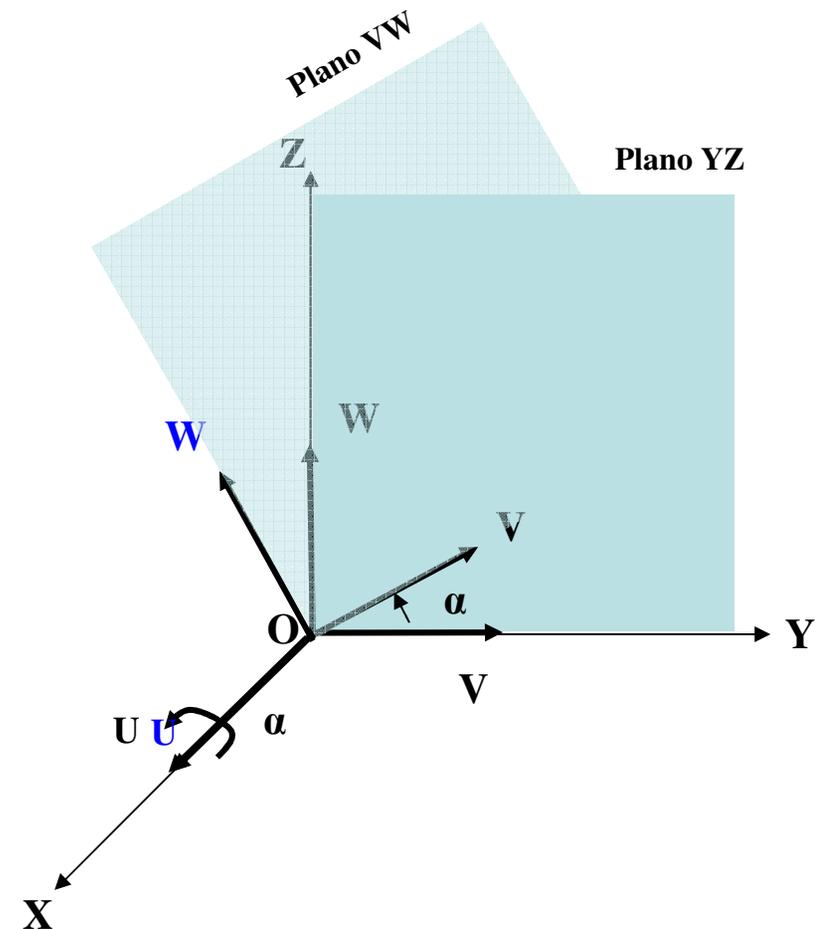




Localización espacial de un cuerpo rígido

- **Matriz de transformación Homogénea T: Rotación X**

$$T(x, \alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\text{sen } \alpha & 0 \\ 0 & \text{sen } \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

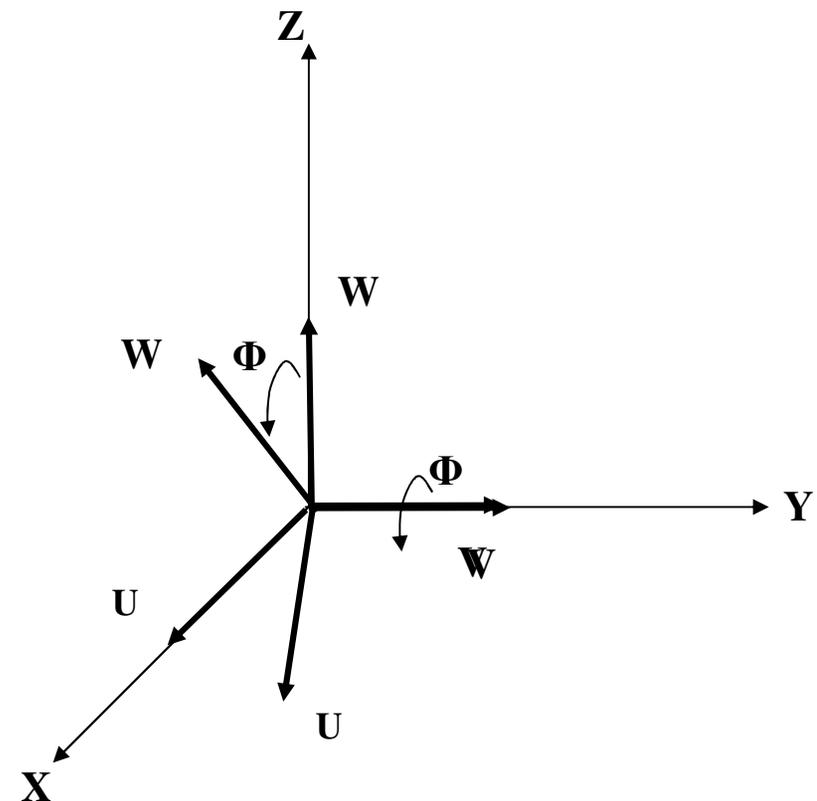




Localización espacial de un cuerpo regido

- **Matriz de transformación Homogénea T: Rotación Y**

$$T(y, \Phi) = \begin{pmatrix} \cos \Phi & 0 & \text{sen } \Phi & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen } \Phi & 0 & \cos \Phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

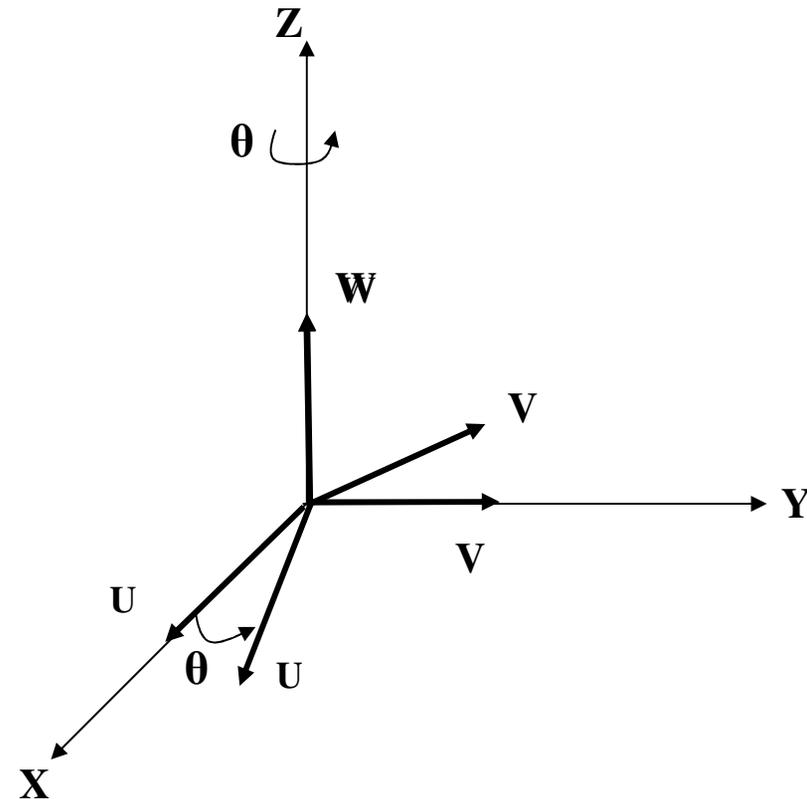




Localización espacial de un cuerpo rígido

- **Matriz de transformación Homogénea T: Rotación Z**

$$T(z, \theta) = \begin{pmatrix} \cos \theta & -\text{sen } \theta & 0 & 0 \\ \text{sen } \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$





Localización espacial de un cuerpo rígido

- **Matriz de transformación Homogénea: Rotación seguida de traslación**

- ▶ Rotación de un ángulo α sobre el eje OX seguido de una traslación p_{xyz} .

$$T((x, \alpha), p) = \begin{pmatrix} 1 & 0 & 0 & p_x \\ 0 & \cos \alpha & -\text{sen } \alpha & p_y \\ 0 & \text{sen } \alpha & \cos \alpha & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- ▶ Rotación de un ángulo Φ sobre el eje OY seguido de una traslación p_{xyz} .

$$T((y, \Phi), p) = \begin{pmatrix} \cos \Phi & 0 & \text{sen } \Phi & p_x \\ 0 & 1 & 0 & p_y \\ -\text{sen } \Phi & 0 & \cos \Phi & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



Localización espacial de un cuerpo rígido

- **Matriz de transformación Homogénea: Rotación seguida de traslación**

▶ Rotación de un ángulo θ sobre el eje OZ seguido de una traslación p_{xyz} .

$$T((z,\theta),p) = \begin{pmatrix} \cos \theta & -\text{sen } \theta & 0 & p_x \\ \text{sen } \theta & \cos \theta & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



Localización espacial de un cuerpo rígido

- **Matriz de transformación Homogénea: Traslación seguida de rotación**

- ▶ Traslación de un vector p_{xyz} seguida de una rotación un ángulo α sobre el eje **OX**

$$T(p,(x,\alpha)) = \begin{pmatrix} 1 & 0 & 0 & p_x \\ 0 & C\alpha & -S\alpha & p_y C\alpha - p_z S\alpha \\ 0 & S\alpha & C\alpha & p_y S\alpha + p_z C\alpha \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- ▶ Traslación de un vector p_{xyz} seguida de una rotación un ángulo Φ sobre el eje **OY**

$$T(p,(y,\Phi)) = \begin{pmatrix} C\Phi & -S\Phi & 0 & p_x C\Phi - p_z S\Phi \\ 0 & 1 & 0 & p_y \\ -S\Phi & 0 & C\Phi & p_z C\Phi + p_x S\Phi \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



Localización espacial de un cuerpo regido

- **Matriz de transformación Homogénea: Traslación seguida de rotación**

▶ Traslación de un vector p_{xyz} seguida de una rotación un ángulo θ sobre el eje **OZ**

$$T(p,(z, \theta)) = \begin{pmatrix} C\theta & -S\theta & 0 & p_x C\theta - p_y S\theta \\ S\theta & C\theta & 0 & p_x S\theta + p_y C\theta \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Ejemplo

Un sistema **OUVW** ha sido girado 90° alrededor del eje **OX** y posteriormente es trasladado un vector $p(8, -4, 12)$ con respecto al sistema **OXYZ**. Calcular las coordenadas (r_x, r_y, r_z) del vector **r** con coordenadas **ruvw** $(-3, 4, -11)$.

$$T((x,\alpha),p) = \begin{pmatrix} r_x \\ r_y \\ r_z \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 8 \\ 0 & 0 & -1 & -4 \\ 0 & 1 & 0 & 12 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} -3 \\ 4 \\ -11 \\ 1 \end{pmatrix} = \begin{pmatrix} 5 \\ 7 \\ 16 \\ 1 \end{pmatrix}$$



Localización espacial de un cuerpo regido

● Composición de matrices homogéneas

- ▶ Las matrices homogéneas se pueden **componer para describir diversos giros y traslaciones consecutivos** sobre un sistema de referencia determinado.
- ▶ Así, una transformación compleja podrá descomponerse en la aplicación consecutiva de **transformaciones simples** (giros y traslaciones).
- ▶ Por ejemplo un giro de un ángulo α sobre el eje OX, seguido de un giro de un ángulo Φ sobre el eje OY y de un ángulo θ sobre el eje OZ se representa como:

$$T = R(z,\theta)R(y,\Phi)R(x,\alpha) = \begin{pmatrix} C\theta & -S\theta & 0 & 0 \\ S\theta & C\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C\Phi & 0 & S\Phi & 0 \\ 0 & 1 & 0 & 0 \\ -S\Phi & 0 & C\Phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha & -S\alpha & 0 \\ 0 & S\alpha & C\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

giro θ sobre OZ

← giro Φ sobre OY

← giro α sobre OX



Localización espacial de un cuerpo regido

- Composición de matrices homogéneas

$$T = R(z,\theta)R(y,\Phi)R(x,\alpha) = \begin{pmatrix} C\theta C\Phi & -S\theta C\alpha + C\theta S\Phi S\alpha & S\theta S\alpha + C\theta S\Phi C\alpha & 0 \\ S\theta C\Phi & C\theta C\alpha + S\theta S\Phi S\alpha & -C\theta S\alpha + S\theta S\Phi C\alpha & 0 \\ -S\Phi & C\Phi S\alpha & C\Phi C\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Ejemplo

Obtener la matriz de transformación para las siguientes transformaciones sobre el sistema fijo OXYZ de referencia: Traslación de un vector $p_{xyz}(-3,10,10)$; giro de -90° sobre el eje O'U del sistema trasladado y giro de 90° sobre el eje OV' del sistema girado.

$$T = T(p) T(U, -90^\circ) T(V, 90^\circ) = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & 10 \\ 0 & 0 & 1 & 10 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 10 \\ 0 & 0 & -1 & -10 \\ -1 & 0 & 0 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

giro 90° sobre OV'

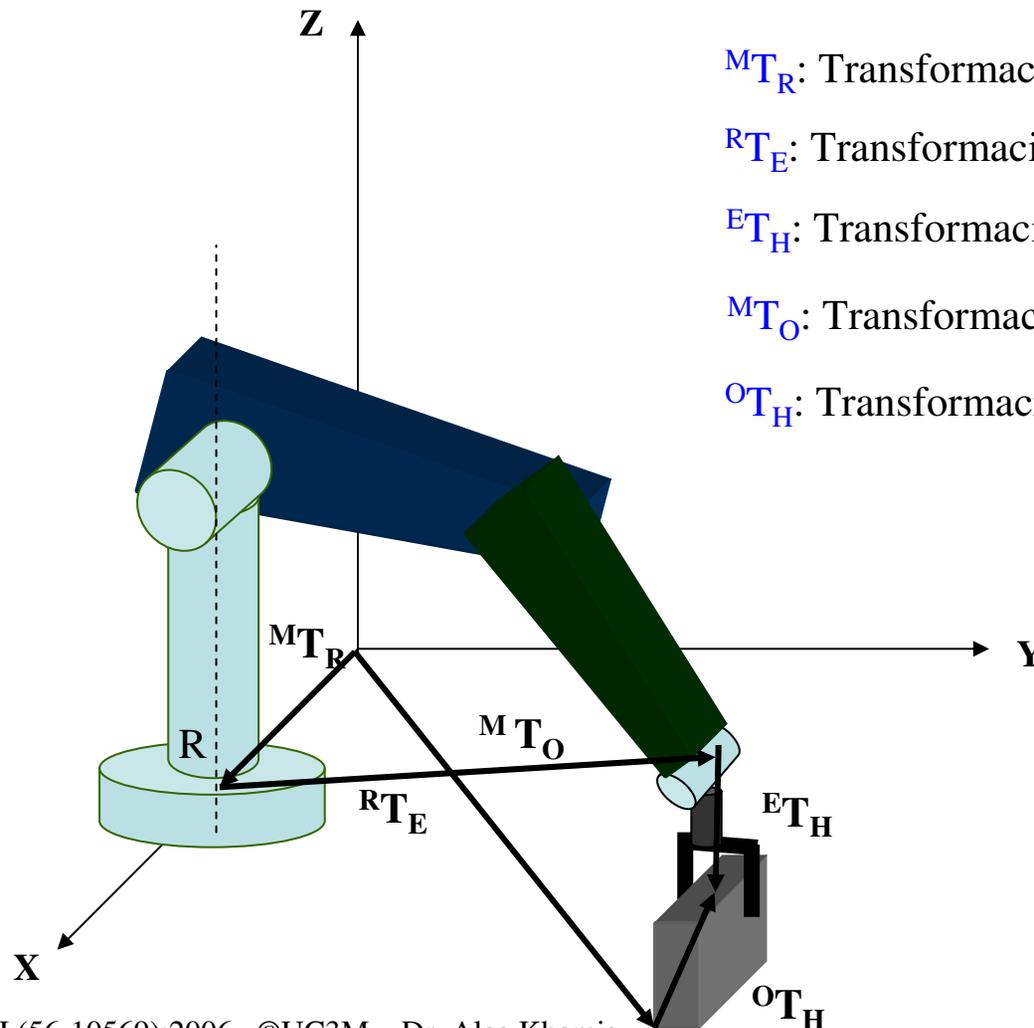
giro -90° sobre O'U

Traslación



Localización espacial de un cuerpo rígido

● Gráficos de transformación



M_{T_R} : Transformación - referencia → base del manipulador

R_{T_E} : Transformación - extremo → referencia del manipulador

E_{T_H} : Transformación - herramienta → extremo

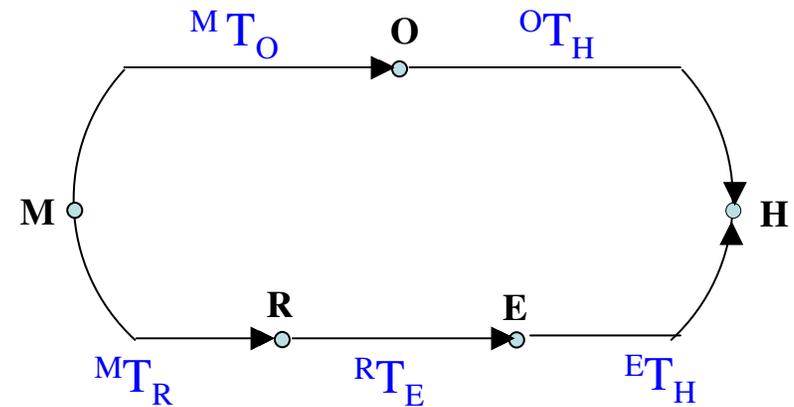
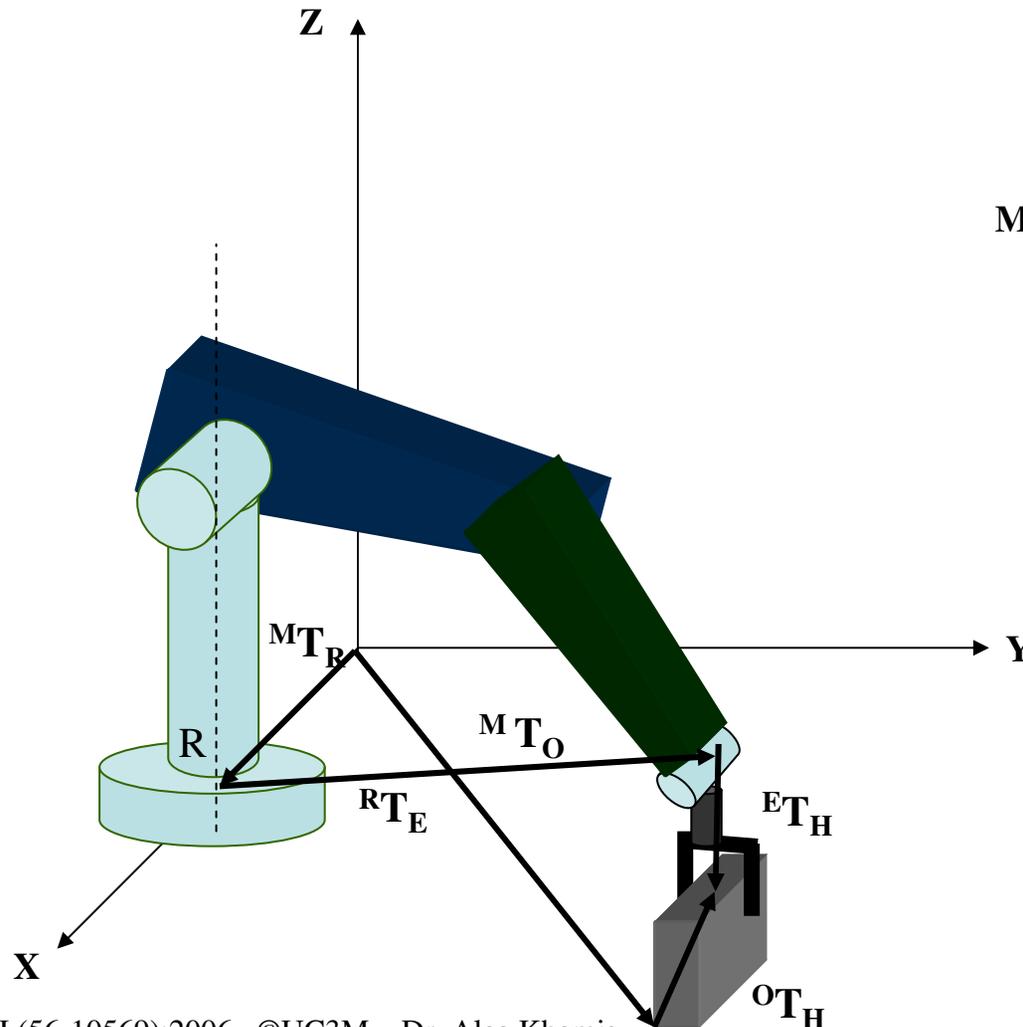
M_{T_O} : Transformación - objeto → base

O_{T_H} : Transformación - herramienta → objeto

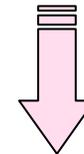


Localización espacial de un cuerpo rígido

● Gráficos de transformación



$$M^T_R \ R^T_E \ E^T_H = M^T_O \ O^T_H$$



$$R^T_O = R^T_E \ E^T_H \ (O^T_H)^{-1}$$

O bien por

$$R^T_O = (M^T_R)^{-1} M^T_O$$



Automatización Industrial - II

56 – 10569 2° Cuatrimestre 2006

Clase 5 – Miércoles 3 de Mayo 2006

Cinemática - I



Automatización Industrial - II

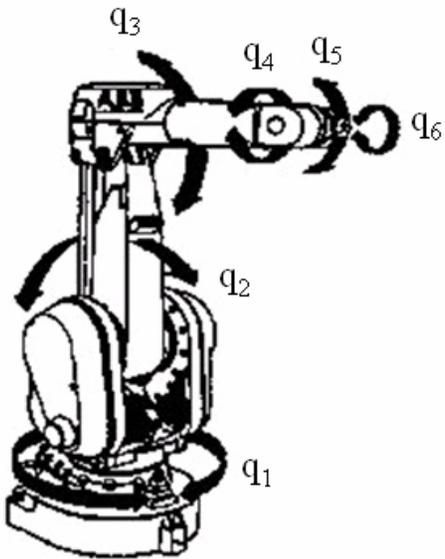
56 – 10569 2º Cuatrimestre 2006

Clase 6 – Jueves 4 de Mayo 2006

Cinemática - II



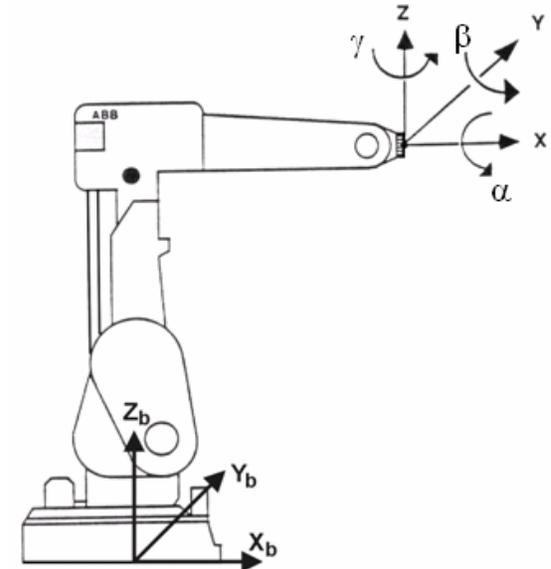
Cinemática directa



Dado: Largos y configuración de cada eslabón. El ángulo de cada articulación

Determinar: Posición y orientación del extremo del robot.

Cinemática Directa →



Sabiendo:

- q_1
- q_2
- q_3
- q_4
- q_5
- q_6

Se piden:

$$\begin{aligned}
 x &= f_x (q_1, q_2, q_3, q_4, q_5, q_6) \\
 y &= f_y (q_1, q_2, q_3, q_4, q_5, q_6) \\
 z &= f_z (q_1, q_2, q_3, q_4, q_5, q_6) \\
 \alpha &= f_\alpha (q_1, q_2, q_3, q_4, q_5, q_6) \\
 \beta &= f_\beta (q_1, q_2, q_3, q_4, q_5, q_6) \\
 \gamma &= f_\gamma (q_1, q_2, q_3, q_4, q_5, q_6)
 \end{aligned}$$



Cinemática directa

● Robot plano con 2-GDL de rotación

► Situación:

Se tiene un brazo robótico que parte alineado con el eje x_0 .

El primer eslabón gira q_1 y el segundo q_2 .

► Pregunta:

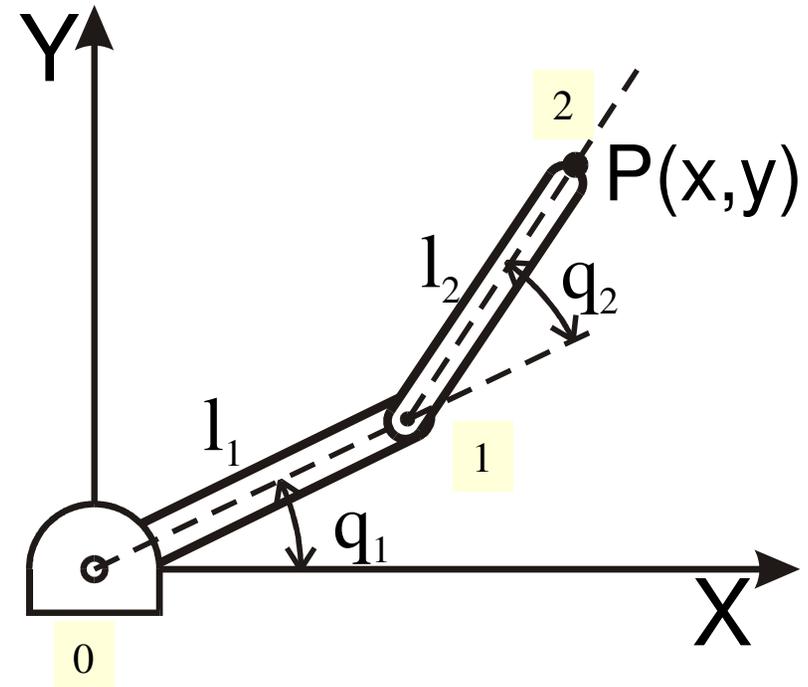
¿Cual es la posición del extremo del brazo?

► Solución:

Enfoque geométrico

$$x = l_1 \cdot \cos q_1 + l_2 \cos(q_1 + q_2)$$

$$y = l_1 \cdot \text{sen} q_1 + l_2 \text{sen}(q_1 + q_2)$$



Enfoque algebraico

$${}^0A_2 = {}^0A_1 {}^1A_2$$

Implica transformaciones de coordenadas



Cinemática directa

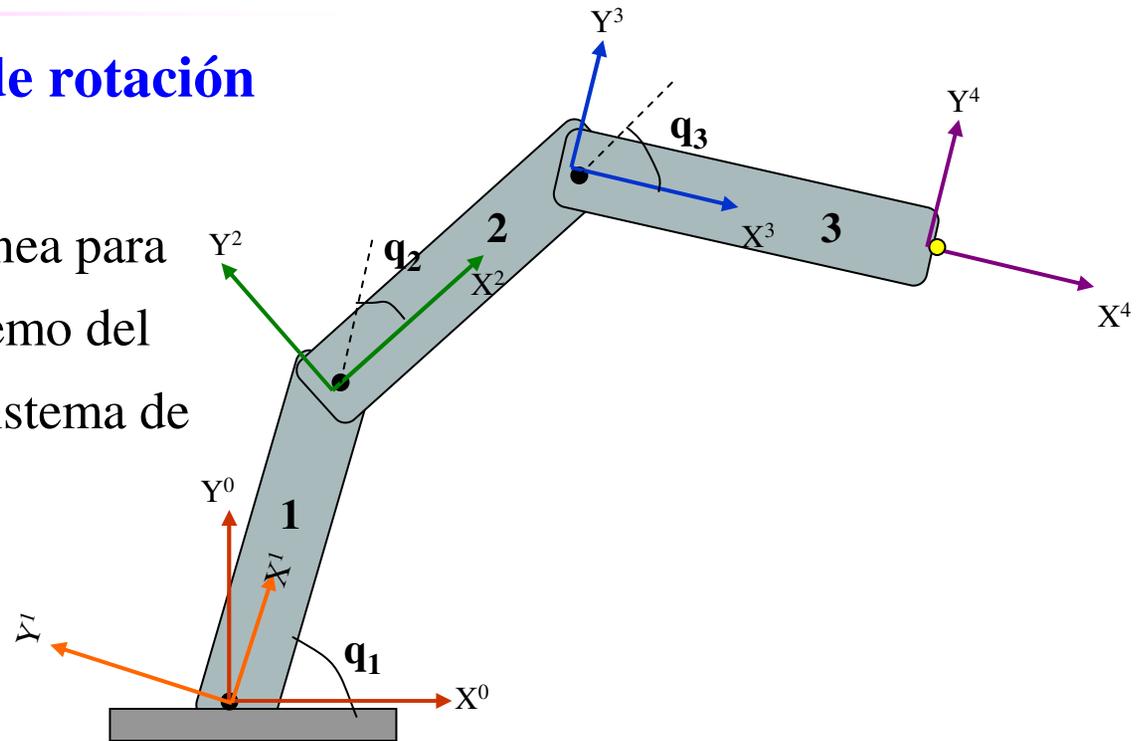
● Robot plano con 3-GDL de rotación

► Se pide:

Determinar la matriz homogénea para encontrar la posición del extremo del brazo (punto amarillo) en el sistema de referencia X₀Y₀.

l_1, l_2, l_3 : largos de los eslabones 1, 2, 3 respectivamente.

► Solución:



${}^0A_4 =$	0A_1	1A_2	2A_3	3A_4
	Rotación (q_1)	Rotación (q_2) seguida traslación (l_1)	Rotación (q_3) seguida traslación (l_2)	traslación (l_3)



Cinemática directa

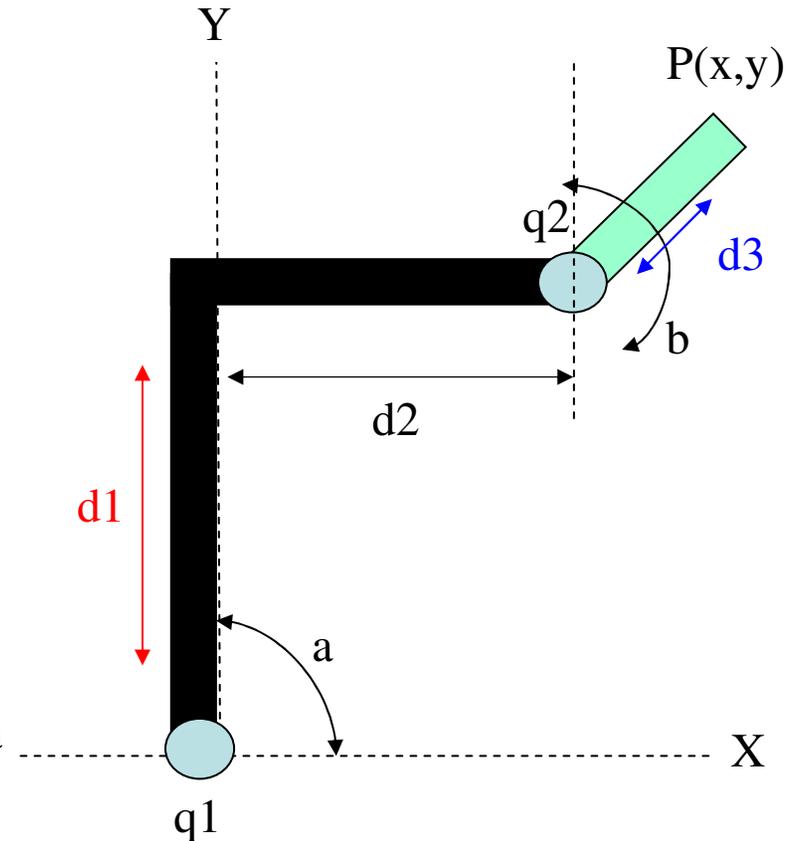
● Cuestión 9, Examen AI-II 2004-2005

► Obtenga las expresiones de la cinemática directa para el extremo $p(x,y)$ del robot de la figura donde $d1$, $d2$ y $d3$ son conocidos y las articulaciones $q1$ y $q2$ cuentan con codificadores angulares de posición que suministran los ángulos a y b respectivamente, $d2$ es una barra fija soldada perpendicularmente al extremo de $d1$.

► Solución:

$$X = d1 \cos a + d3 \cos b + d2 \cos (-90 + a)$$

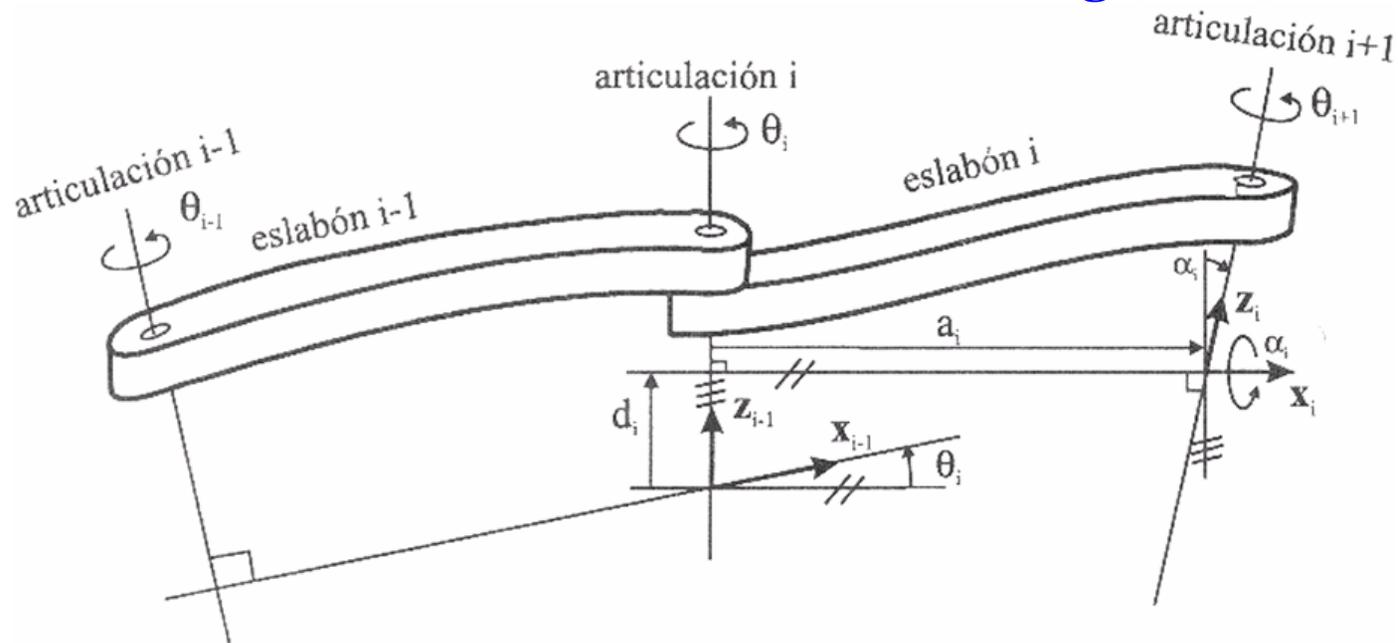
$$Y = d2 \sin a + d3 \sin b + d1 \sin (-90 + a)$$





Cinemática directa

- **Método sistemático: Notación Denavit-Hartenberg (1955)**

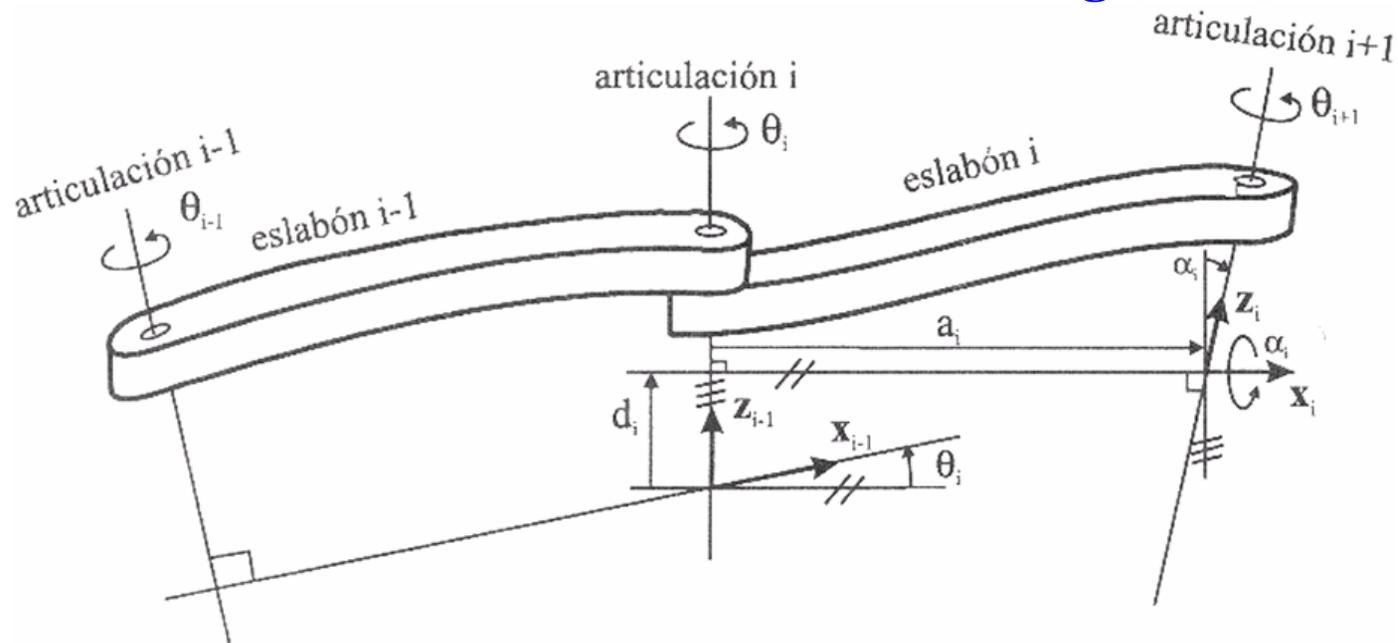


- ▶ **IDEA:** A cada articulación se asigna un sistema de coordenadas de acuerdo al sistema D-H. Es posible relacionar el sistema (i) con el sistema ($i - 1$) mediante **4 transformaciones** básicas que dependen exclusivamente de las características geométricas del eslabón.



Cinemática directa

- **Método sistemático: Notación Denavit-Hartenberg (1955)**



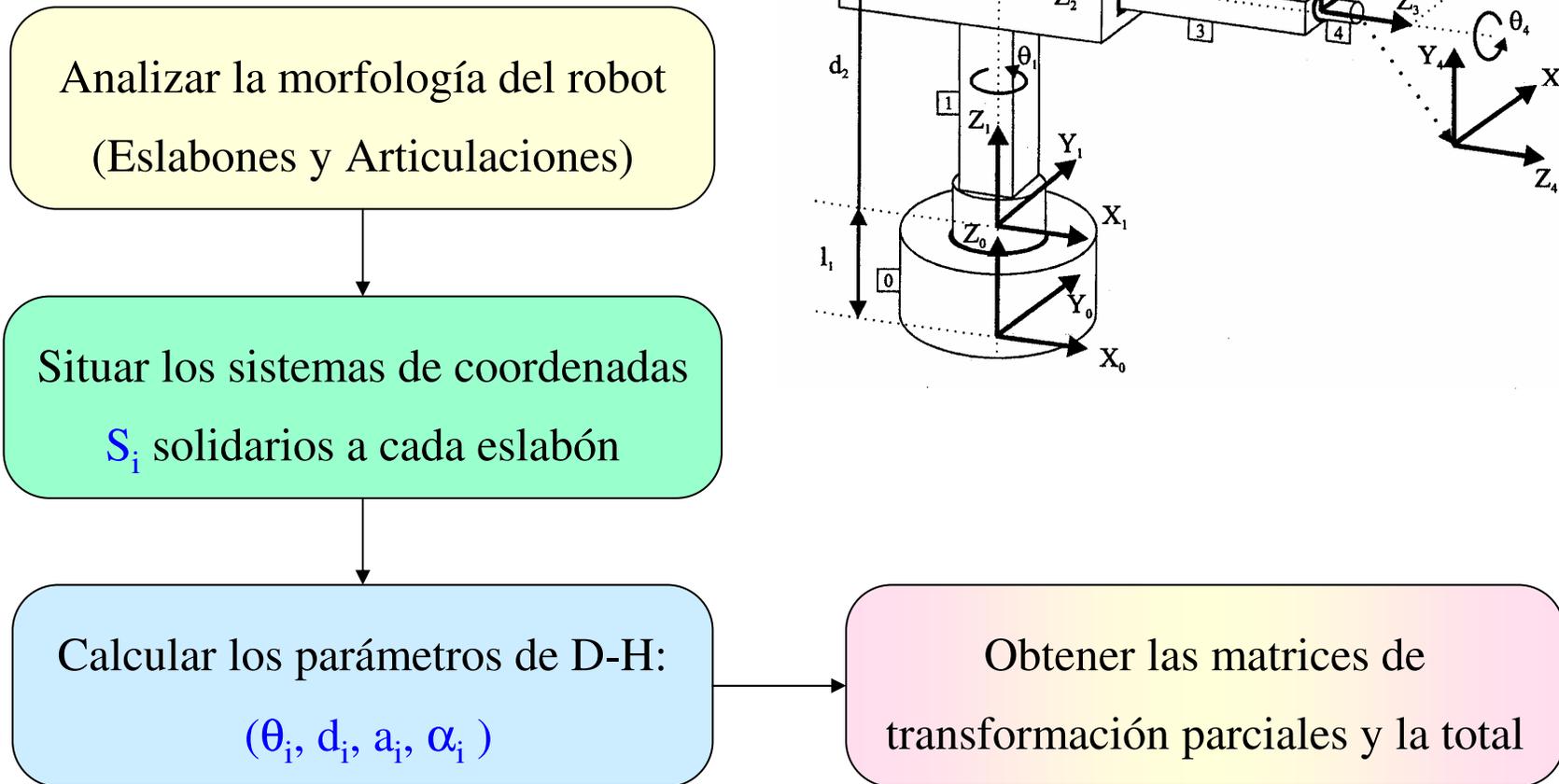
Parámetros de
Denavit-Hartenberg

- Rotación alrededor del eje z_{i-1} un ángulo θ_i
- Traslación a lo largo de z_{i-1} una distancia d_i
- Traslación a lo largo de x_i una distancia a_i
- Rotación alrededor del eje x_i un ángulo α_i



Cinemática directa

● Algoritmo de Denavit-Hartenberg

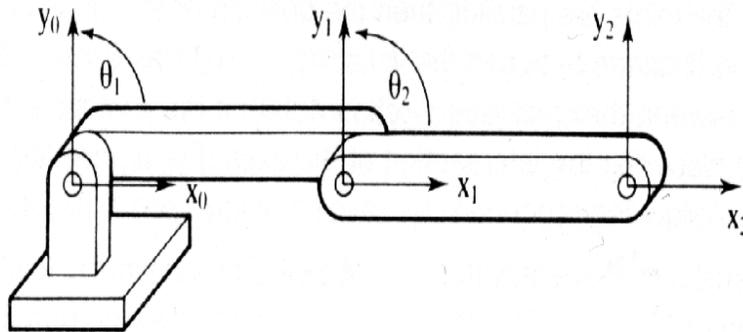




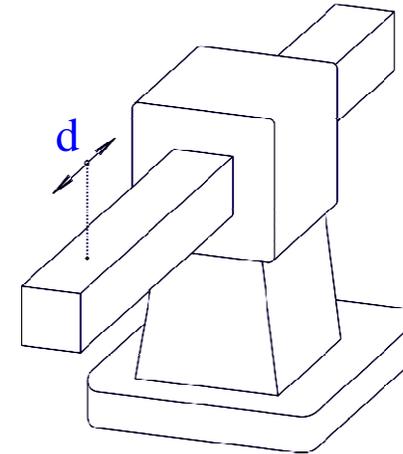
Cinemática directa

- **Algoritmo de Denavit-Hartenberg:** Analizar la morfología del robot

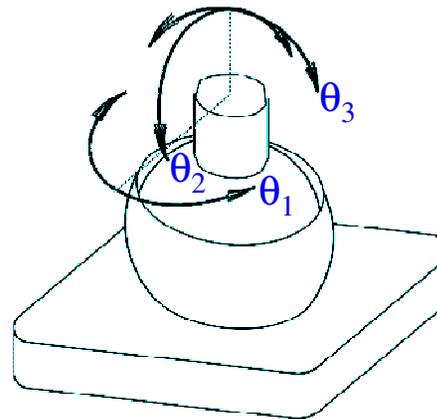
► Articulaciones básicas



Articulación rótula
1 DOF (Variable - θ)



Articulación deslizante
1 DOF (linear) (Variables - d)

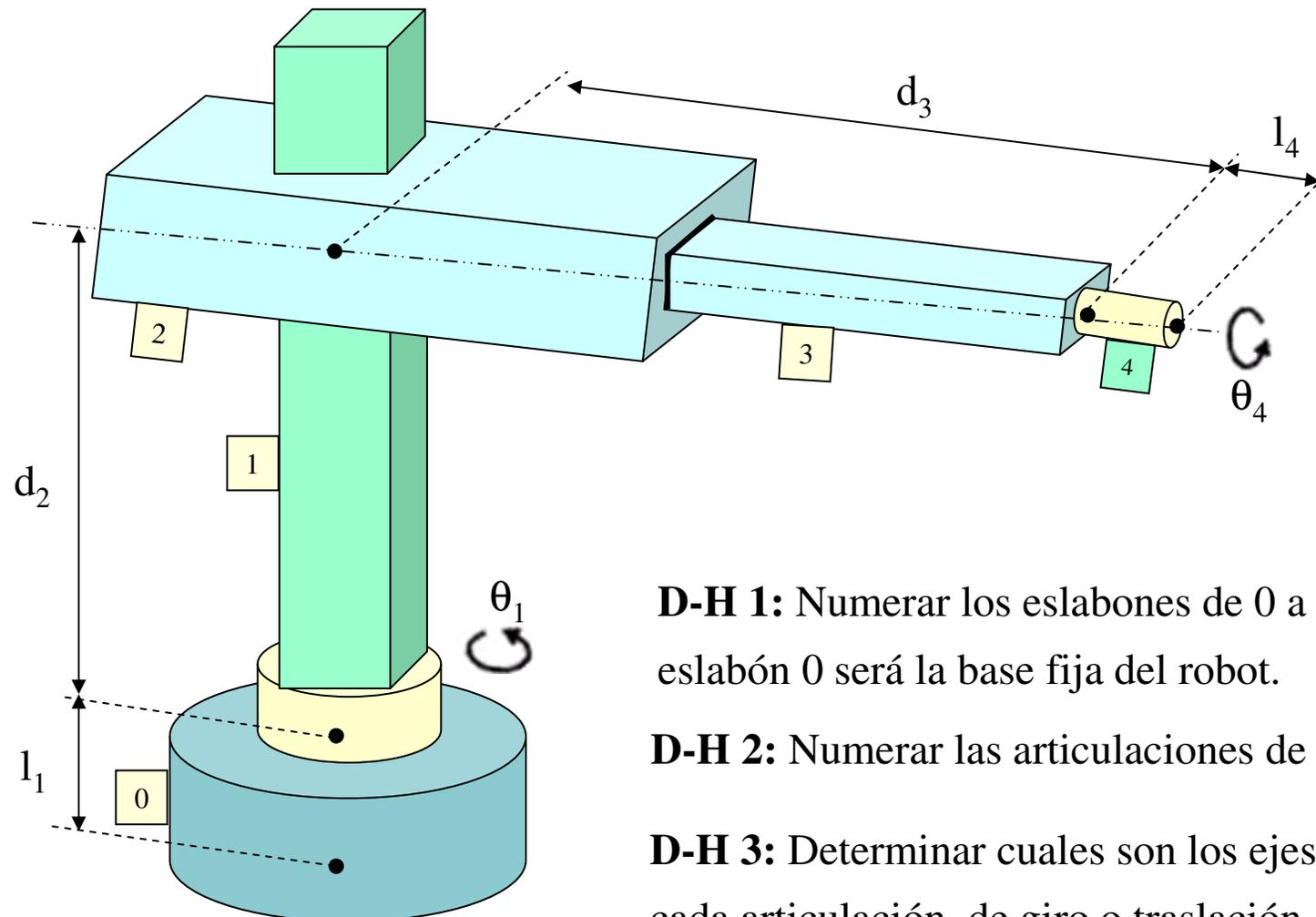


Articulación esférica
3 DOF (Variables - $\theta_1, \theta_2, \theta_3$)



Cinemática directa

- **Algoritmo de Denavit-Hartenberg: Analizar la morfología del robot**



D-H 1: Numerar los eslabones de 0 a N. El eslabón 0 será la base fija del robot.

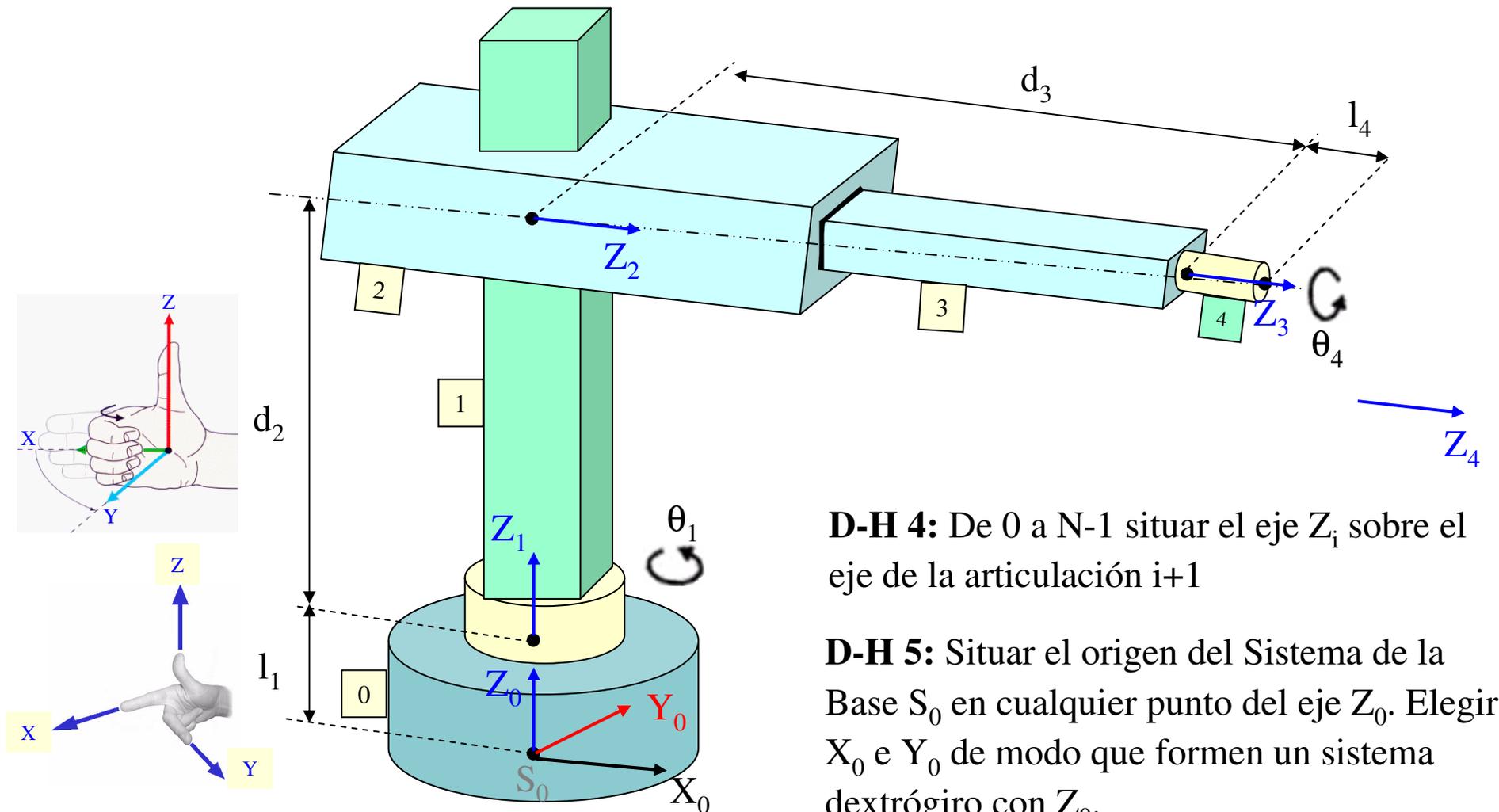
D-H 2: Numerar las articulaciones de 1 a N

D-H 3: Determinar cuales son los ejes de cada articulación, de giro o traslación.



Cinemática directa

- **Algoritmo de Denavit-Hartenberg: Situar los sistemas de coordenadas**



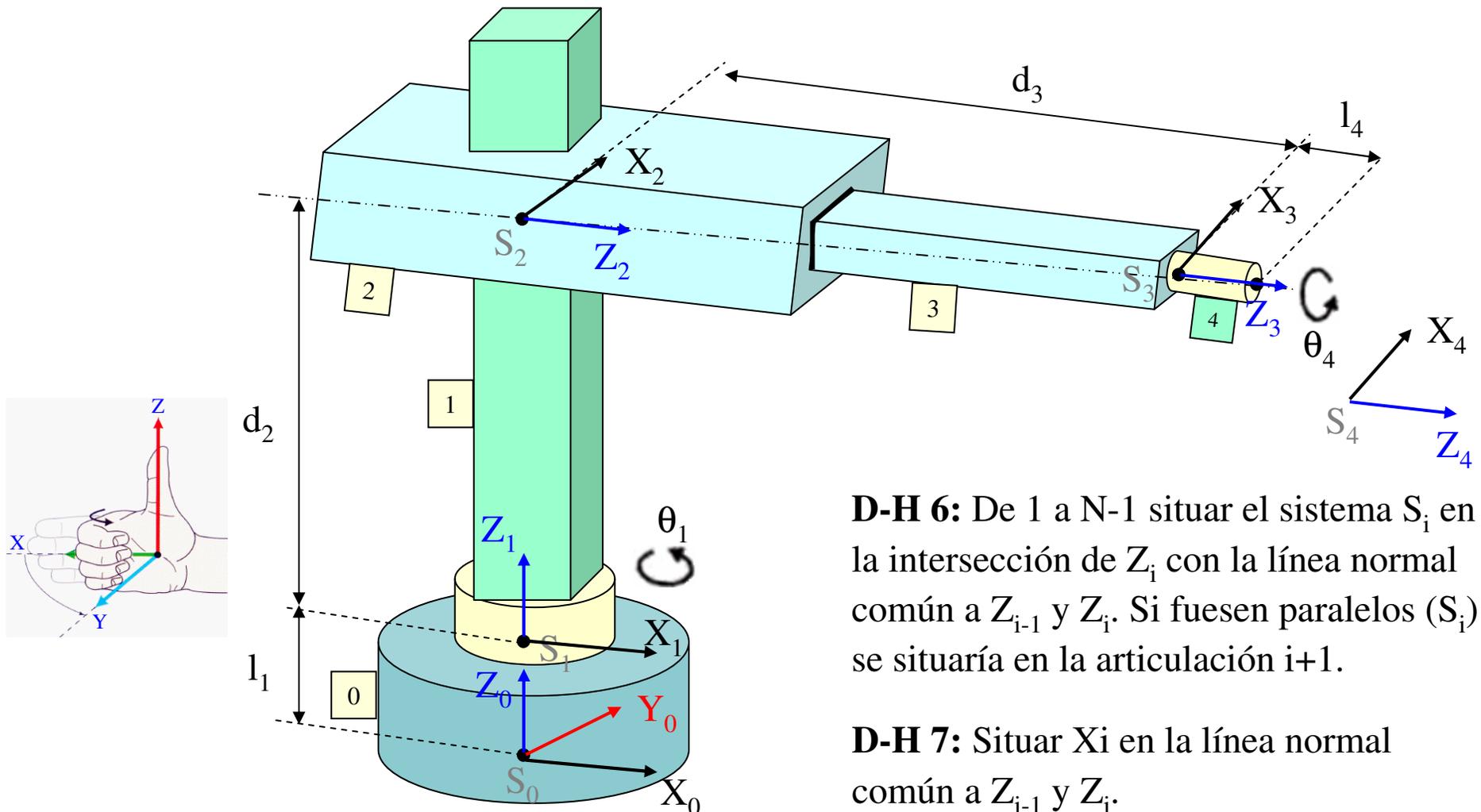
D-H 4: De 0 a N-1 situar el eje Z_i sobre el eje de la articulación $i+1$

D-H 5: Situar el origen del Sistema de la Base S_0 en cualquier punto del eje Z_0 . Elegir X_0 e Y_0 de modo que formen un sistema dextrógiro con Z_0 .



Cinemática directa

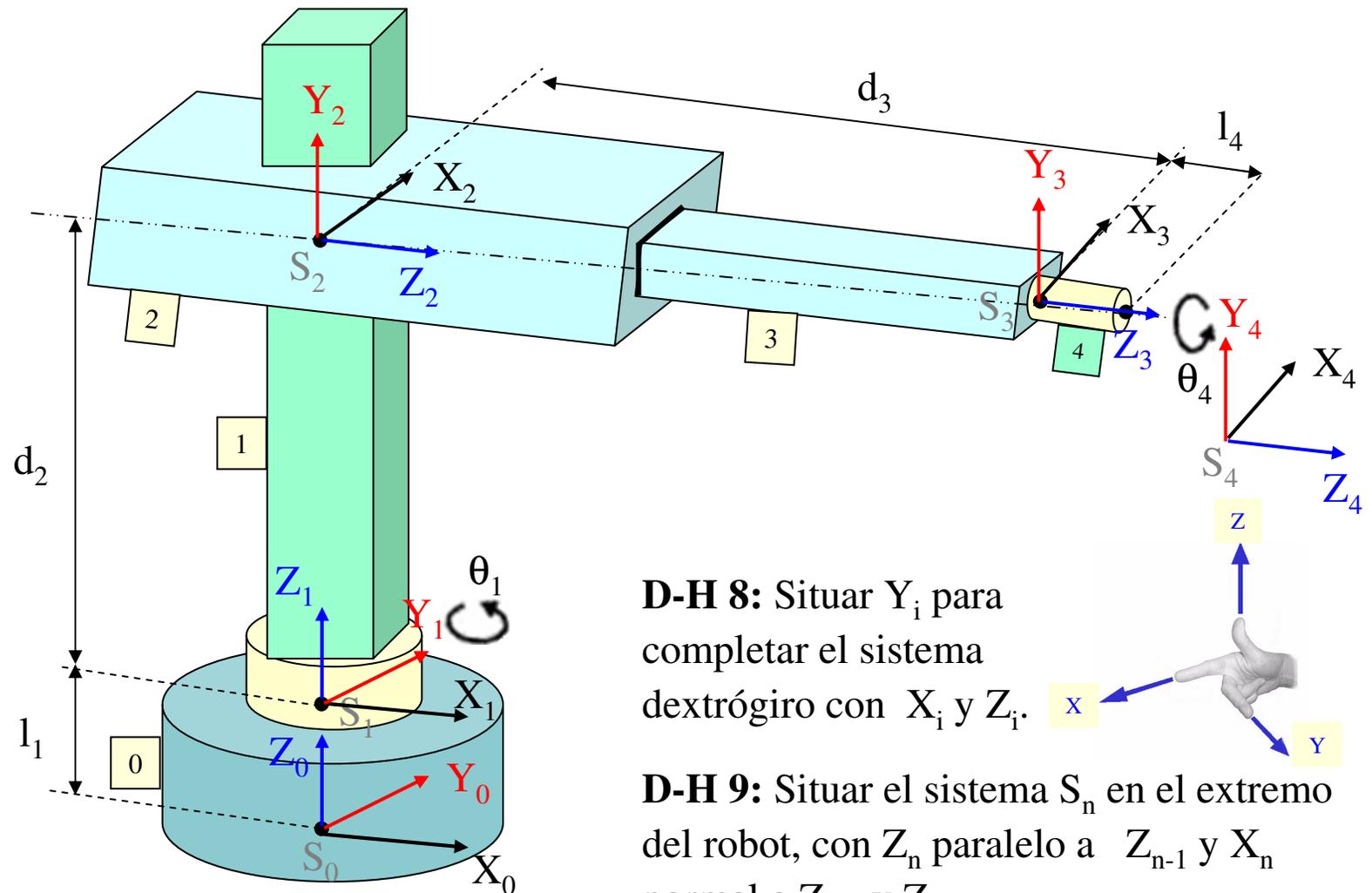
- **Algoritmo de Denavit-Hartenberg: Situar los sistemas de coordenadas**





Cinemática directa

- **Algoritmo de Denavit-Hartenberg: Situar los sistemas de coordenadas**



D-H 8: Situar Y_i para completar el sistema dextrógiro con X_i y Z_i .

D-H 9: Situar el sistema S_n en el extremo del robot, con Z_n paralelo a Z_{n-1} y X_n normal a Z_{n-1} y Z_n .

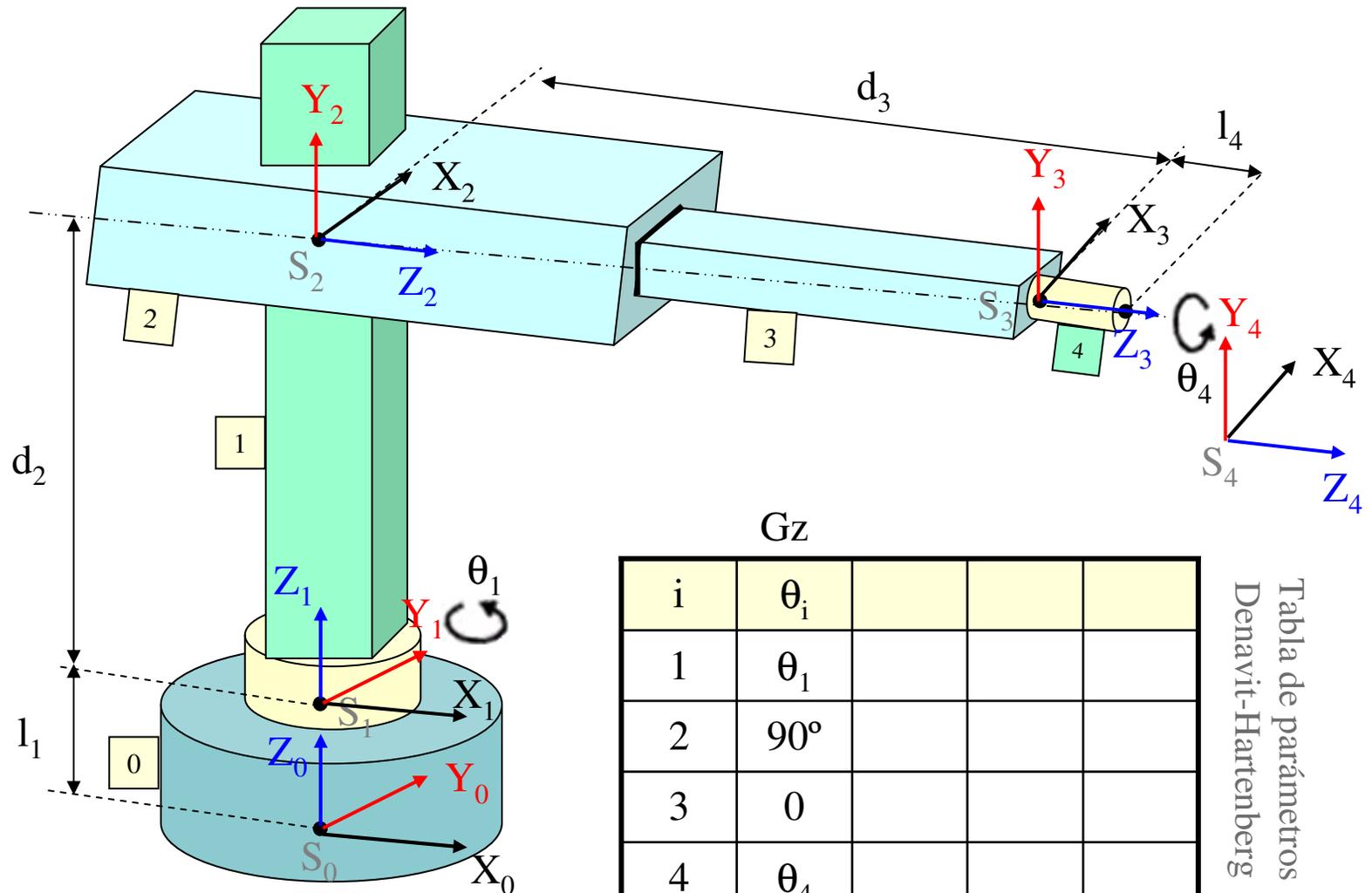


Cinemática directa

- **Algoritmo de Denavit-Hartenberg: Obtener los parámetros de D-H**

θ_i

Es el giro en torno a Z_{i-1} para que X_{i-1} y X_i queden paralelos.





Cinemática directa

- **Algoritmo de Denavit-Hartenberg: Obtener los parámetros de D-H**

d_i

El desplazamiento a lo largo de Z_{i-1} para que X_{i-1} y X_i queden alineados

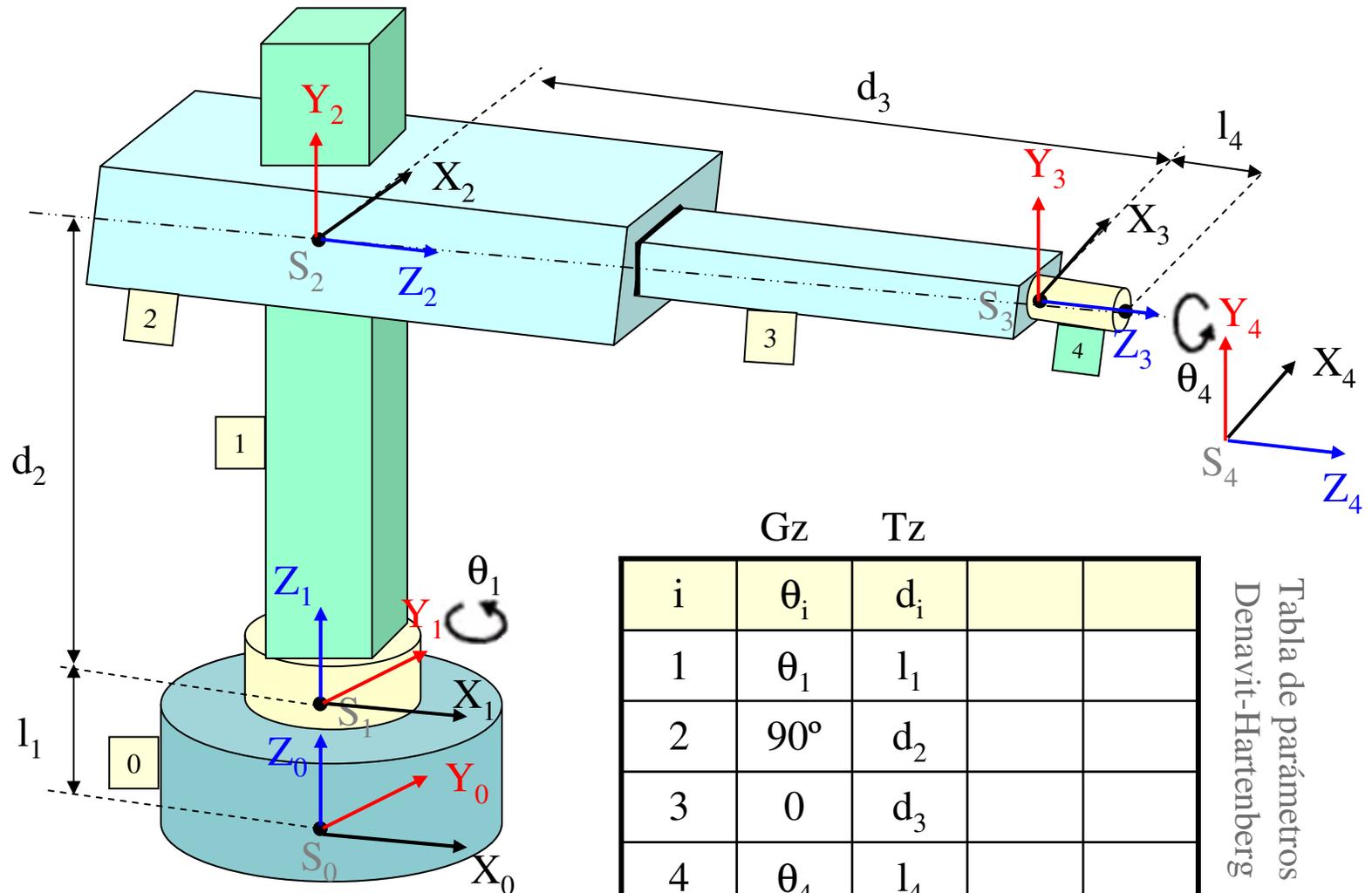


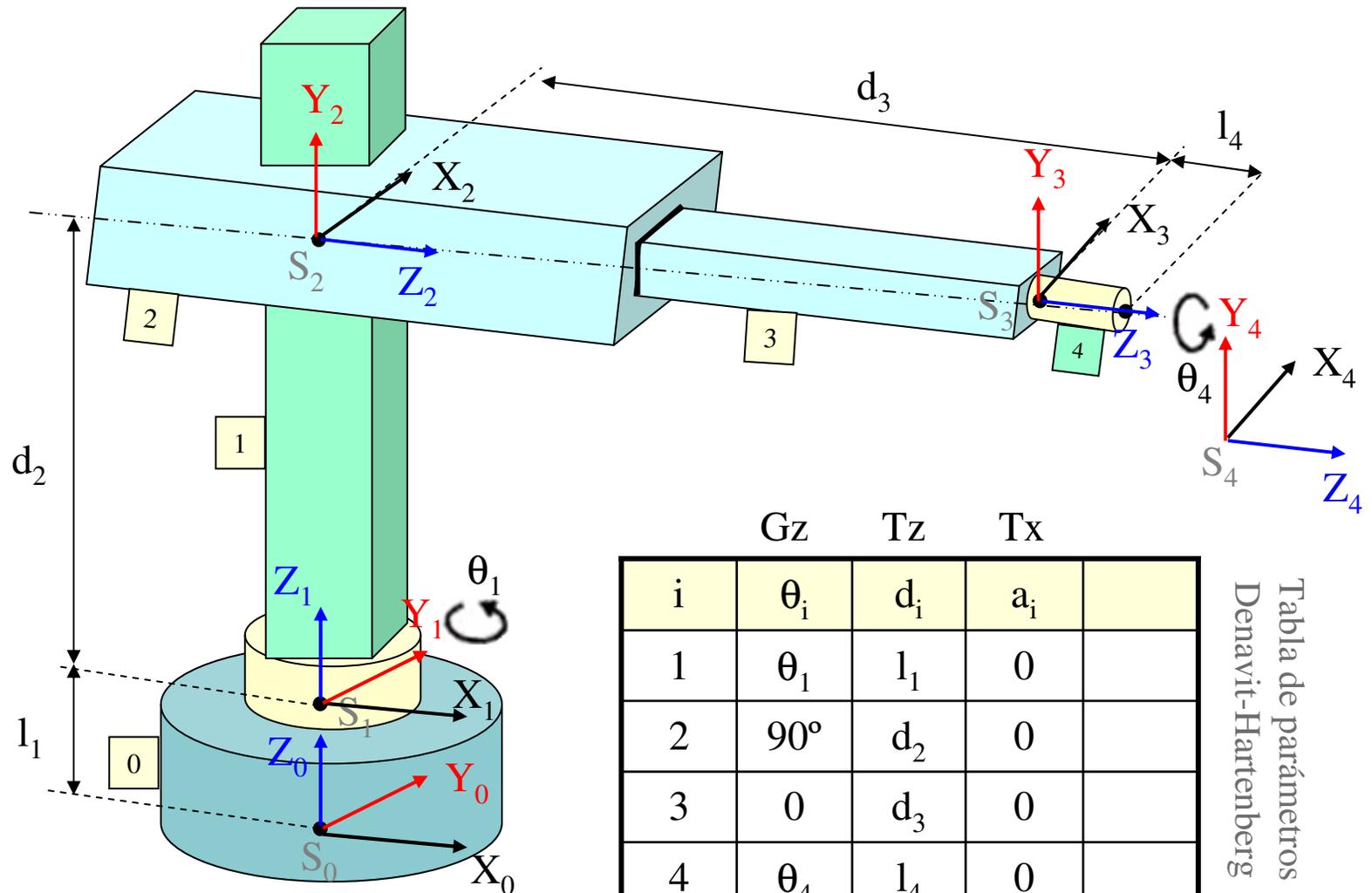
Tabla de parámetros Denavit-Hartenberg



Cinemática directa

- **Algoritmo de Denavit-Hartenberg: Obtener los parámetros de D-H**

a_i
Es el desplazamiento a lo largo de X_i para que coincidan los orígenes de S_{i-1} y S_i



	Gz	Tz	Tx	
i	θ_i	d_i	a_i	
1	θ_1	l_1	0	
2	90°	d_2	0	
3	0	d_3	0	
4	θ_4	l_4	0	

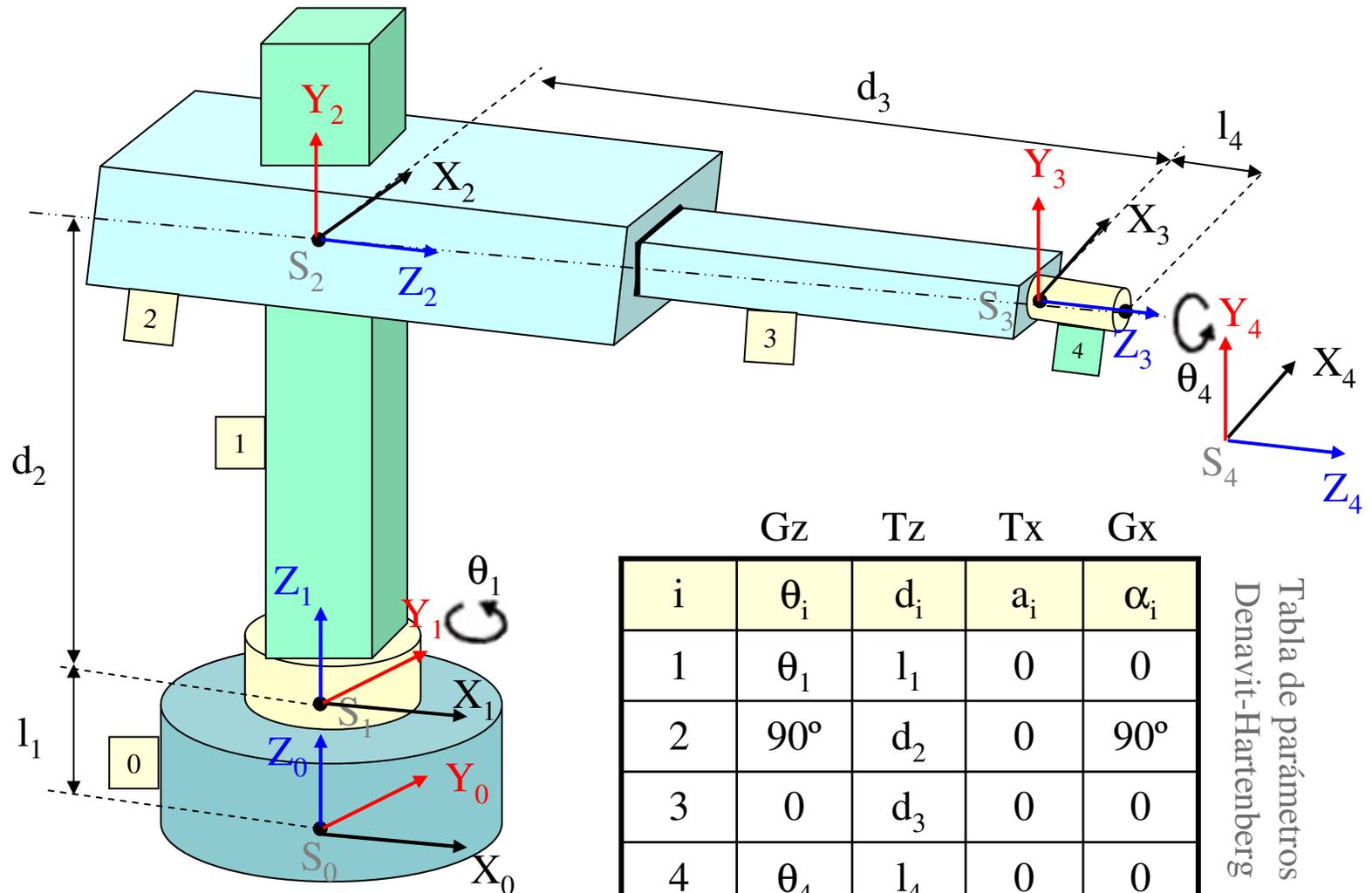
Tabla de parámetros Denavit-Hartenberg



Cinemática directa

- **Algoritmo de Denavit-Hartenberg: Obtener los parámetros de D-H**

α_i
Es el giro en torno a X_i para que coincidan los sistemas S_{i-1} y S_i



	Gz	Tz	Tx	Gx
i	θ_i	d_i	a_i	α_i
1	θ_1	l_1	0	0
2	90°	d_2	0	90°
3	0	d_3	0	0
4	θ_4	l_4	0	0

Tabla de parámetros Denavit-Hartenberg



Cinemática directa

- **Algoritmo de Denavit-Hartenberg: Obtener las matrices de transformación homogénea**

Parciales

$${}^{i-1}A_i = \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Total

$$T_n = {}^0A_n = {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3 \cdot \dots \cdot {}^{(n-1)}A_n$$



Cinemática directa

- **Algoritmo de Denavit-Hartenberg: Obtener las matrices de transformación homogénea**

Parciales

$${}^0A_1 = \begin{bmatrix} C\theta_1 & -S\theta_1 & 0 & 0 \\ S\theta_1 & C\theta_1 & 0 & 0 \\ 0 & 0 & 1 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1A_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^2A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3A_4 = \begin{bmatrix} C\theta_4 & -S\theta_4 & 0 & 0 \\ S\theta_4 & C\theta_4 & 0 & 0 \\ 0 & 0 & 1 & l_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

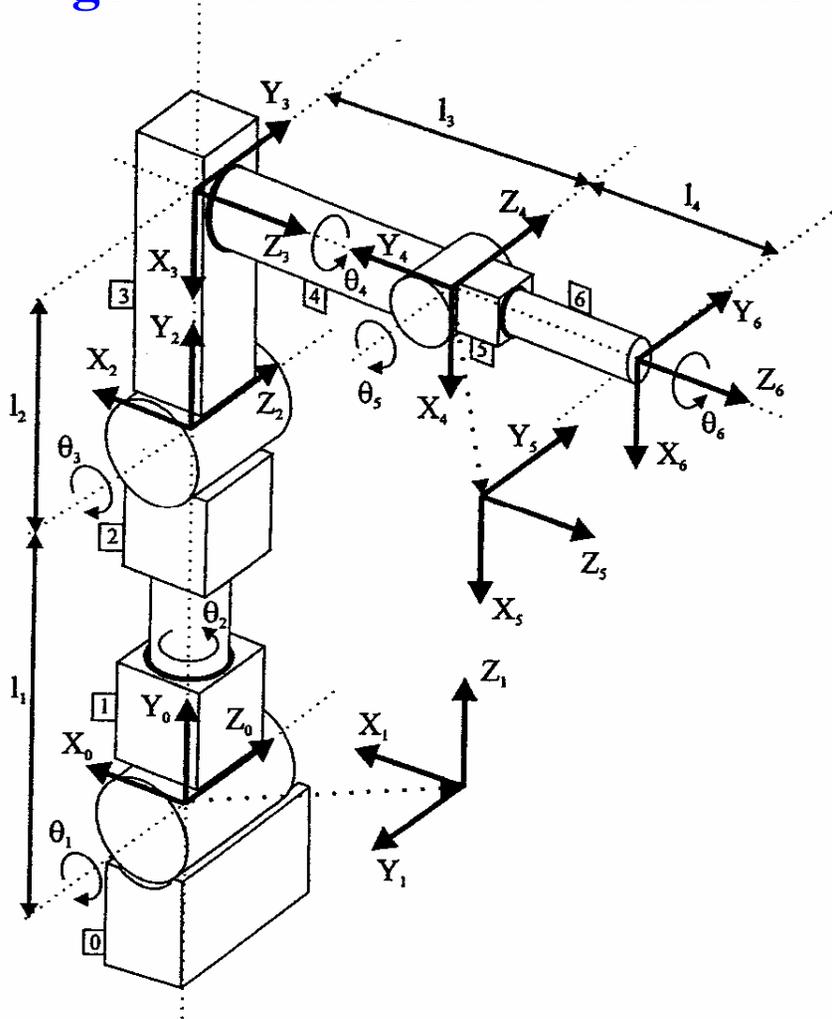
Total $T = {}^0A_4 = {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3 \cdot {}^3A_4$

$$T = {}^0A_4 = \begin{bmatrix} -S_1C_4 & S_1S_4 & C_1 & C_1(d_3 + l_4) \\ C_1C_4 & -C_1S_4 & S_1 & S_1(d_3 + l_4) \\ S_4 & C_4 & 0 & d_2 + l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Cinemática directa

● Algoritmo de Denavit-Hartenberg: Ejemplos – Robot IRB6400C



	Gz	Tz	Tx	Gx
i	θ_i	d_i	a_i	α_i
1	θ_1	0	0	-90
2	θ_2	l_1	0	90
3	$\theta_3 - 90$	0	$-l_2$	90
4	θ_4	l_3	0	-90
5	θ_5	0	0	90
6	θ_6	l_4	0	0

Denavit-Hartenberg
Tabla de parámetros



Cinemática directa

● Algoritmo de Denavit-Hartenberg: Ejemplos – Robot IRB6400C

$${}^0A_1 = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1A_2 = \begin{bmatrix} C_2 & 0 & -S_2 & 0 \\ S_2 & 0 & C_2 & 0 \\ 0 & -1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

i	θ_i	d_i	a_i	α_i
1	θ_1	0	0	-90
2	θ_2	l_1	0	90
3	θ_3-90	0	$-l_2$	90
4	θ_4	l_3	0	-90
5	θ_5	0	0	90
6	θ_6	l_4	0	0

$${}^2A_3 = \begin{bmatrix} S_3 & 0 & -C_3 & -l_2S_3 \\ -C_3 & 0 & -S_3 & l_2C_3 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3A_4 = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & l_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4A_5 = \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ S_5 & 0 & -C_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^5A_6 = \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 1 & 0 & l_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Cinemática directa

● Algoritmo de Denavit-Hartenberg: Ejemplos – Robot IRB6400C

$$\mathbf{T}^{=0} \mathbf{A}_1^1 \mathbf{A}_2^2 \mathbf{A}_3^3 \mathbf{A}_4^4 \mathbf{A}_5^5 \mathbf{A}_6^6 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Extremo del robot

(P_x, P_y, P_z) : posición
 (n, o, a) : orientación

$$n_x = (C_1 C_2 S_3 + S_1 C_3)(C_4 C_5 C_6 - S_4 S_6) + C_1 S_2 (S_4 C_5 C_6 + C_4 S_6) + (-C_1 C_2 C_3 + S_1 S_3) S_5 C_6$$

$$n_y = (-S_1 C_2 S_3 + S_1 C_3)(C_4 C_5 C_6 - S_4 S_6) + S_1 S_2 (S_4 C_5 C_6 + C_4 S_6) + (-S_1 C_2 C_3 - C_1 S_3) S_5 C_6$$

$$n_z = (-S_2 S_3)(C_4 C_5 C_6 - S_4 S_6) + C_2 (S_4 C_5 C_6 + C_4 S_6) + S_2 C_3 S_5 C_6$$

$$o_x = (C_1 C_2 S_3 + S_1 C_3)(-C_4 C_5 C_6 - S_4 S_6) + C_1 S_2 (-S_4 C_5 C_6 + C_4 S_6) + (-C_1 C_2 C_3 + S_1 S_3)(-S_5 C_6)$$

$$o_y = (-S_1 C_2 S_3 + S_1 C_3)(-C_4 C_5 C_6 - S_4 S_6) + S_1 S_2 (-S_4 C_5 C_6 + C_4 S_6) + (-S_1 C_2 C_3 - C_1 S_3)(-S_5 C_6)$$

$$o_z = (-S_2 S_3)(-C_4 C_5 C_6 - S_4 S_6) + C_2 (-S_4 C_5 C_6 + C_4 S_6) + S_2 C_3 (-S_5 C_6)$$

$$p_x = (C_1 C_2 S_3 + S_1 C_3)(l_4 C_4 S_5) + C_1 S_2 (l_4 S_4 S_5) + (C_1 C_2 C_3 + S_1 S_3)(-l_4 C_5 + l_3) + (-l_2 C_1 C_2 S_3 - l_2 S_1 C_3 - l_1 S_1)$$

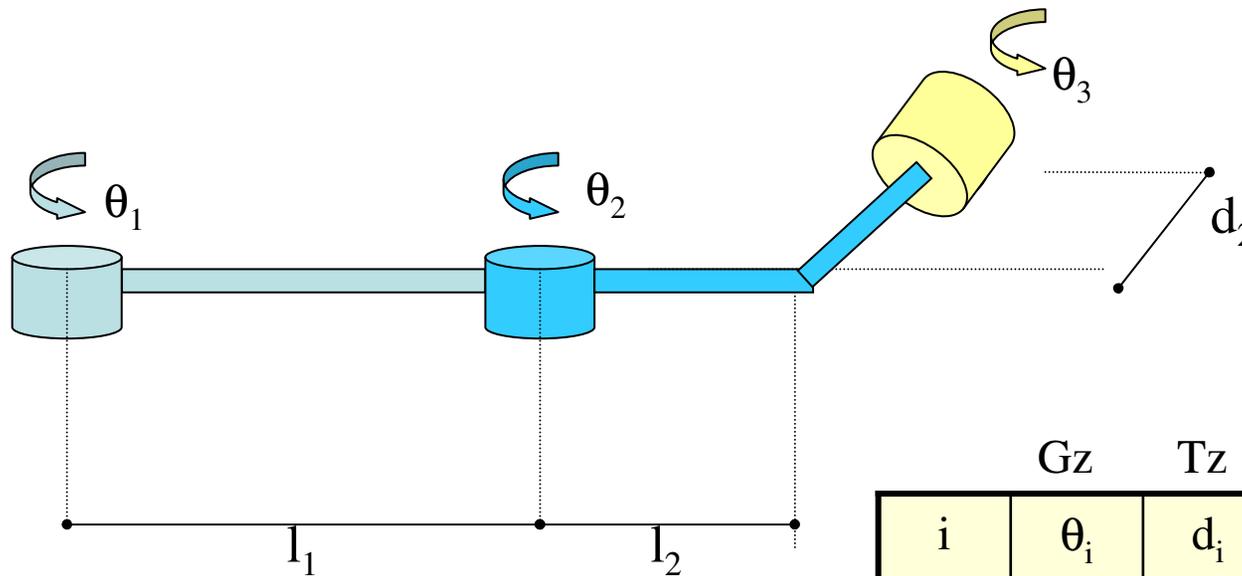
$$p_y = (-S_1 C_2 S_3 - C_1 C_3)(l_4 C_4 S_5) + S_1 S_2 (l_4 S_4 S_5) + (-C_1 C_2 C_3 - C_1 S_3)(-l_4 C_5 + l_3) + (-l_2 S_1 C_2 S_3 - l_2 C_1 C_3 + l_1 C_1)$$

$$p_z = (-S_2 S_3)(l_4 C_4 C_5) + C_2 (l_4 S_4 S_5) + S_2 C_3 (-l_4 C_5 + l_3) + l_2 S_2 S_3$$



Cinemática directa

- Algoritmo de Denavit-Hartenberg: Ejemplos - 3 ejes de rotación



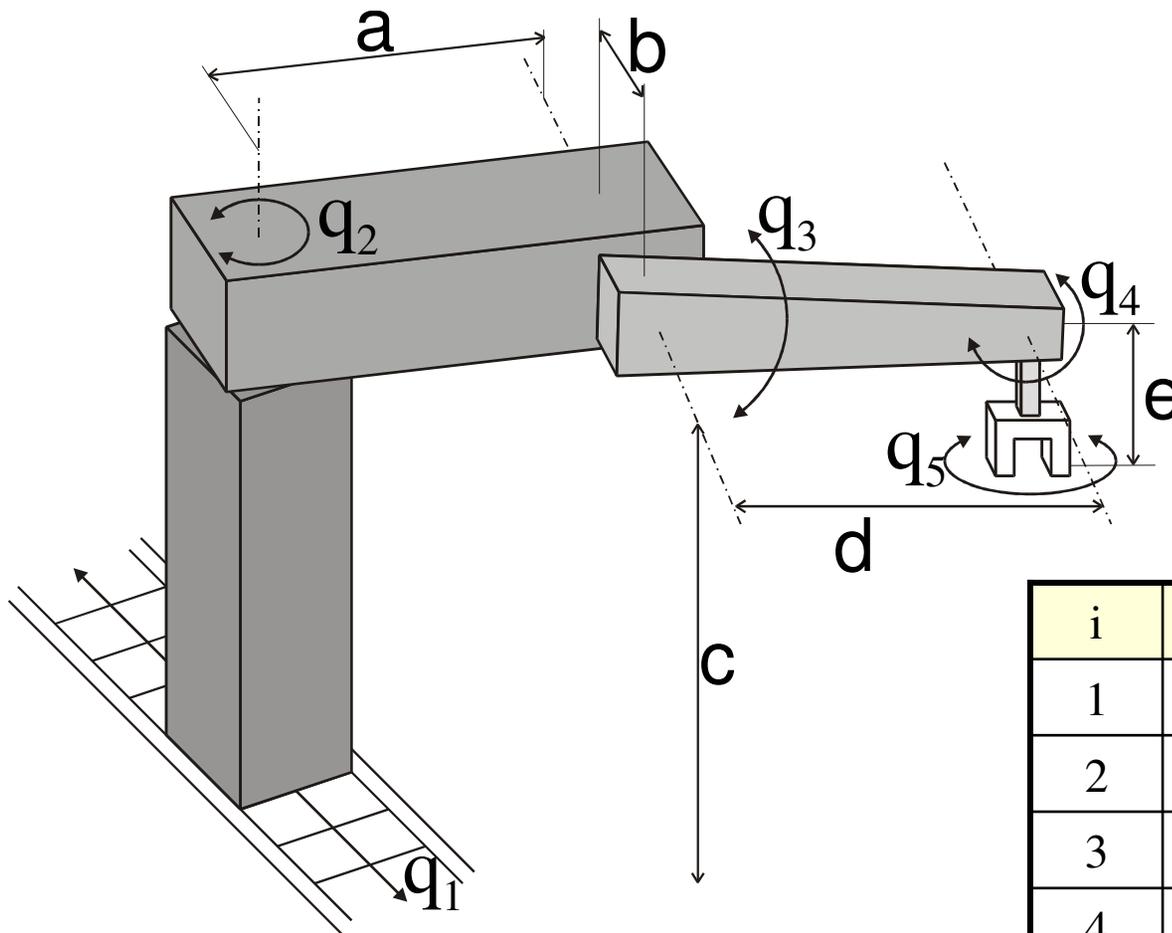
	Gz	Tz	Tx	Gx
i	θ_i	d_i	a_i	α_i
1				
2				
3				

Tabla de parámetros
Denavit-Hartenberg



Cinemática directa

- Algoritmo de Denavit-Hartenberg: Ejemplos – Robot con 5 GDL



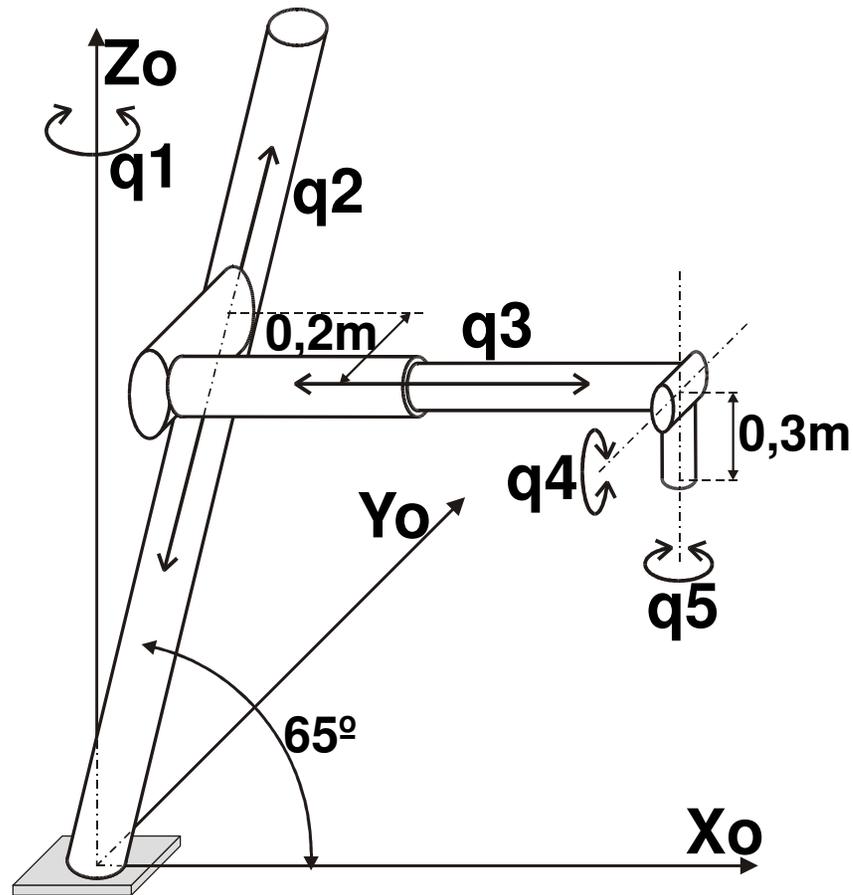
	Gz	Tz	Tx	Gx
i	θ_i	d_i	a_i	α_i
1				
2				
3				
4				
5				

Tabla de parámetros
Denavit-Hartenberg
24/64



Cinemática directa

- Algoritmo de Denavit-Hartenberg: Ejemplos – Robot con 5 GDL



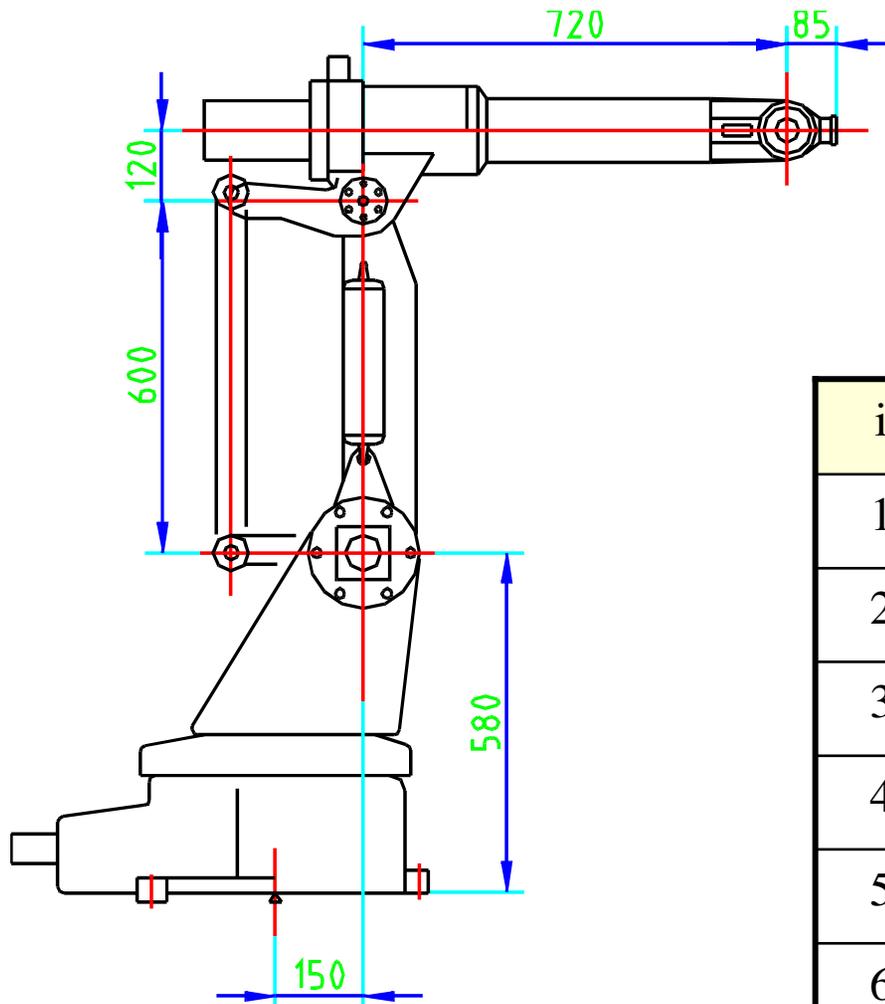
	Gz	Tz	Tx	Gx
i	θ_i	d_i	a_i	α_i
1				
2				
3				
4				
5				

Tabla de parámetros
Denavit-Hartenberg
25/64



Cinemática directa

- Algoritmo de Denavit-Hartenberg: Ejemplos – Robot con 6 GDL



	Gz	Tz	Tx	Gx
i	θ_i	d_i	a_i	α_i
1				
2				
3				
4				
5				
6				

Denavit-Hartenberg
Tabla de parámetros



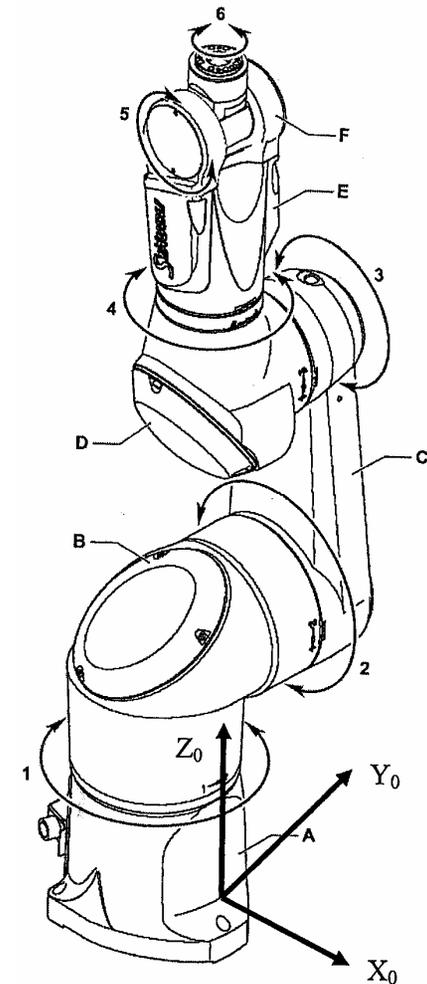
Cinemática directa

● Algoritmo de Denavit-Hartenberg: Ejemplos – Robot STAUBLI Parte D, Examen AI-II 2004-2005

► Para el robot de la figura siguiente, un robot

STAUBLI:

- Situar los sistemas de coordenadas de cada eslabón de acuerdo al algoritmo de Denavit-Hartenberg
- Obtener la tabla más simple de los parámetros de Denavit-Hartenberg del robot.
- Obtener la matriz 1A_2 que refleja el cambio de sistema de referencia entre las articulaciones 1 y 2.



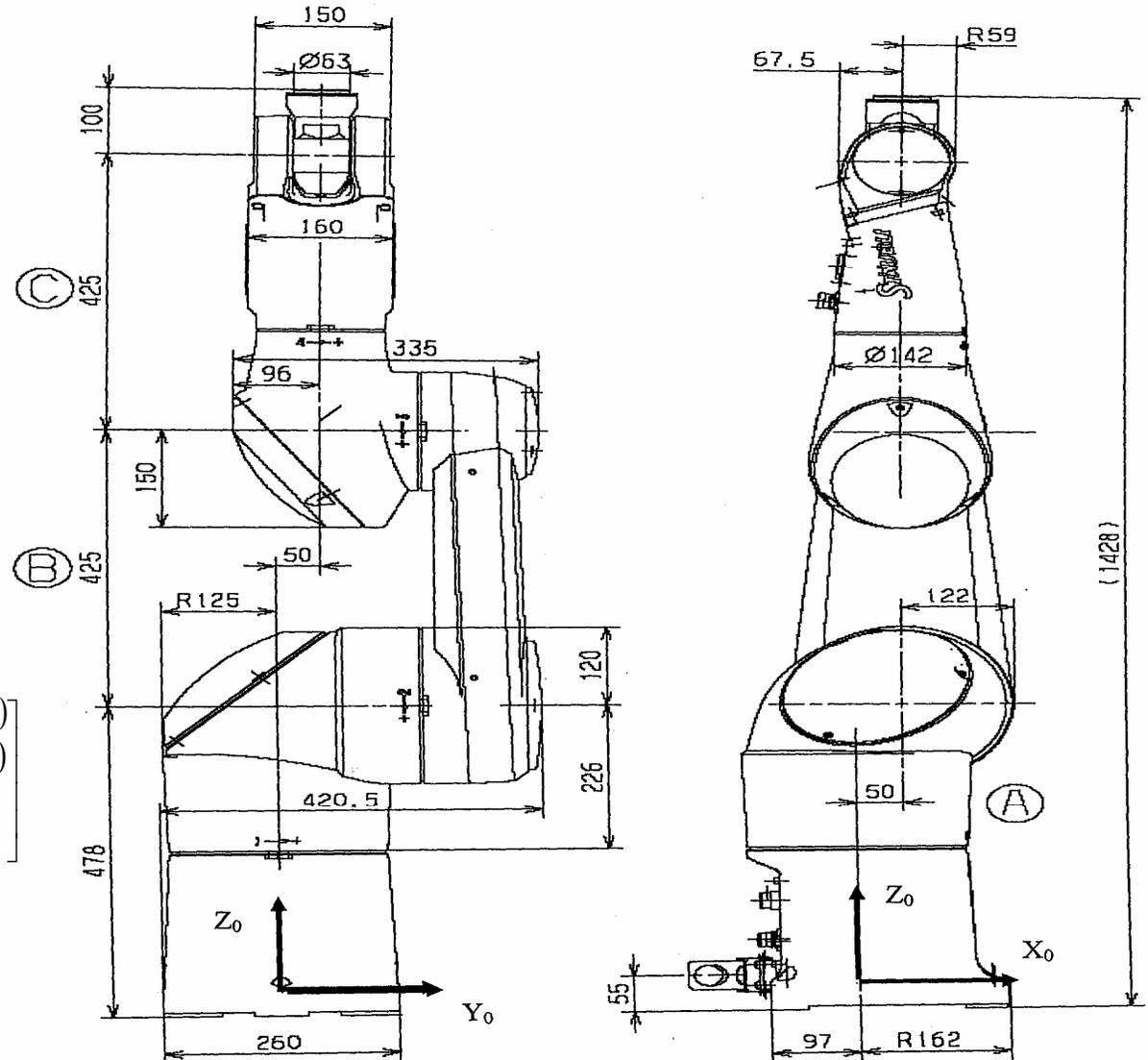


Cinemática directa

- **Algoritmo de Dt-H:**
Robot STAUBLI
Parte D
AI-II 2004-2005

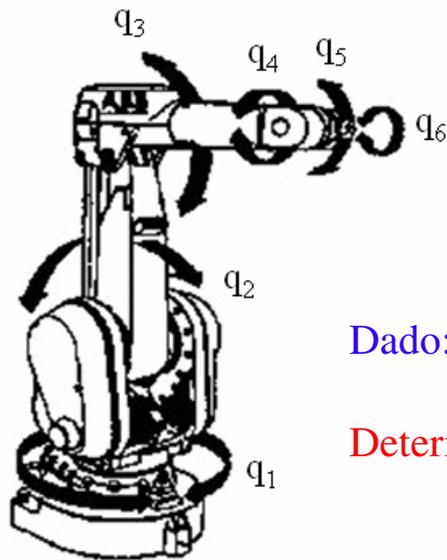
Solución

$${}^1A_2 = \begin{bmatrix} C(q_2+90) & -S(q_2+90) & 0 & 425C(q_2+90) \\ S(q_2+90) & C(q_2+90) & 0 & 425S(q_2+90) \\ 0 & 0 & 1 & -50 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



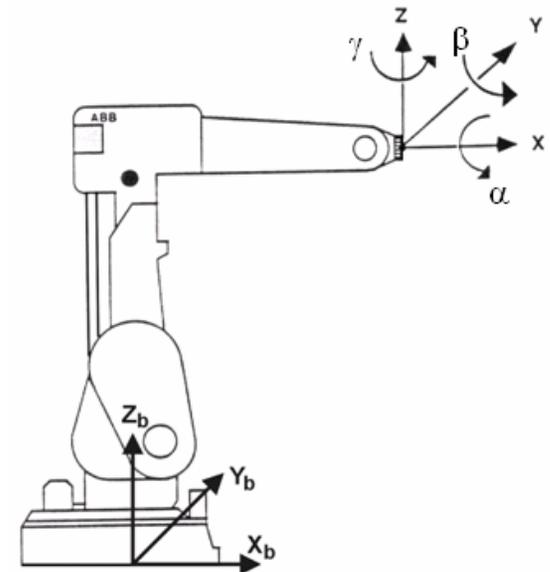


Cinemática inversa



Cinemática Inversa

Dado: Largos y configuración de cada eslabón.
 Posición y orientación del extremo del robot.
Determinar: Los ángulos de cada articulación necesarios para determinar aquella posición.



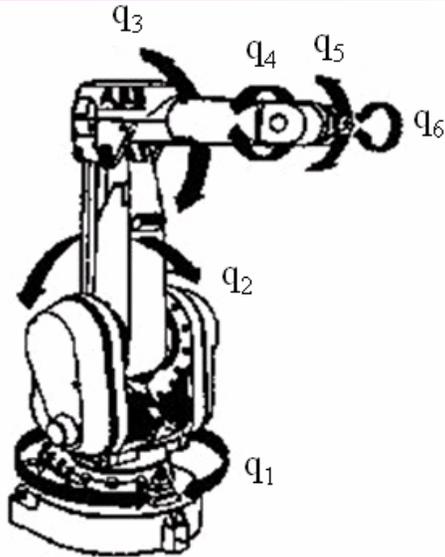
$x, y, z, \alpha, \beta, \gamma$

$$q_k = f_k(x, y, z, \alpha, \beta, \gamma)$$

$K = 1, \dots, n$ (n es GDL = 6)



Cinemática inversa

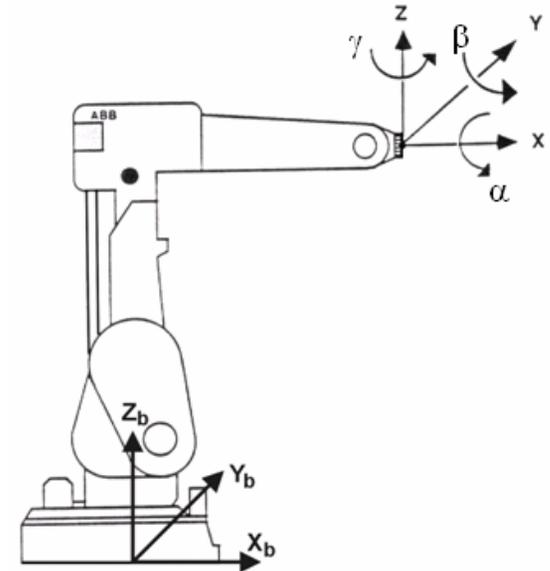


Cinemática Inversa

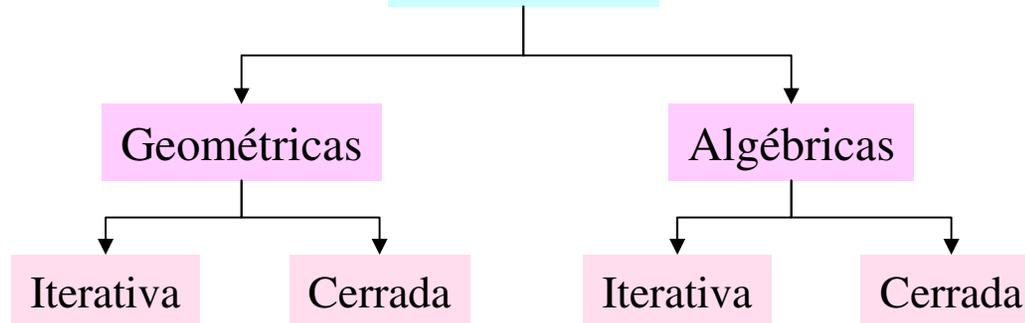
Objetivo: Encontrar los valores de las coordenadas articulares del robot para que su extremo adopte una determinada posición y orientación.

$$q_k = f_k(x, y, z, \alpha, \beta, \gamma)$$

$K = 1, \dots, n$ (n es GDL = 6)



Soluciones



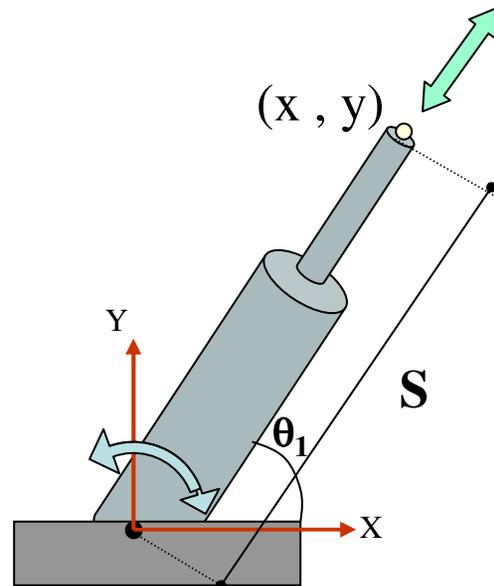
Iterativa: métodos genéricos independientes del tipo de robot

Cerrada: Relación matemática explícita



Cinemática inversa

- **Métodos geométricos: Un eje de rotula y uno deslizante**



Encontrar θ :

$$\theta = \arctan\left(\frac{y}{x}\right)$$

Largo S :

$$S = \sqrt{(x^2 + y^2)}$$



Cinemática inversa

● Métodos geométricos: Dos articulaciones de rótula

Dados: l_1, l_2, x, y

Encontrar: θ_1, θ_2

Redundante: No existe solución única

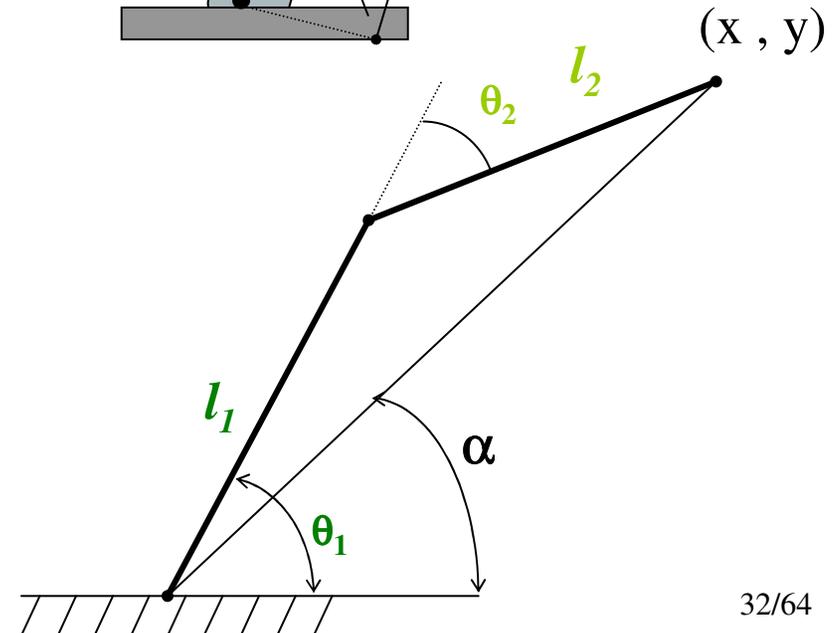
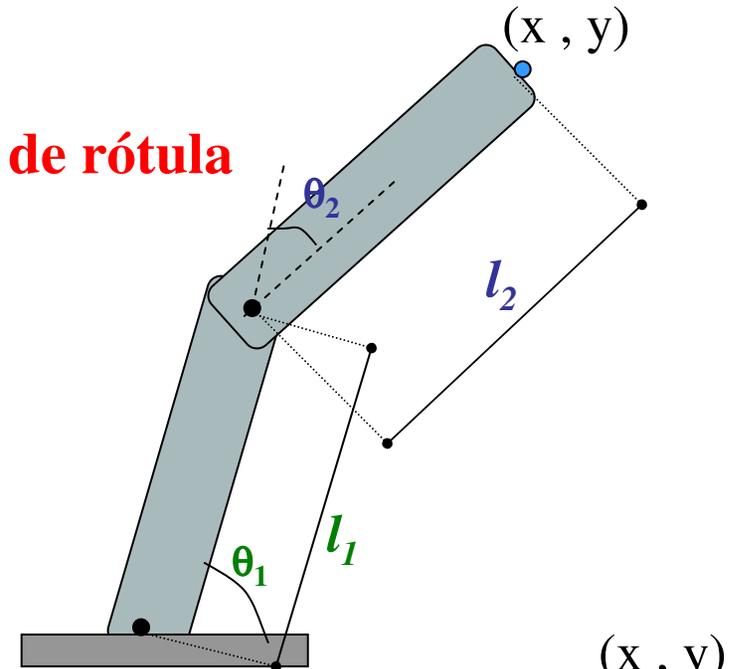
Con los datos existen dos soluciones:

$$\theta_2 = \arccos\left(\frac{x^2 + y^2 - l_1^2 - l_2^2}{2l_1l_2}\right)$$

Redundante, puesto que θ_2 puede estar en el primero o el cuarto cuadrante.

$$\theta_1 = \arcsen\left(\frac{l_2 \text{sen}(\theta_2)}{\sqrt{x^2 + y^2}}\right) + \arctan 2\left(\frac{y}{x}\right)$$

Redundante, puesto que θ_1 posee dos valores posibles.





Cinemática inversa

- **Métodos algebraicos**

▶ Resolver el sistema de ecuaciones:

$$X = f_x(q_1, \dots, q_n)$$

$$Y = f_y(q_1, \dots, q_n)$$

$$Z = f_z(q_1, \dots, q_n)$$

$$\alpha = f_\alpha(q_1, \dots, q_n)$$

$$\beta = f_\beta(q_1, \dots, q_n)$$

$$\gamma = f_\gamma(q_1, \dots, q_n)$$

$$q_k = f_k(x, y, z, \alpha, \beta, \gamma) \quad , k = 1, \dots, n$$

n es GDL < 6

= 6

> 6

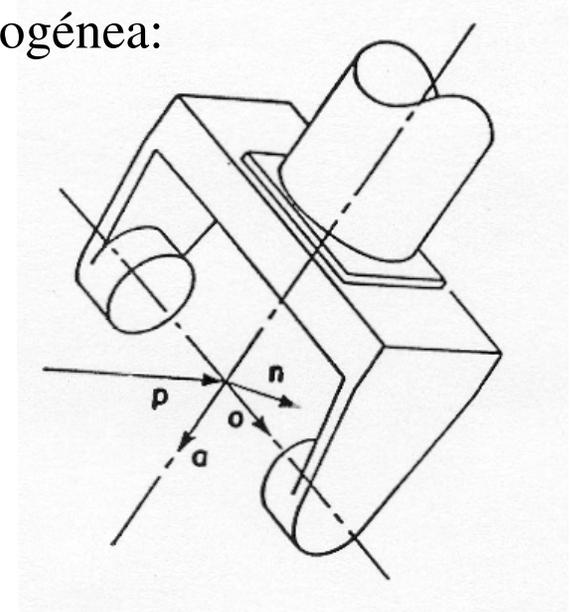


Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea**

▶ Resolución a partir de la matriz de transformación homogénea:

$$\begin{bmatrix} \mathbf{n} & \mathbf{o} & \mathbf{a} & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} = [\mathbf{T}] = {}^0A_1 \dots {}^{n-1}A_n$$



En caso de un robot de 6 GDL:

- 12 ecuaciones
- Se buscan sólo 6 relaciones (una por cada GDL)
- Existirán necesariamente ciertas dependencias entre las 12 expresiones
(resultado de la condición de ortonormalidad de los vectores \mathbf{n} , \mathbf{o} y \mathbf{a})



Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea**

$$[\mathbf{T}] = {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3$$

$$\left({}^0A_1\right)^{-1} \cdot [\mathbf{T}] = {}^1A_2 \cdot {}^2A_3$$

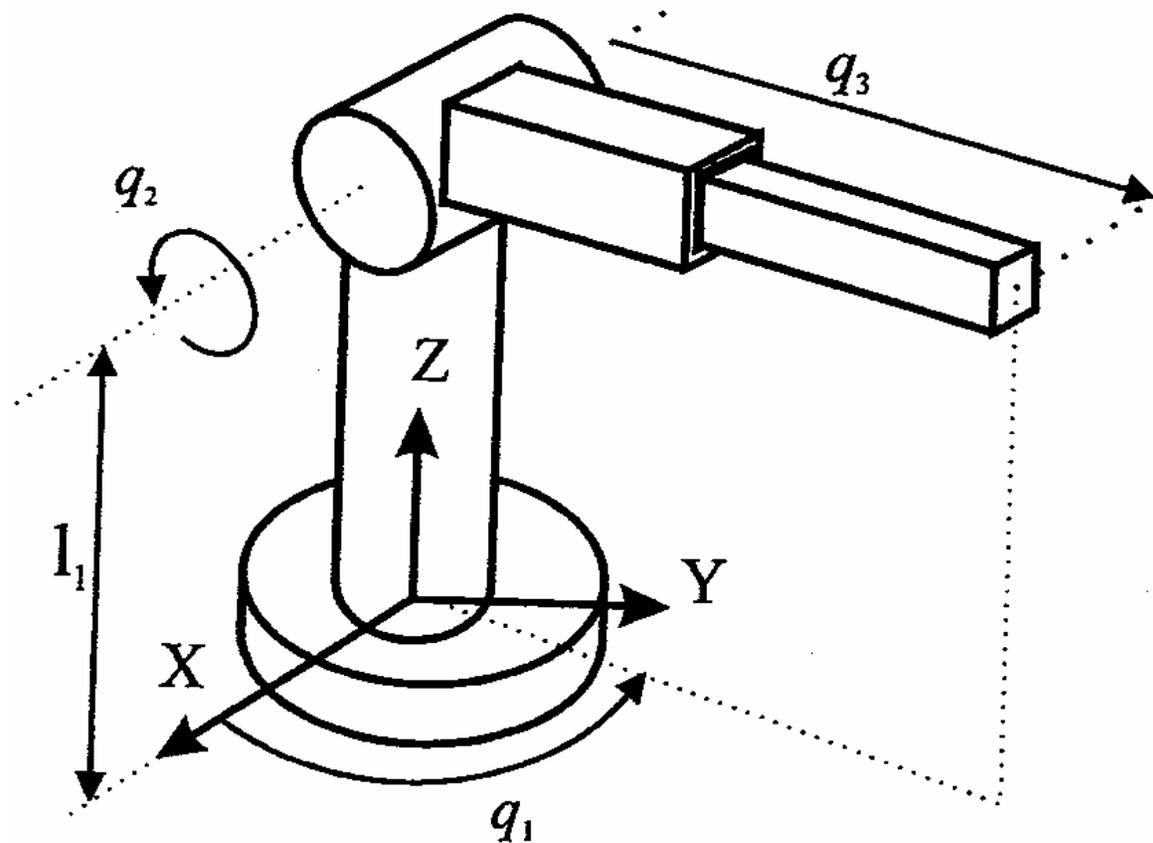
$$\left({}^1A_2\right)^{-1} \cdot \left({}^0A_1\right)^{-1} \cdot [\mathbf{T}] = {}^2A_3$$

- ▶ Los miembros de la izquierda son función de: (q_1, \dots, q_k)
- ▶ Los miembros de la derecha son función de: (q_{k+1}, \dots, q_n)



Cinemática inversa

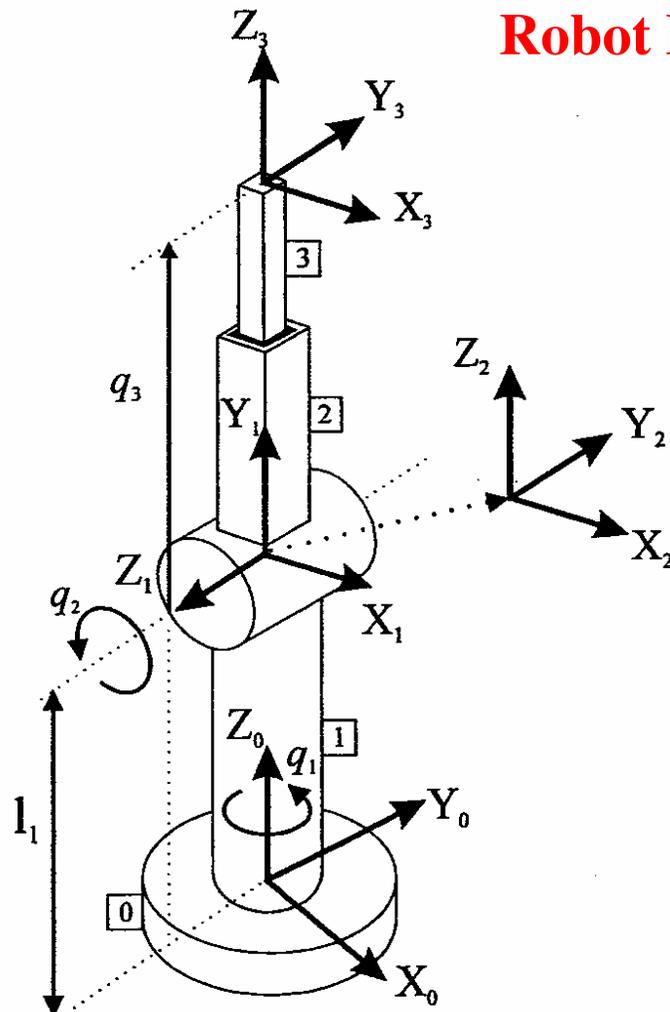
- **Métodos algebraicos: Matriz de transformación homogénea**
Robot Polar de 3 GDL





Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea Robot Polar de 3 GDL**



	Gz	Tz	Tx	Gx
i	θ_i	d_i	a_i	α_i
1	q_1	l_1	0	90°
2	q_2	0	0	-90°
3	0	q_3	0	0

Tabla de parámetros
Denavit-Hartenberg



Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea Robot Polar de 3 GDL**

$${}^0A_1 = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1A_2 = \begin{bmatrix} C_2 & 0 & -S_2 & 0 \\ S_2 & 0 & C_2 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^2A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0A_2 = \begin{bmatrix} C_1C_2 & -S_1 & -C_1S_2 & 0 \\ S_1C_2 & C_1 & -S_1S_2 & 0 \\ S_2 & 0 & C_2 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T = {}^0A_3 = \begin{bmatrix} C_1C_2 & -S_1 & -C_1S_2 & -q_3C_1S_2 \\ S_1C_2 & C_1 & -S_1S_2 & -q_3S_1S_2 \\ S_2 & 0 & C_2 & q_3C_2 + l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea Robot Polar de 3 GDL**

$$[\mathbf{T}] = {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3 \quad \Rightarrow \quad ({}^0A_1)^{-1} \cdot [\mathbf{T}] = {}^1A_2 \cdot {}^2A_3 \quad \Rightarrow \quad ({}^1A_2)^{-1} \cdot ({}^0A_1)^{-1} \cdot [\mathbf{T}] = {}^2A_3$$

$$[\mathbf{T}] = \begin{bmatrix} \mathbf{n} & \mathbf{o} & \mathbf{a} & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\Downarrow \quad [\mathbf{T}] = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea**
Robot Polar de 3 GDL

$$[\mathbf{T}] = {}^0A_1 \cdot {}^1A_2 \cdot {}^2A_3 \quad \Rightarrow \quad ({}^0A_1)^{-1} \cdot [\mathbf{T}] = {}^1A_2 \cdot {}^2A_3 \quad \Rightarrow \quad ({}^1A_2)^{-1} \cdot ({}^0A_1)^{-1} \cdot [\mathbf{T}] = {}^2A_3$$

$${}^0A_1^{-1} = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} C_1 & S_1 & 0 & 0 \\ 0 & 0 & 1 & -l_1 \\ S_1 & -C_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1A_2^{-1} = \begin{bmatrix} C_2 & 0 & -S_2 & 0 \\ S_2 & 0 & C_2 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} C_2 & S_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -S_2 & C_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2A_3^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea Robot Polar de 3 GDL**

$$\begin{aligned}
 \left({}^0A_1\right)^{-1} \cdot [\mathbf{T}] &= {}^1A_2 \cdot {}^2A_3 \\
 &= \begin{bmatrix} C_1 & S_1 & 0 & 0 \\ 0 & 0 & 1 & -l_1 \\ S_1 & -C_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_2 & 0 & -S_2 & 0 \\ S_2 & 0 & C_2 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} C_2 & 0 & -S_2 & -S_2q_3 \\ S_2 & 0 & C_2 & C_2q_3 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

De las 12 relaciones establecidas en esta ecuación, interesan aquellas que expresan q_1 en función de constantes.

$$S_1 p_x - C_1 p_y = 0 \quad \Rightarrow \quad \tan(q_1) = \frac{p_x}{p_y} \quad \Rightarrow \quad q_1 = \arctan\left(\frac{p_x}{p_y}\right)$$



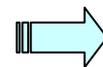
Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea Robot Polar de 3 GDL**

$$\begin{aligned}
 & \left({}^1A_2\right)^{-1} \cdot \left({}^0A_1\right)^{-1} \cdot [\mathbf{T}] = {}^2A_3 \\
 & = \begin{bmatrix} C_2 & S_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -S_2 & C_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_1 & S_1 & 0 & 0 \\ 0 & 0 & 1 & -l_1 \\ S_1 & -C_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 & = \begin{bmatrix} C_2C_1 & C_1S_1 & S_2 & -l_1S_2 \\ -S_1 & C_1 & 0 & 0 \\ -S_2C_1 & -S_1S_2 & C_2 & -C_2l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

Tomando el elemento (1,4), se tiene:

$$C_2C_1p_x + C_2S_1p_y + S_2p_z - l_1S_2 = 0$$



$$C_2(C_1p_x + S_1p_y) + S_2(p_z - l_1) = 0$$

$$\tan(q_2) = \frac{C_1p_x + S_1p_y}{(p_z - l_1)}$$





Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea Robot Polar de 3 GDL**

$$\tan(q_2) = \frac{C_1 p_x + S_1 p_y}{(p_z - l_1)}$$

Considerando que:

$$(S_1 p_x - C_1 p_y)^2 = S_1^2 p_x^2 - C_1^2 p_y^2 - 2S_1 C_1 p_x p_y = 0$$

$$(1 - C_1^2) p_x^2 + (1 - S_1^2) p_y^2 = 2S_1 C_1 p_x p_y$$

$$C_1^2 p_x^2 + S_1^2 p_y^2 + 2S_1 C_1 p_x p_y = p_x^2 + p_y^2$$

$$C_1 p_x + S_1 p_y = \sqrt{p_x^2 + p_y^2}$$

$$q_2 = \arctan \frac{\sqrt{p_x^2 + p_y^2}}{(p_z - l_1)}$$



Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea Robot Polar de 3 GDL**

$$\begin{aligned}
 & \left({}^1A_2\right)^{-1} \cdot \left({}^0A_1\right)^{-1} \cdot [\mathbf{T}] = {}^2A_3 \\
 & = \begin{bmatrix} C_2 & S_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -S_2 & C_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_1 & S_1 & 0 & 0 \\ 0 & 0 & 1 & -l_1 \\ S_1 & -C_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 & = \begin{bmatrix} C_2C_1 & C_1S_1 & S_2 & -l_1S_2 \\ -S_1 & C_1 & 0 & 0 \\ -S_2C_1 & -S_1S_2 & C_2 & -C_2l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

Tomando el elemento (3,4), se tiene:

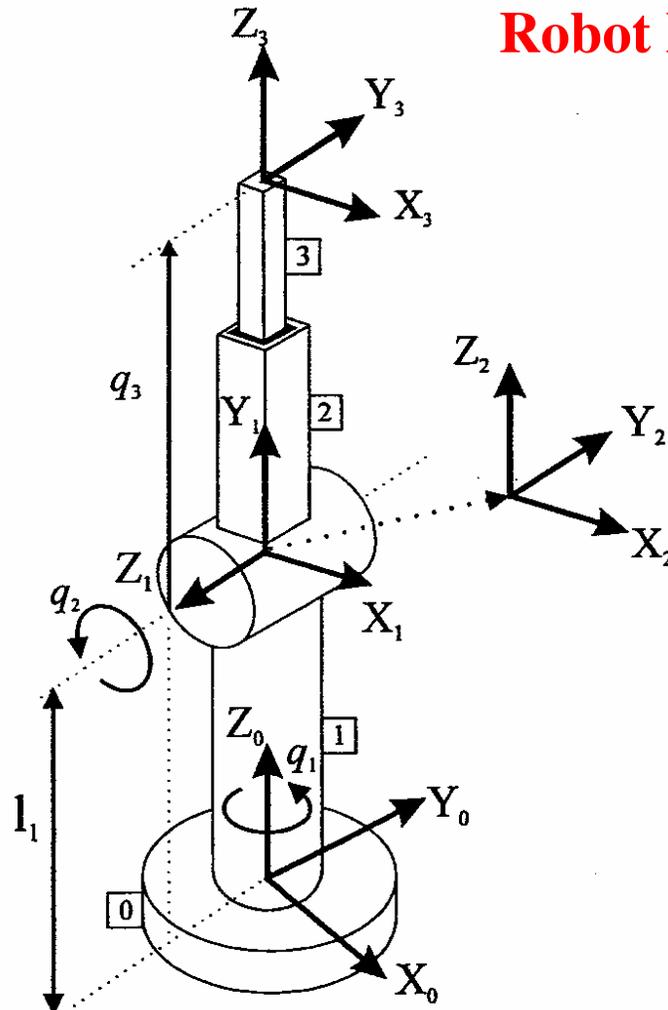
$$-S_2C_1p_x - S_2S_1p_y + C_2p_z - l_1C_2 = q_3 \quad \Rightarrow \quad C_2(p_z - l_1) - S_2(C_1p_x + S_1p_y) = q_3$$

$$q_3 = C_2(p_z - l_1) - S_2\sqrt{p_x^2 + p_y^2}$$



Cinemática inversa

- **Métodos algebraicos: Matriz de transformación homogénea**
Robot Polar de 3 GDL



Cinemática Inversa

$$q_1 = \arctan\left(\frac{p_x}{p_y}\right)$$

$$q_2 = \arctan\frac{\sqrt{p_x^2 + p_y^2}}{(p_z - l_1)}$$

$$q_3 = C_2(p_z - l_1) - S_2\sqrt{p_x^2 + p_y^2}$$

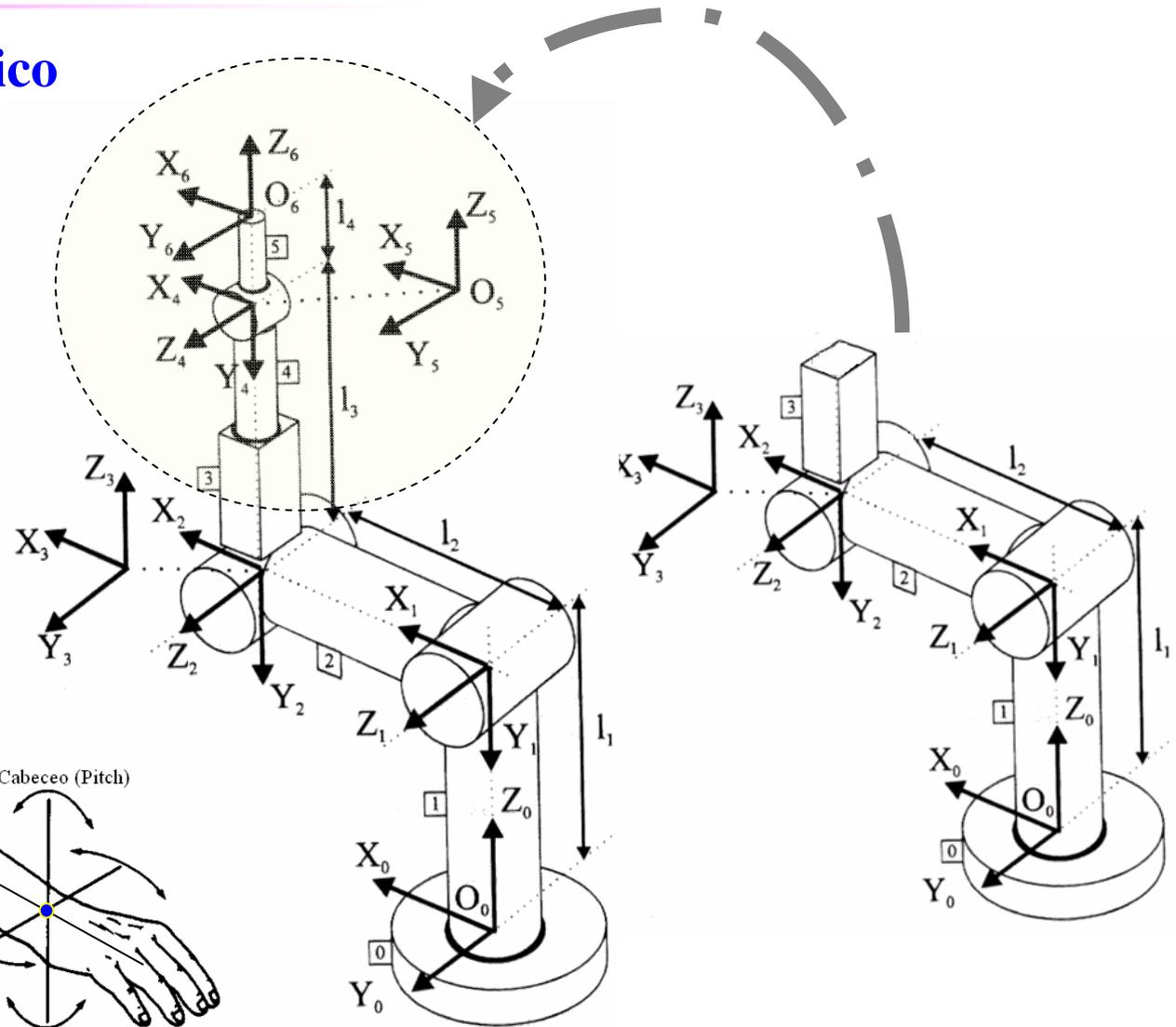
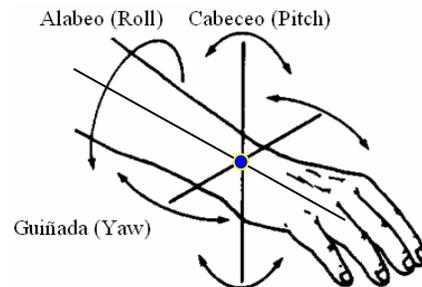


Cinemática inversa

Desacoplo cinemático

Los robots cuentan con otros grados de libertad adicionales, situado al final de la cadena cinemática y cuyos ejes, generalmente, se cortan en un punto, que informalmente se denomina **muñeca del robot**.

O **TCP (Tool Center Point)**





Cinemática inversa

● Desacoplo cinemático

Dados

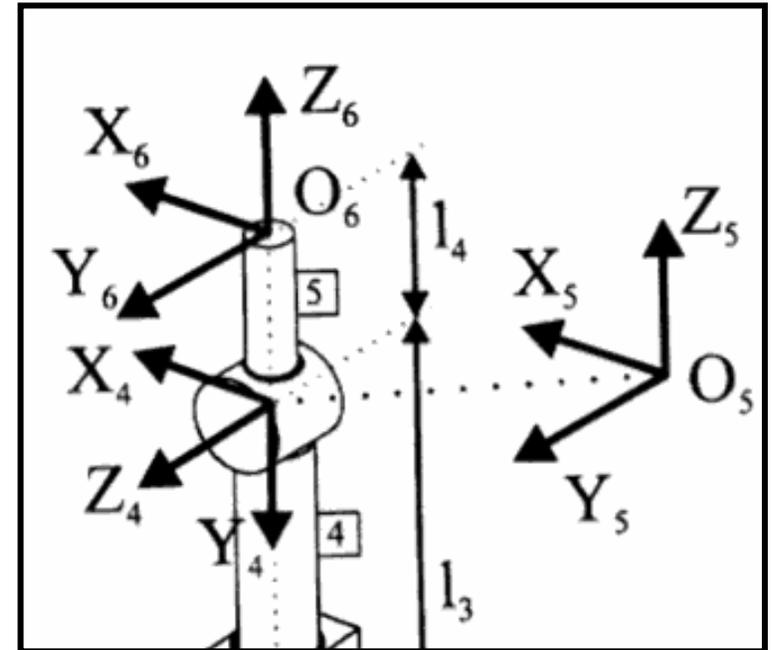
Una posición y orientación final deseadas

Se requiere

Establecer las coordenadas del punto de corte de los 3 últimos ejes (muñeca del robot)

¿Cómo?

- Calculando los valores de las tres (o bien $n-3$ en caso general) primeras variables articulares (q_1, q_2, q_3) que consiguen posicionar este punto.
- A partir de los datos de orientación y de los ya calculados (q_1, q_2, q_3), obtener los valores del resto de las variables articulares.





Cinemática inversa

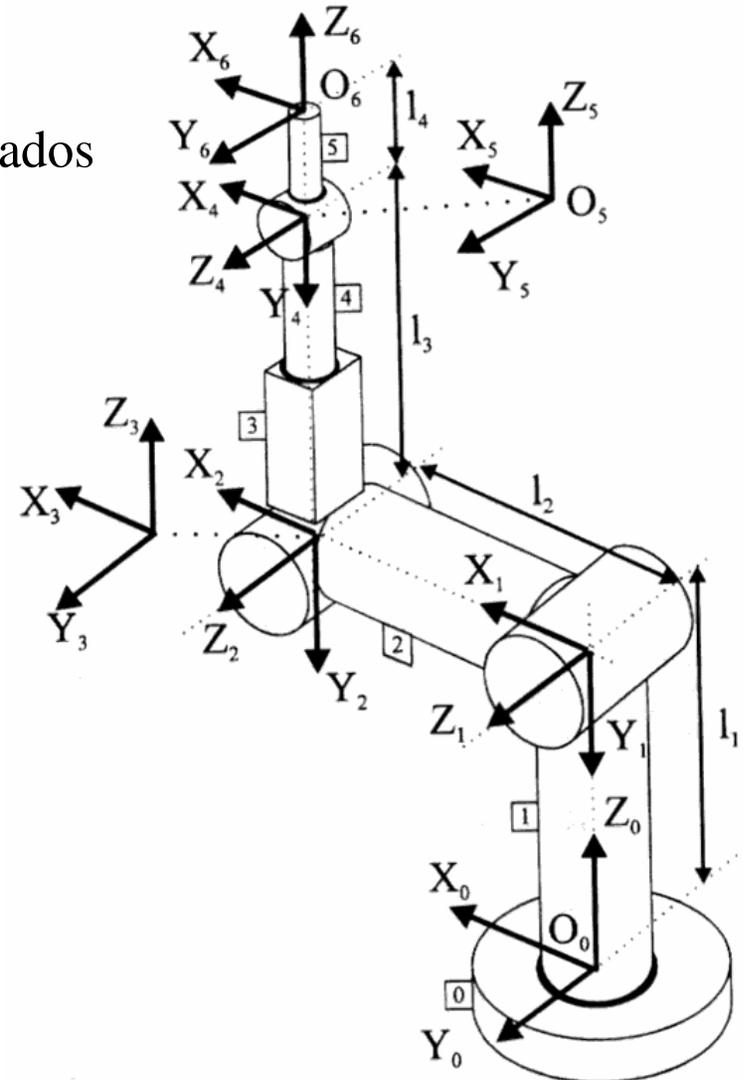
● Desacoplo cinemático

- ▶ Calculando los valores de los tres primeros grados de libertad: q_1 , q_2 y q_3 .

$$q_1 = \arctan \left(\frac{p_x}{p_y} \right)$$

$$q_2 = \arctan \frac{\sqrt{p_x^2 + p_y^2}}{(p_z - l_1)}$$

$$q_3 = C_2(p_z - l_1) - S_2 \sqrt{p_x^2 + p_y^2}$$





Cinemática inversa

Desacoplo cinemático

- Mediante las matrices de rotación obtengo los tres últimos grados de libertad: q_4 , q_5 y q_6 .

$${}^0R_6 = [n \ o \ a] = {}^0R_3 \ {}^3R_6$$

$${}^0R_3 = {}^0A_1 \ {}^1A_2 \ {}^2A_3$$

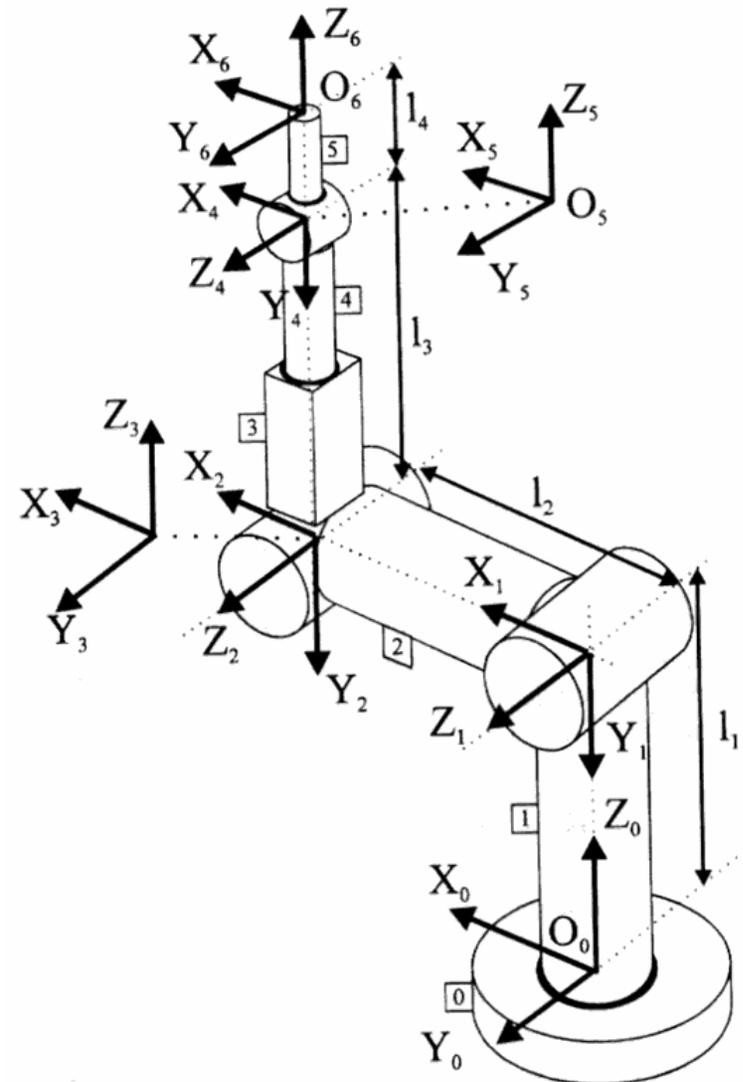
$${}^3R_6 = [r_{ij}] = ({}^0R_3)^{-1} \ {}^0R_6 = ({}^0R_3)^{-1} [n \ o \ a]$$

$${}^3R_6 = {}^3R_4 \ {}^4R_5 \ {}^5R_6$$

$${}^3R_4 = \begin{bmatrix} C_4 & 0 & -S_4 \\ S_4 & 0 & C_4 \\ 0 & -1 & 0 \end{bmatrix}$$

$${}^4R_5 = \begin{bmatrix} C_5 & 0 & S_5 \\ S_5 & 0 & -C_5 \\ 0 & 1 & 0 \end{bmatrix}$$

$${}^5R_6 = \begin{bmatrix} C_6 & -S_6 & 0 \\ S_6 & C_6 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$





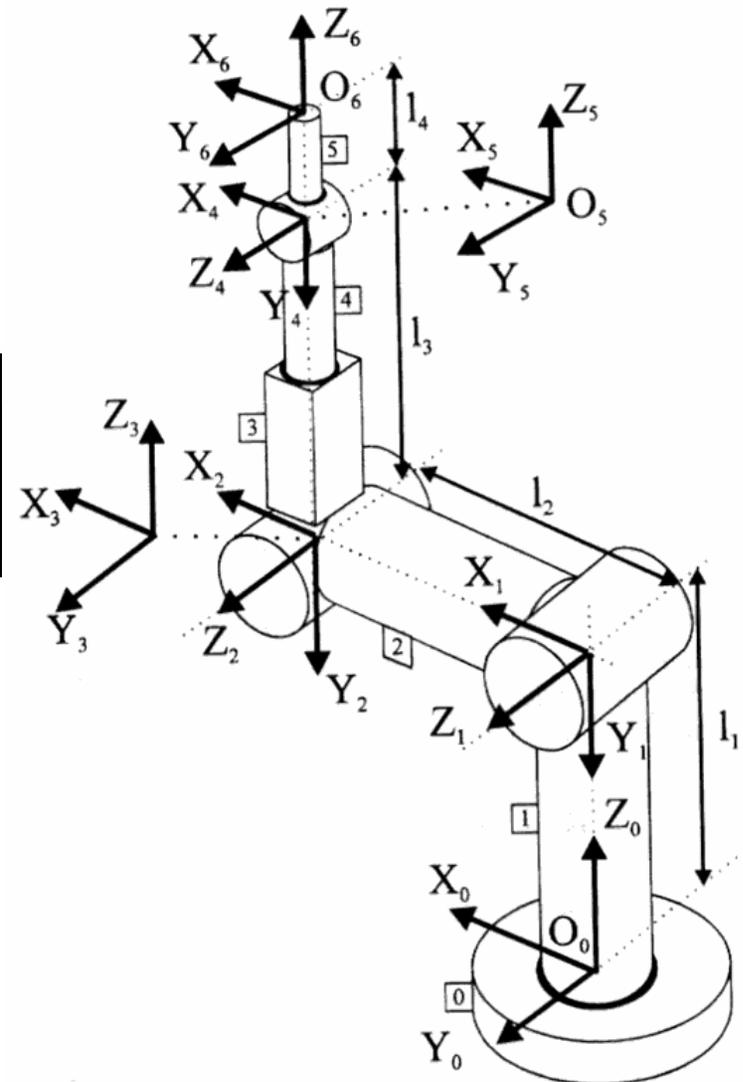
Cinemática inversa

Desacoplo cinemático

- Mediante las matrices de rotación obtengo los tres últimos grados de libertad: q_4 , q_5 y q_6 .

$${}^3R_6 = \begin{bmatrix} C_4 C_5 C_6 - S_4 S_6 & -C_4 C_5 S_6 - S_4 C_6 & C_4 S_5 \\ S_4 C_5 C_6 + C_4 S_6 & -S_4 C_5 S_6 + C_4 C_6 & -S_4 C_5 \\ -S_5 C_6 & S_5 S_6 & C_5 \end{bmatrix}$$

$$[r_{ij}] = \begin{bmatrix} C_4 C_5 C_6 - S_4 S_6 & -C_4 C_5 S_6 - S_4 C_6 & C_4 S_5 \\ S_4 C_5 C_6 + C_4 S_6 & -S_4 C_5 S_6 + C_4 C_6 & -S_4 C_5 \\ -S_5 C_6 & S_5 S_6 & C_5 \end{bmatrix}$$





Cinemática inversa

Desacoplo cinemático

- Mediante las matrices de rotación obtengo los tres últimos grados de libertad: q_4 , q_5 y q_6 .

$$[r_{ij}] = \begin{bmatrix} C_4 C_5 C_6 - S_4 S_6 & -C_4 C_5 S_6 - S_4 C_6 & C_4 S_5 \\ S_4 C_5 C_6 + C_4 S_6 & -S_4 C_5 S_6 + C_4 C_6 & -S_4 C_5 \\ -S_5 C_6 & S_5 S_6 & C_5 \end{bmatrix}$$

$$r_{13} = C_4 S_5 \quad r_{23} = -S_4 C_5$$

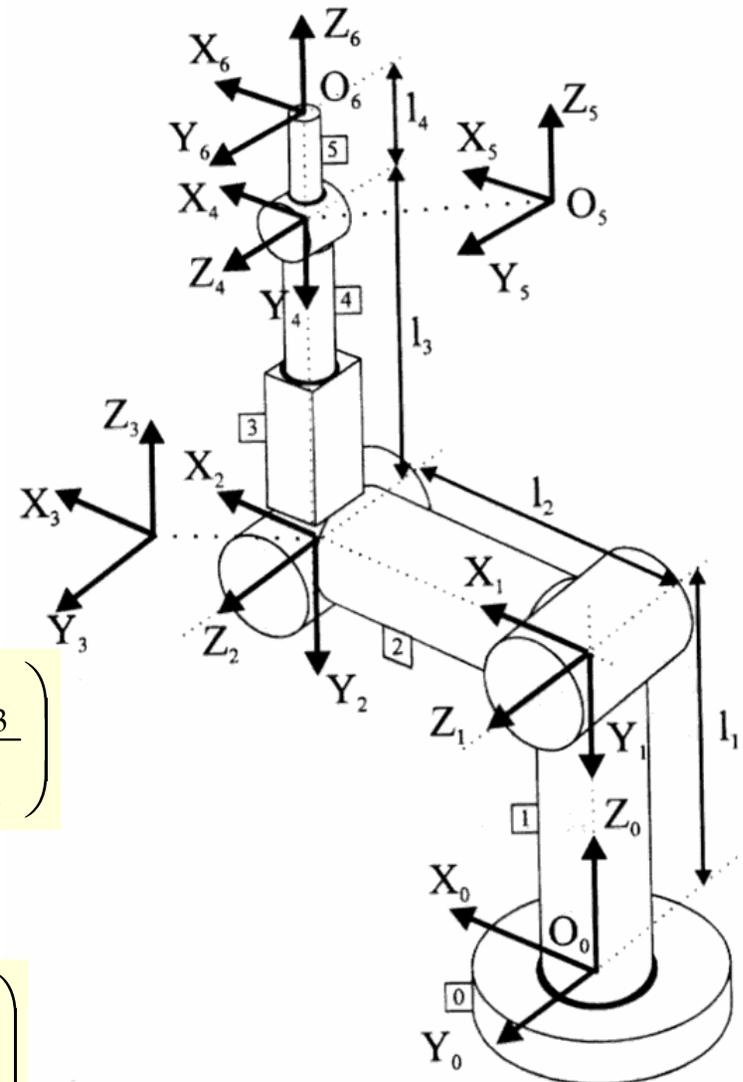
$$r_{33} = C_5$$

$$r_{31} = -S_5 C_6 \quad r_{32} = S_5 S_6$$

$$q_4 = \arcsen\left(\frac{r_{23}}{r_{33}}\right)$$

$$q_5 = \arccos(r_{33})$$

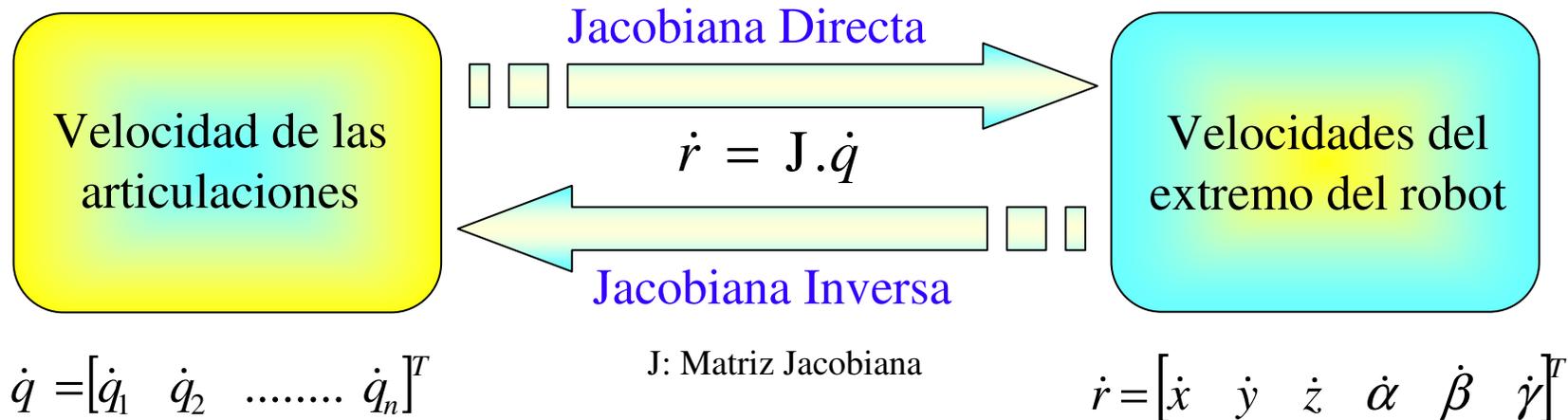
$$q_6 = \arctan\left(-\frac{r_{32}}{r_{31}}\right)$$





Cinemática diferencial

- El método más directo para obtener la relación entre velocidades articulares y del extremo del robot consiste en **diferenciar las ecuaciones** correspondientes al modelo cinemática directo.



q no necesariamente existe ni ha de ser único para una posición y orientación dada

Para todo q obtenemos una posición y orientación



Cinemática diferencial

Jacobiana Directa

Dados:

$$x = f_x(q_1, q_2, q_3, q_4, q_5, q_6)$$

$$y = f_y(q_1, q_2, q_3, q_4, q_5, q_6)$$

$$z = f_z(q_1, q_2, q_3, q_4, q_5, q_6)$$

$$\alpha = f_\alpha(q_1, q_2, q_3, q_4, q_5, q_6)$$

$$\beta = f_\beta(q_1, q_2, q_3, q_4, q_5, q_6)$$

$$\gamma = f_\gamma(q_1, q_2, q_3, q_4, q_5, q_6)$$

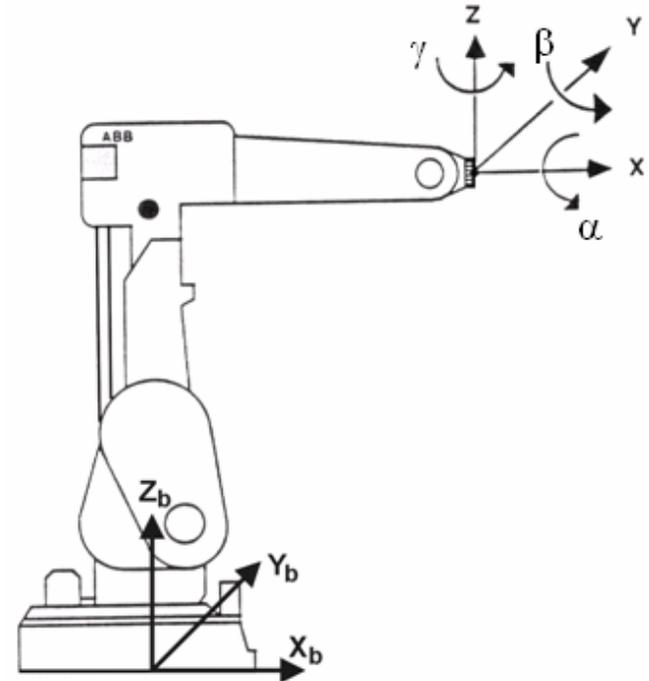
Encontrar:

$$\dot{r} = [\dot{x} \quad \dot{y} \quad \dot{z} \quad \dot{\alpha} \quad \dot{\beta} \quad \dot{\gamma}]^T$$

Solución:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \vdots \\ \dot{q}_n \end{bmatrix}$$

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_x}{\partial q_1} & \dots & \frac{\partial f_x}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_\gamma}{\partial q_1} & \dots & \frac{\partial f_\gamma}{\partial q_n} \end{bmatrix}$$





Cinemática diferencial

- Jacobiana Directa: Ejemplo - Robot plano de 3 eslabones**

El problema cinemático directo viene determinado por las ecuaciones

$$x = l_1 C_1 + l_2 C_{12} + l_3 C_{123}$$

$$y = l_1 S_1 + l_2 S_{12} + l_3 S_{123}$$

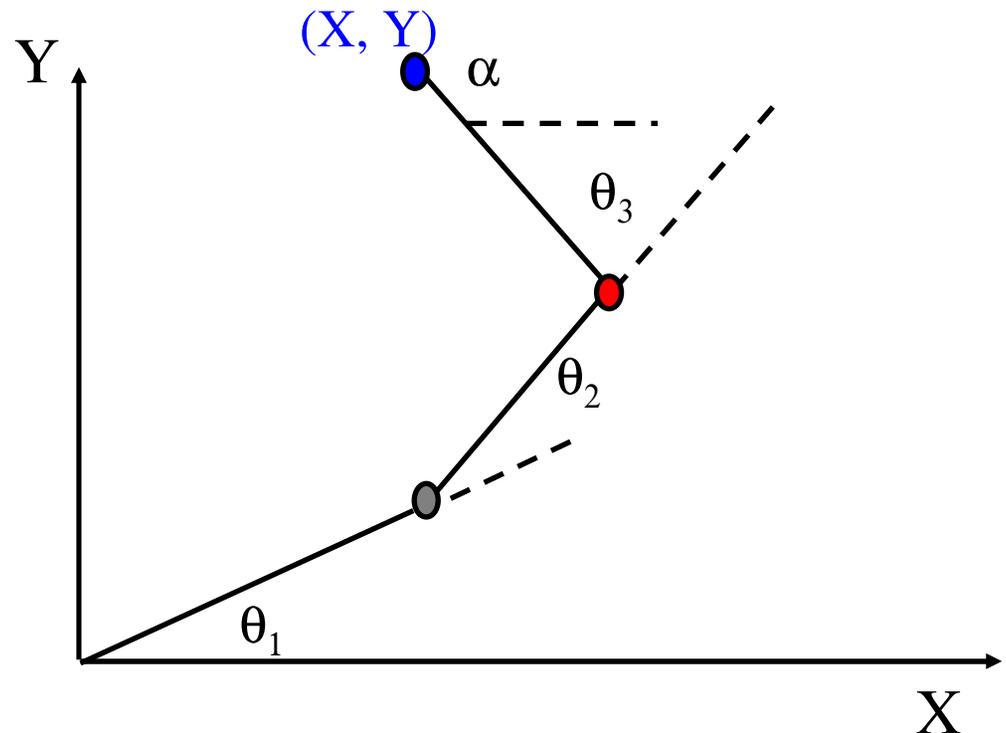
$$\alpha = \theta_1 + \theta_2 + \theta_3$$

Diferenciando obtendríamos:

$$V_x = -(l_1 S_1 + l_2 S_{12} + l_3 S_{123}) \omega_1 - (l_2 S_{12} + l_3 S_{123}) \omega_2 - l_3 S_{123} \omega_3$$

$$V_y = (l_1 C_1 + l_2 C_{12} + l_3 C_{123}) \omega_1 + (l_2 C_{12} + l_3 C_{123}) \omega_2 + l_3 C_{123} \omega_3$$

$$\omega_\alpha = \omega_1 + \omega_2 + \omega_3$$





Cinemática diferencial

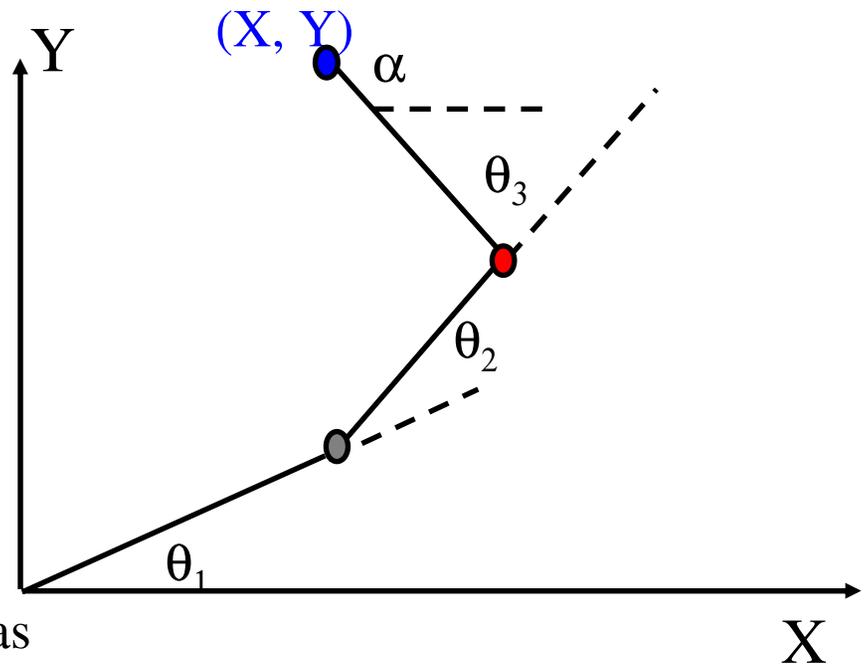
● Jacobiana Directa: Ejemplo - Robot plano de 3 eslabones

$$J = \begin{bmatrix} -(l_1 S_1 + l_2 S_{12} + l_3 S_{123}) & -(l_2 S_{12} + l_3 S_{123}) & -l_3 S_{123} \\ (l_1 C_1 + l_2 C_{12} + l_3 C_{123}) & (l_2 C_{12} + l_3 C_{123}) & l_3 C_{123} \\ 1 & 1 & 1 \end{bmatrix}$$

Con este robot podemos alcanzar cualquier posición (x, y) y orientación (α) en el plano de trabajo.

Pero tendremos algunas **limitaciones**:

- ▶ Geométricas: Longitudes de los eslabones (Campo de acción)
- ▶ Mecánicas:
 - Ángulos límite
 - Velocidades y aceleraciones máximas





Cinemática diferencial

Jacobiana Directa: Ejemplo - Robot SCARA

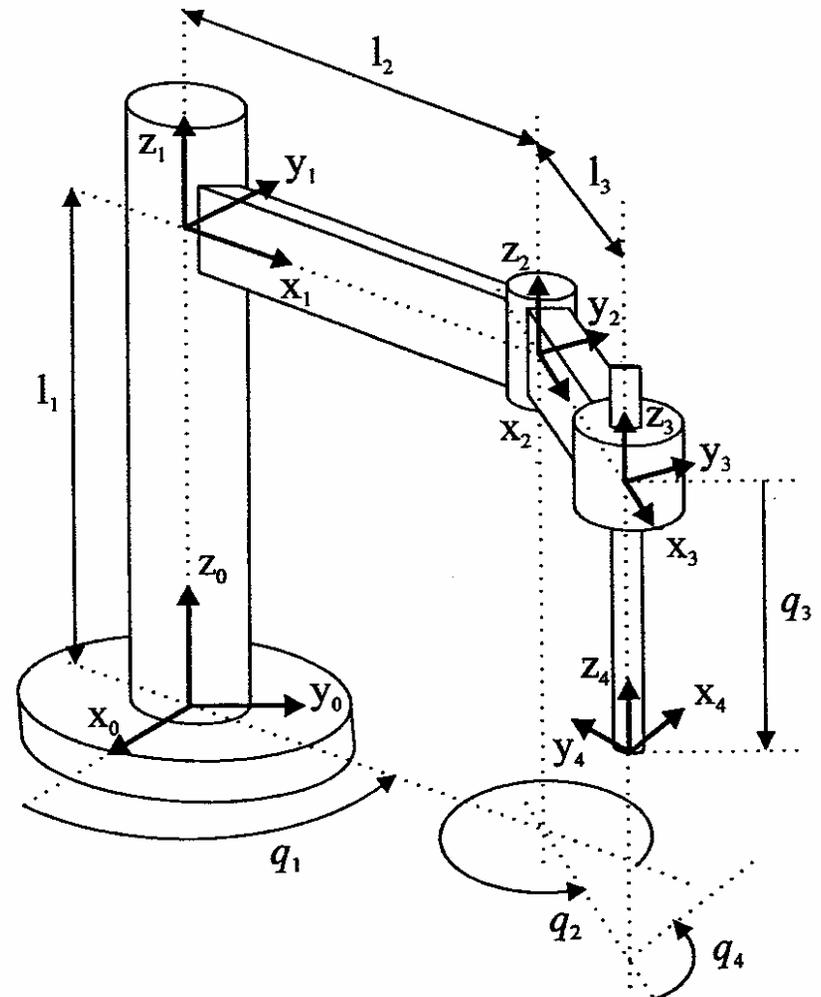
El problema cinemático directo viene determinado por las ecuaciones

$$x = l_3 C_{12} + l_2 C_1$$

$$y = l_3 S_{12} + l_2 S_1$$

$$z = l_1 - q_3$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -(l_3 S_{12} + l_2 S_1) & -l_3 S_{12} & 0 \\ l_3 C_{12} + l_2 C_1 & l_3 C_{12} & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix}$$





Cinemática diferencial

Jacobiana Directa: Ejemplo - Robot SCARA

Si el robot se encuentra en un momento determinado en la posición dada por:

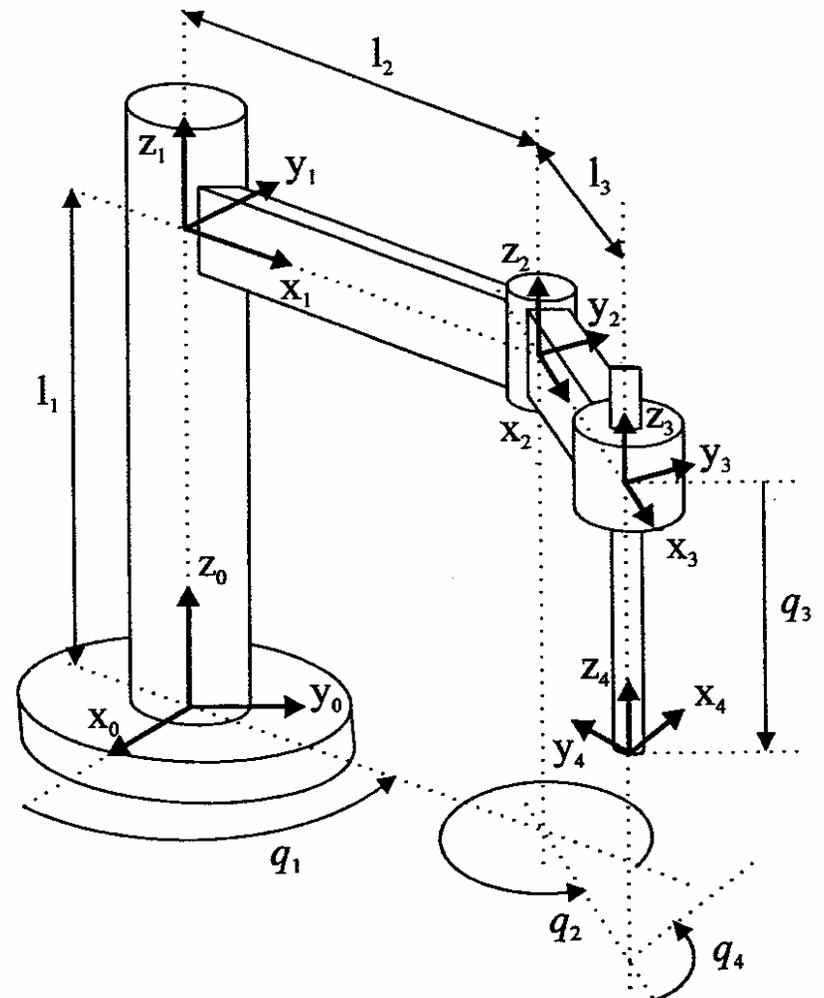
$$q_1 = \frac{\pi}{6} \text{ rad.} \quad q_2 = \frac{\pi}{4} \text{ rad.} \quad q_3 = 0,75 \text{ rad.}$$

moviéndose a una velocidad articular de valor instantáneo:

$$\dot{q}_1 = \frac{\pi}{2} \text{ rad/s.} \quad \dot{q}_2 = \frac{\pi}{2} \text{ rad/s.} \quad \dot{q}_3 = 1 \text{ m/s.}$$

con $l_2=l_3=1 \text{ m}$, la velocidad de su extremo será:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -1,465 & -0,965 & 0 \\ 1,124 & 0,258 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \frac{\pi}{2} \\ \frac{\pi}{2} \\ 1 \end{bmatrix} = \begin{bmatrix} -3,81 \\ 2,17 \\ -1 \end{bmatrix}$$





Cinemática diferencial

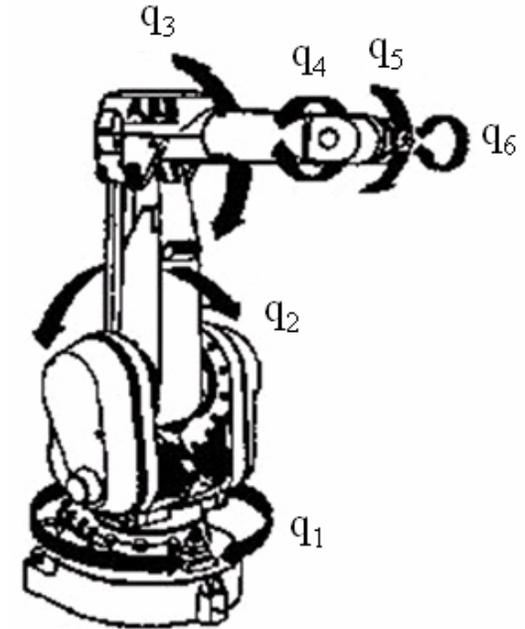
Jacobiana Inversa:

Dados: $q_k = f_k(x, y, z, \alpha, \beta, \gamma)$

$K = 1, \dots, n$ (n es GDL = 6)

Encontrar: $\dot{q} = [\dot{q}_1 \quad \dot{q}_2 \quad \dots \quad \dot{q}_n]^T$

Solución:
$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \cdot \\ \cdot \\ \cdot \\ \dot{q}_n \end{bmatrix} = \mathbf{J}^{-1} \cdot \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix}$$



$$\mathbf{J}^{-1} = \begin{bmatrix} \frac{\partial f_1}{\partial q_x} & \dots & \frac{\partial f_1}{\partial q_y} \\ \frac{\partial f_n}{\partial q_x} & \dots & \frac{\partial f_n}{\partial q_y} \end{bmatrix}$$



Cinemática diferencial

● Jacobiana Inversa:

▶ **Simbólicamente:** De gran dificultad y complejidad porque hay que invertir una matriz de $m \times n$ elementos, en el caso más típico de 6×6 .

$$J^{-1} = \begin{bmatrix} \frac{\partial f_1}{\partial q_x} & \dots & \frac{\partial f_1}{\partial q_y} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial q_x} & \dots & \frac{\partial f_n}{\partial q_y} \end{bmatrix}$$

▶ **Analíticamente:** A partir de la cinemática inversa con el problema de que la solución no es única o no existe, y queda en función de las coord. Cartesianas.

▶ **Numéricamente:** para una configuración concreta del robot se invierte la Matriz Jacobiana Directa

▶ Si no tenemos una solución analítica de la cinemática inversa obtenemos la matriz Jacobiana Inversa en una posición concreta y multiplicamos por el vector de velocidades deseadas en ese punto.

- Se han ideado métodos especiales para el cálculo rápido de la Jacobiana Inversa:

Pseudoinversa $(JJ^T)^{-1}$

- A la hora de ejecutar trayectorias hay que ir recalculándola constantemente.



Cinemática diferencial

● Configuraciones singulares

- ▶ **Definición física:** Las configuraciones singulares de un robot son aquellas para las cuales **las velocidades del extremo no se pueden realizar** por una velocidad finita de las articulaciones, o cuando **se pierde algún grado de libertad** en el movimiento
- ▶ **Definición matemática:** El robot esta en una configuración singular cuando el rango de la matriz jacobiana[mxn] sea menor m.

En el caso de que $n=m$: existirá singularidad cuando

El **determinante** de la matriz jacobiana es **nulo** $\{\det J_r(q)=0\}$

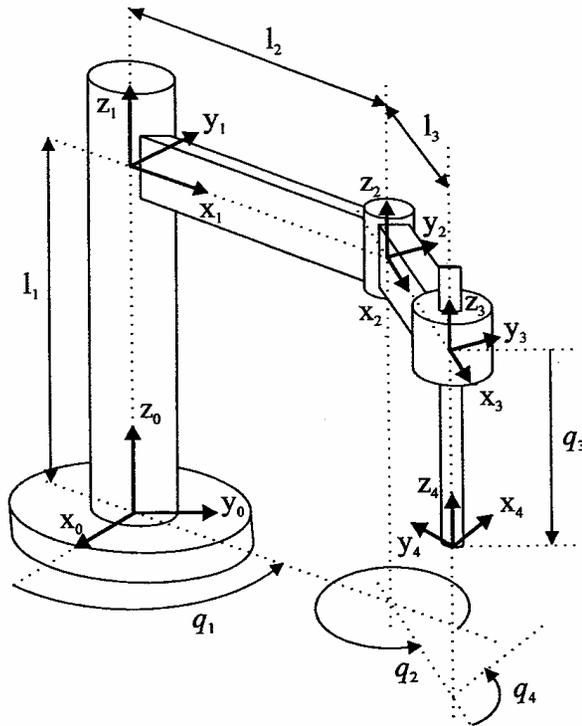
▶ De forma práctica:

- Cuando el robot esta en los límites de su espacio de trabajo porque pierde posibilidades de movimiento
- Cuando están alineados dos o más ejes de articulaciones del mismo tipo (rotación o traslación)

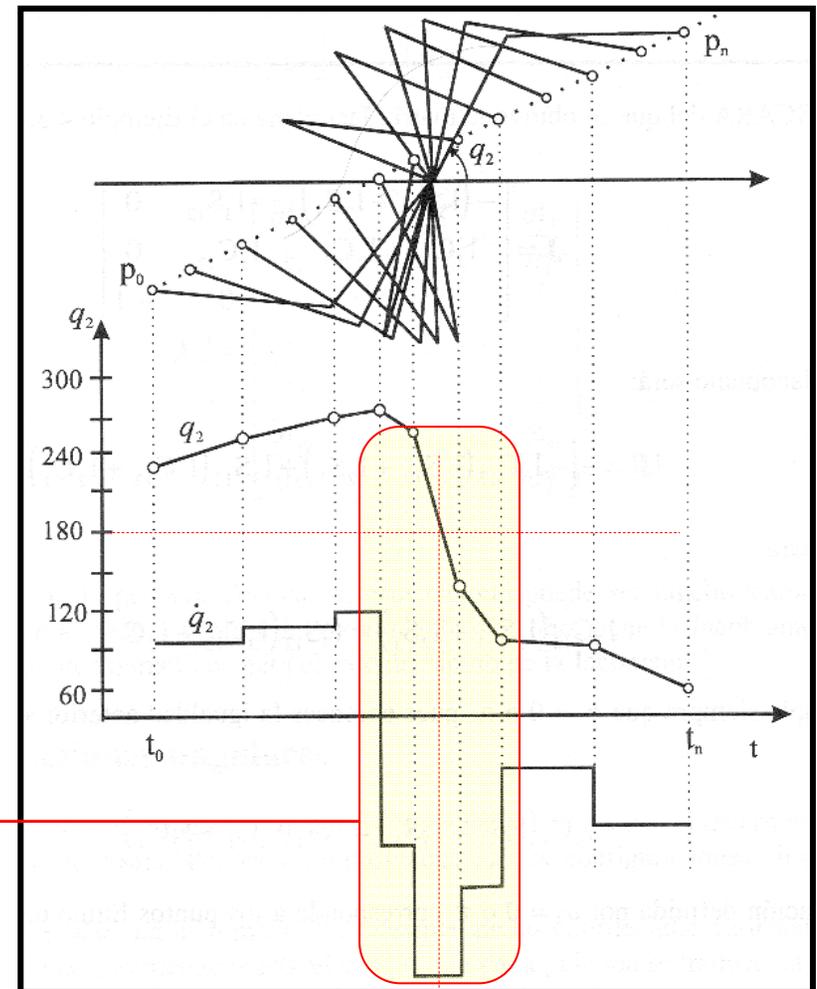


Cinemática diferencial

● Configuraciones singulares: Robot SCARA



Brusca variación de la velocidad articular que crece hasta valores **inalcanzables** en la practica





Cinemática diferencial

● Configuraciones singulares: Robot SCARA

$$\mathbf{J} = \begin{bmatrix} -(l_3 S_{12} + l_2 S_1) & -l_3 S_{12} & 0 \\ l_3 C_{12} + l_2 C_1 & l_3 C_{12} & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

el Jacobiano es:

$$|\mathbf{J}| = -[-l_3 C_{12}(l_3 S_{12} + l_2 S_1) + l_3 S_{12}(l_3 C_{12} + l_2 C_1)]$$

que se anula para:

$$l_3 C_{12}(l_3 S_{12} + l_2 S_1) = l_3 S_{12}(l_3 C_{12} + l_2 C_1)$$

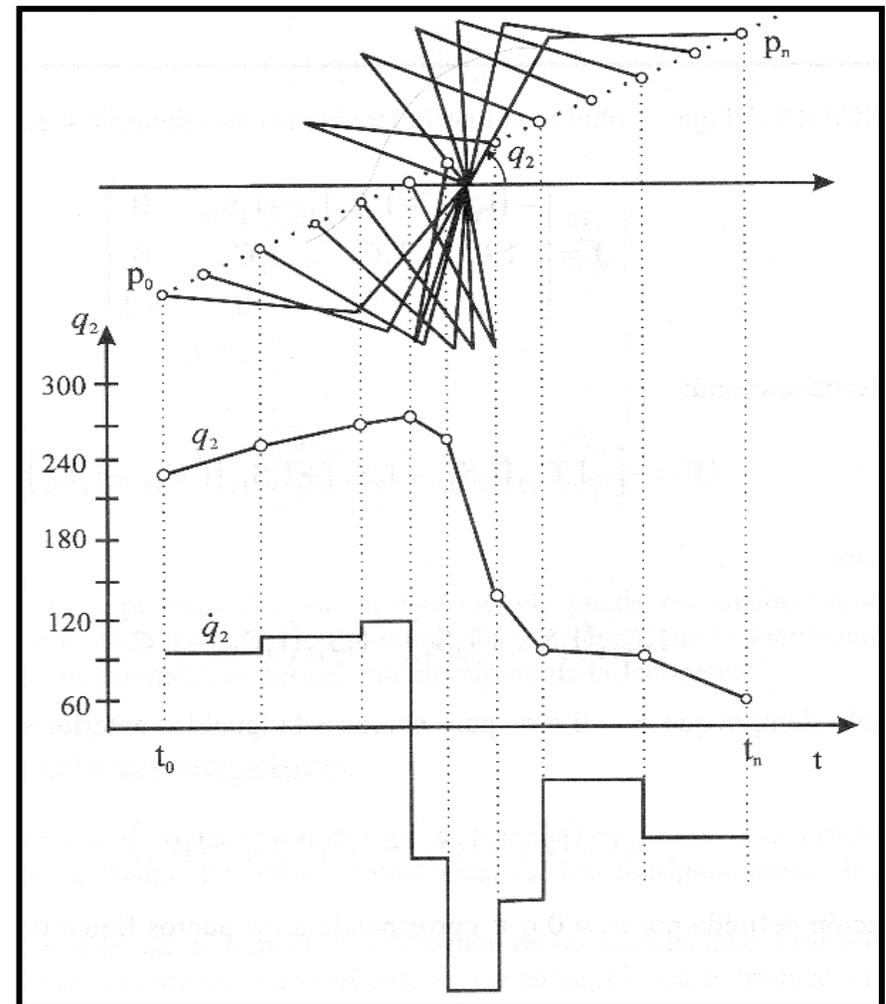
lo que se cumple siempre que $q_2=0$ o π

$q_2=0$: Limite exterior del espacio del trabajo

$q_2=\pi$: Limite interior del espacio de trabajo

la igualdad se verifica para cualquier q_1

$$l_3 C_1(l_3 S_1 + l_2 S_1) = l_3 S_1(l_3 C_1 + l_2 C_1)$$





Cinemática diferencial

- **Configuraciones singulares: Solución**

- ▶ Para resolver la aparición de una singularidad interior:

1. Identificar la articulación correspondiente al grado de libertad perdido.
2. Eliminar la fila de la Jacobiana correspondiente al grado de libertad perdido y la columna correspondiente a la articulación causante.
3. Con la nueva Jacobina reducida, obtener las velocidades de todas las articulaciones, a excepción de la eliminada, necesarias para conseguir las velocidades cartesianas deseadas. La velocidad de la articulación eliminada se mantendrá a cero.



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 6 – Jueves 4 de Mayo 2006

Cinemática - II



Automatización Industrial - II

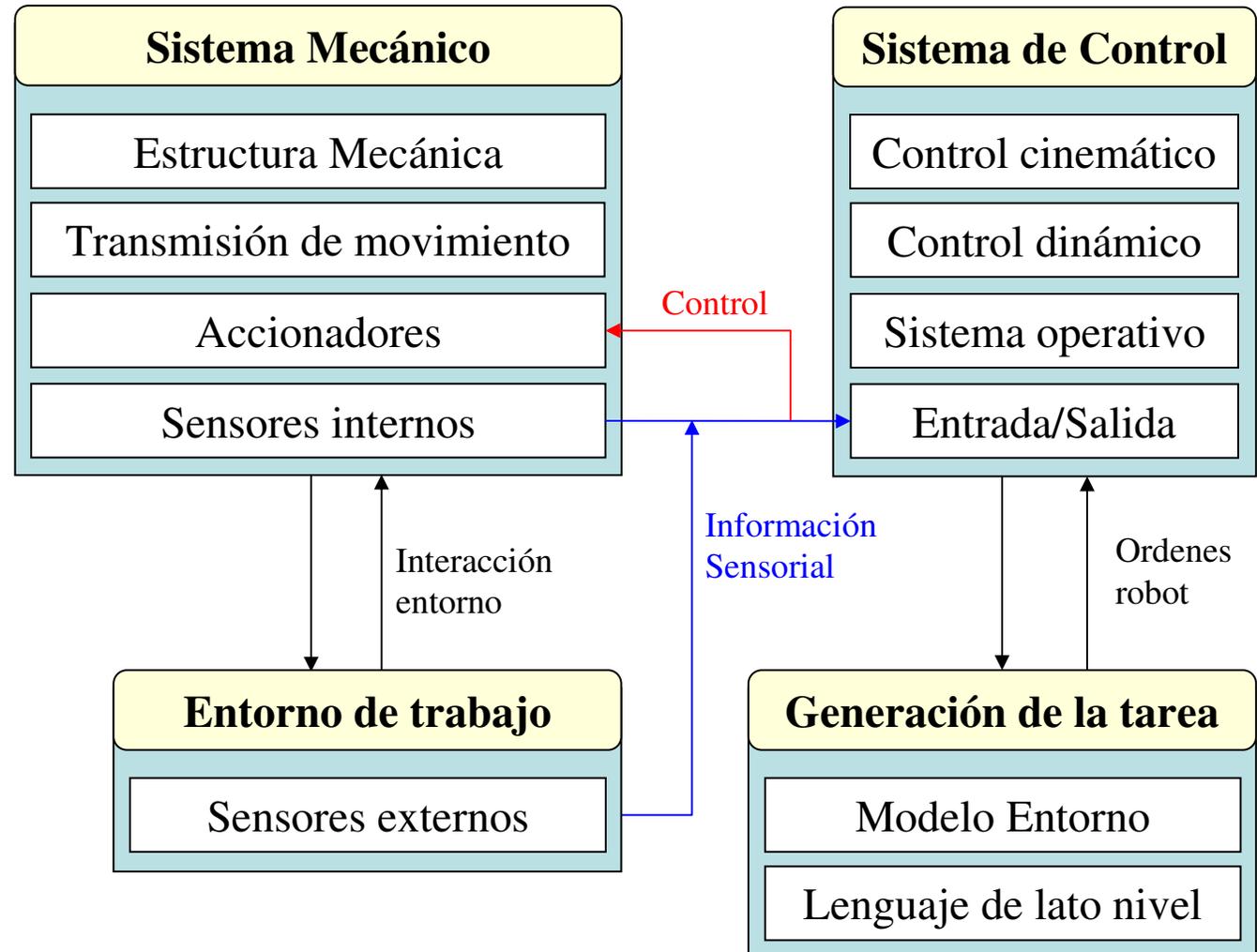
56 – 10569 2º Cuatrimestre 2006

Clase 7 – Miércoles 10 de Mayo 2006

Programación de Robots



Estructura Hardware & Software





Estructura Hardware & Software

- **Software: Características**
 - ▶ Sistema en tiempo real
 - ▶ Sistemas de múltiples ejes
 - ▶ Necesidad de sincronización de procesos
 - ▶ Gran volumen de cálculo matemático
 - ▶ Lenguaje propio de programación
 - ▶ Alto nivel de seguridad y robustez



Estructura Hardware & Software

- **Software: Procesos**

- ▶ Alta prioridad

- Sistemas de seguridad
- Parada de emergencia
- Controladores de varios ejes simultáneos (6 o más): 10-50 μ sec por eje

- ▶ Prioridad intermedia

- Sistema operativo
- Interprete de comandos
- Cálculo de trayectorias: 1-5 msec.
- Comunicaciones de tipo E/S digital y similares
- Comunicaciones de tipo Computer Link



Estructura Hardware & Software

- **Software: Procesos**

- ▶ **Prioridad baja**

- Atención a la unidad programación
- Compilador
- Editor
- Comunicaciones con disco, diskette, etc.
- Visualización de variables
- Simulación de variables.



Estructura Hardware & Software

● Hardware

▶ Monoprocesador

- Procesador único para todo el sistema.

▶ Multiprocesador

- Procesador por cada eje de control.



PUMA 560

▶ Biprocesador

- Procesadores principal y de ejes.



ABB S3

▶ Triprocesador

- Procesadores principal, de ejes y de comunicaciones.



ABB S4



Programación de robots

El robot es un manipulador multifuncional **reprogramable**.

El sistema de programación es la herramienta con que cuenta el usuario para **acceder a las prestaciones** del robot



Adaptación rápida y económica a diferentes tareas y aplicaciones



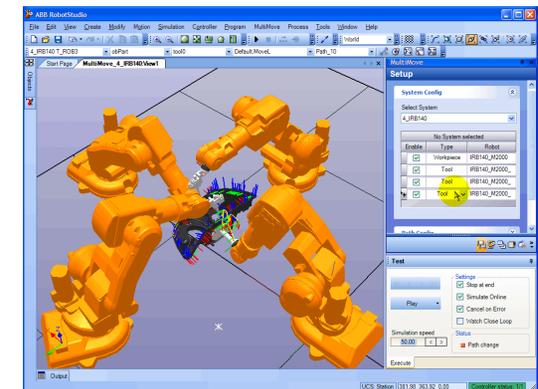
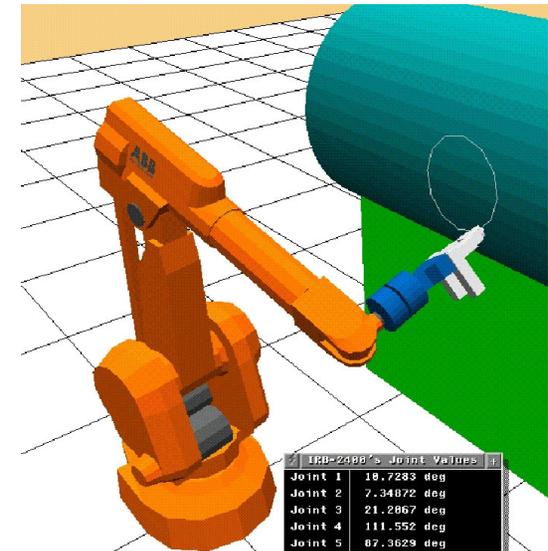
Programación de robots

► Programación es el proceso mediante el cuál se le indica al robot la **secuencia de acciones** que deberá llevar a cabo durante la realización de su tarea.

```
MoveJ A,v100,fine,pinza;  
WaitDI econtrol,listo;  
coger_pieza;  
MoveL B1,v80,z5,pinza;  
MoveL C,v80,fine,pinza;  
abrir_pinza;
```

Estas acciones consisten en su mayor parte a:

- Movimientos
- Manipulación
- Manejo de Entrada / Salida

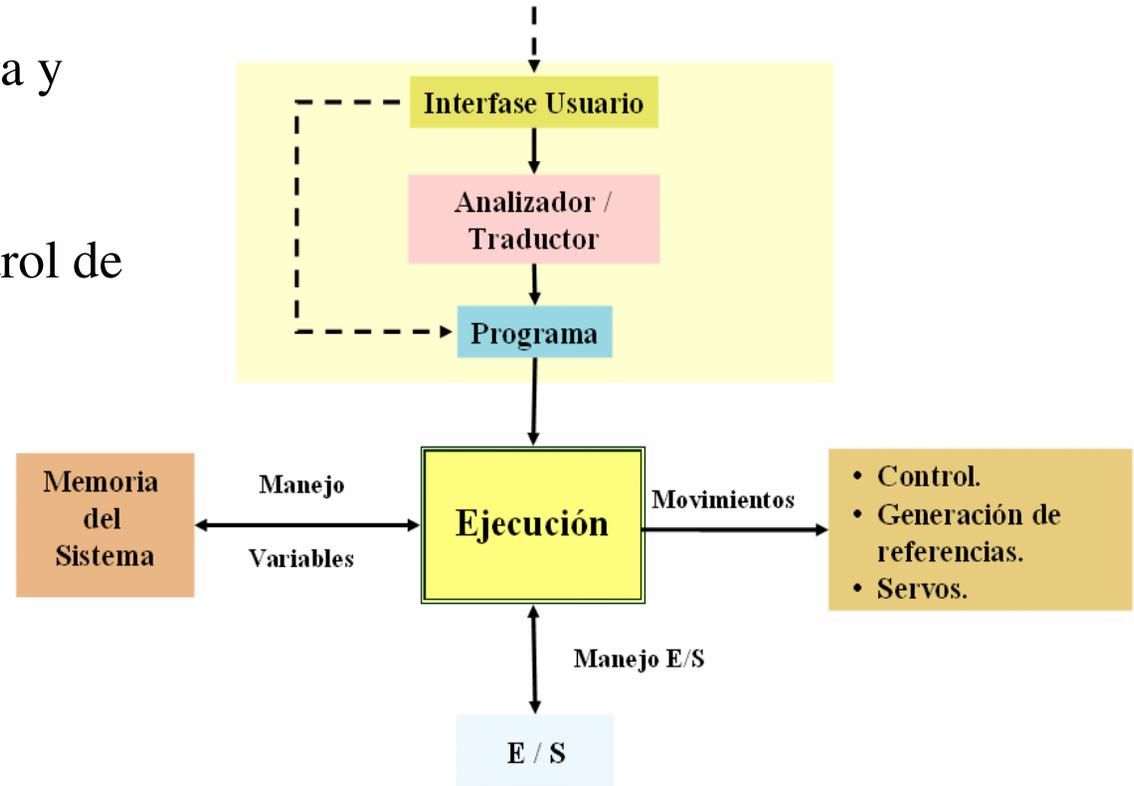




Programación de robots

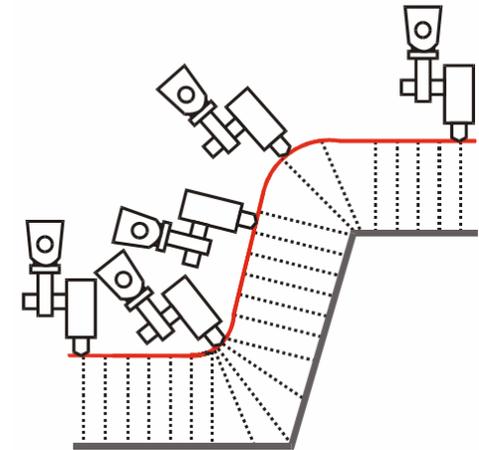
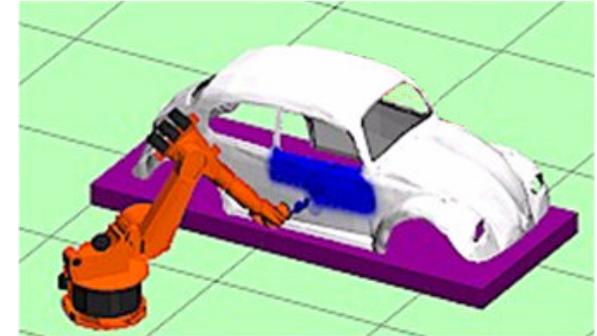
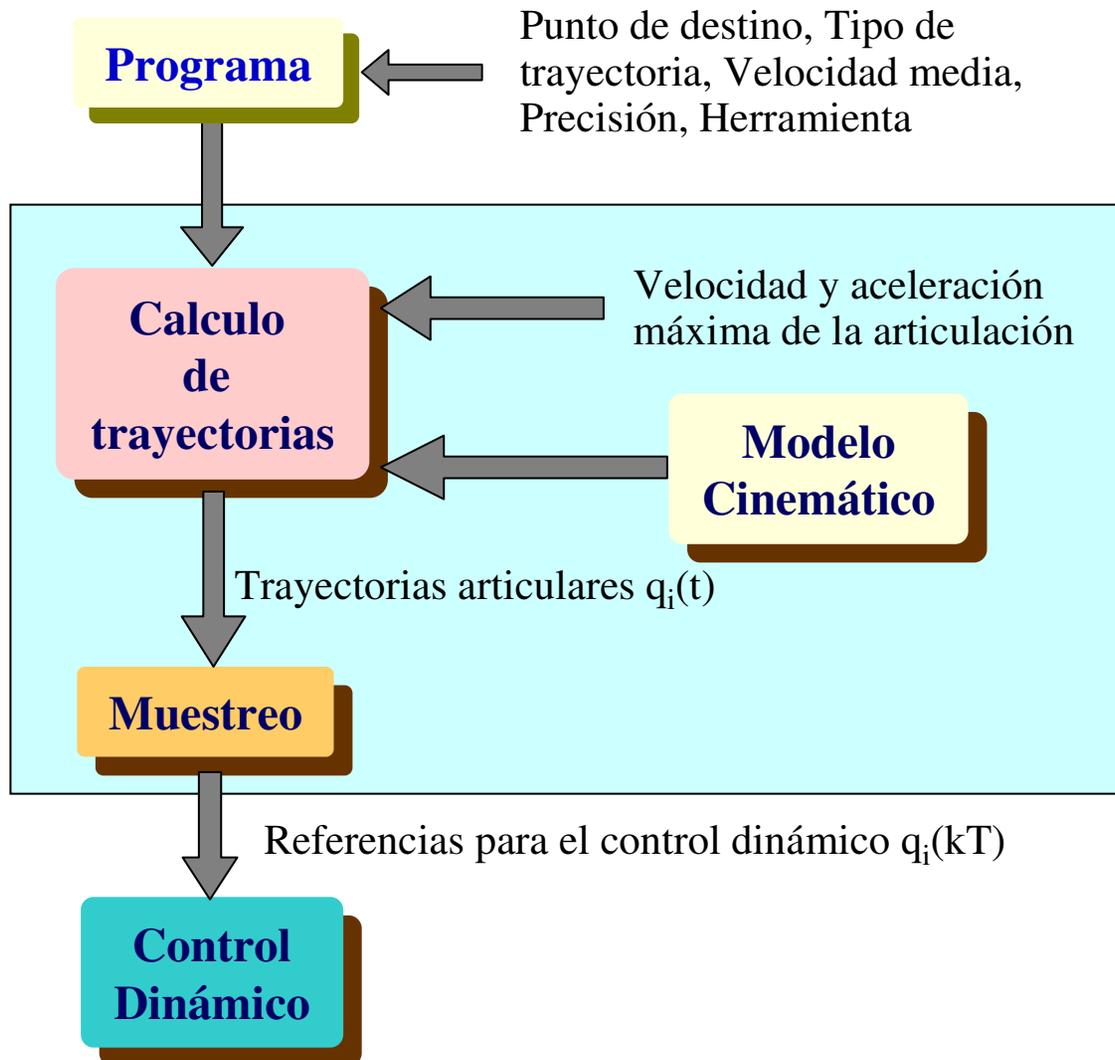
Durante la ejecución de un programa se interacciona con:

- ▶ La memoria del sistema: lectura y actualización de variables
- ▶ La memoria de programa: control de flujo de ejecución
- ▶ El sistema de control dinámico y cinemático: da las señales de mando de los accionadores
- ▶ Las entradas-salidas del sistema: digitales o analógicas, para sincronizarse con otros equipos
- ▶ Entorno exterior: mediante las acciones que ejecuta





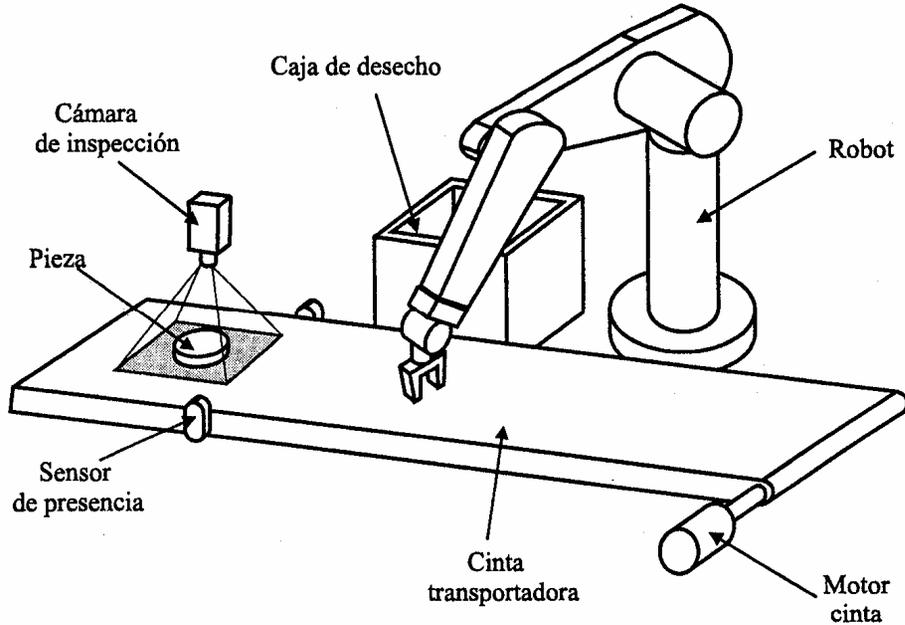
Programación de robots





Programación de robots

● Ejemplo de programa en RAPID



```
%%%  
VERSION:1  
LANGUAGE:ENGLISH  
%%%  
MODULE EJEMPLO  
PERS loaddata carga:=[5,[50,0,50],[1,0,0,0],0,0,0];  
...  
PROC Cogrer ()  
...  
ENDPROC  
PROC Dejar ()  
...  
ENDPROC  
PROC Cogrer_pieza ()  
...  
ENDPROC  
PROC Dejar_pieza ()  
...  
ENDPROC  
PROC main ()  
    Cogrer_pieza;  
    ...  
ENDPROC  
ENDMODULE
```



Métodos de Programación de robots

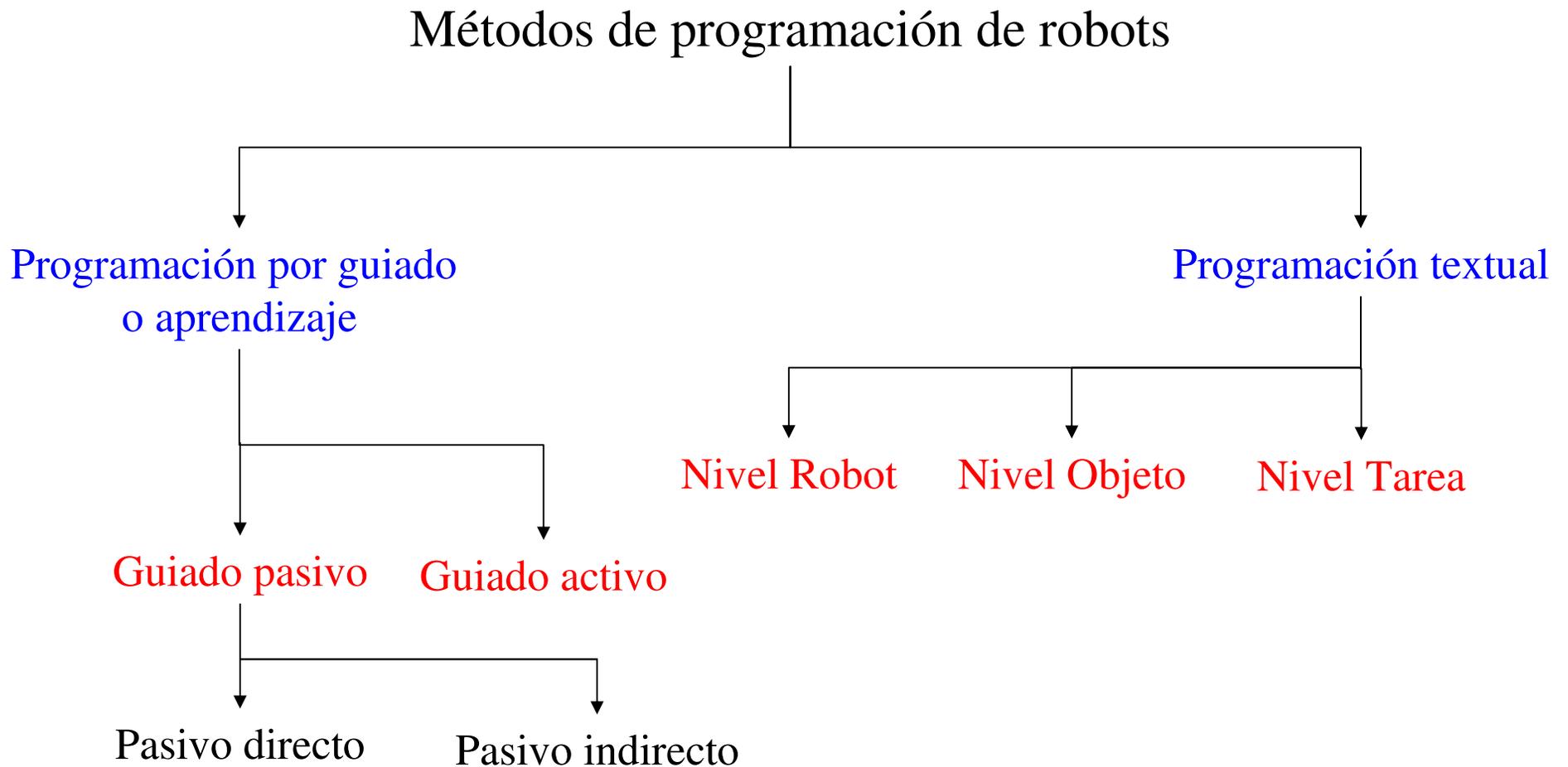
- **Falta de normalización**

- ▶ Cada fabricante ha desarrollado su método
- ▶ Hay orígenes y patrones comunes que dan lugar a semejanzas:
AL Universidad de Stanford 1974
- ▶ Intento de normalización: **IRDATA: Universidad de Karlsruhe**
 - Reconocido como estándar por la Norma VDI alemana.
 - Código intermedio entre el sistema de programación y el lenguaje propio del robot



Métodos de Programación de robots

● Clasificación





Métodos de Programación de robots

- **Programación por Guiado**

- ▶ Se lleva al robot por el camino que se desea que repita posteriormente en modo automático.

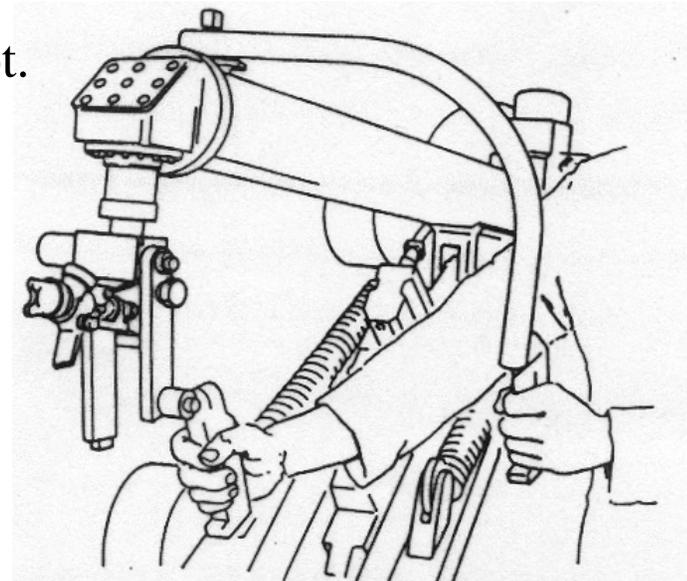
- **Programación por Guiado: Guiado Pasivo**

- ▶ El programador aporta la energía para mover el robot.

- **Programación por Guiado: Guiado Pasivo Directo**

- ▶ Se mueve directamente el extremo del robot.

La unidad de control **registra** posiciones de forma automática, realizando un **muestreo** de la trayectoria con un intervalo de tiempo determinado.

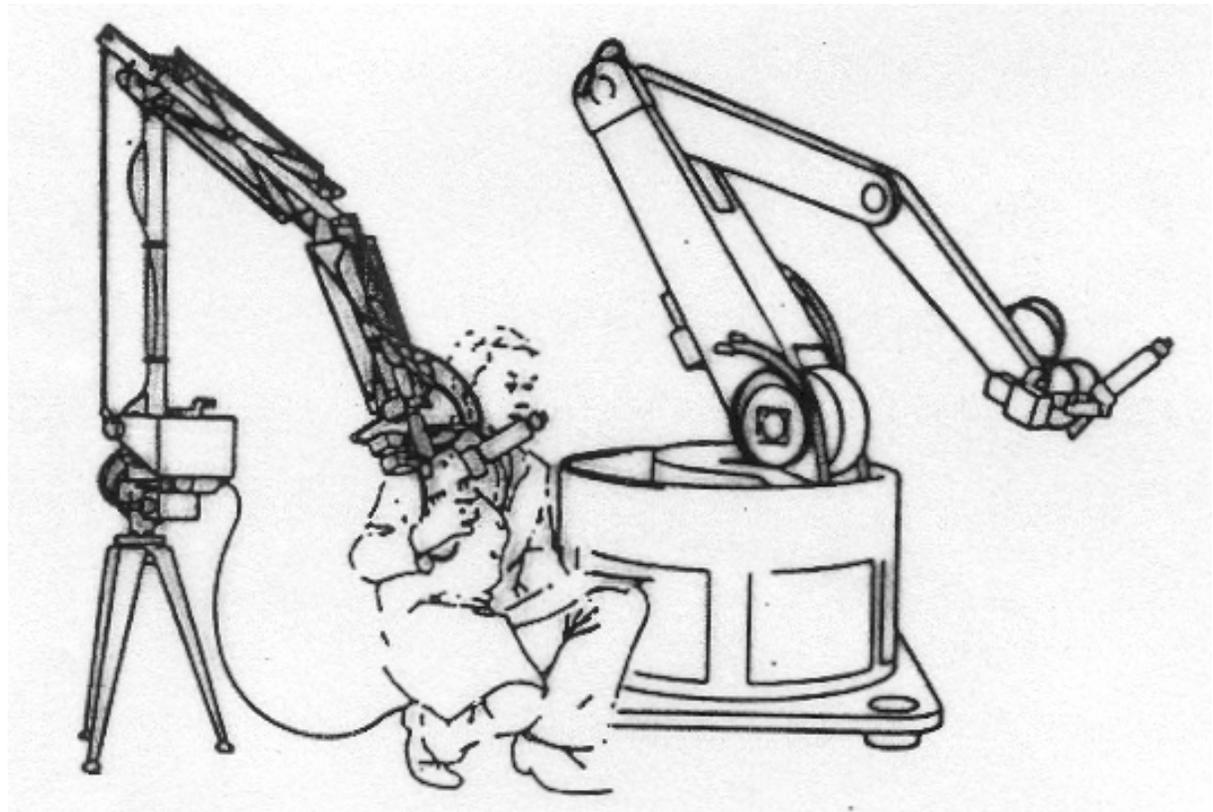




Métodos de Programación de robots

- **Programación por Guiado: Guiado Pasivo Indirecto o por Maniquí**
 - ▶ En vez del robot real se mueve un maniquí con su misma configuración cinemática pero mucho más ligero y fácil de mover.

El maniquí necesitará los sensores necesarios para leer las posiciones de las articulaciones





Métodos de Programación de robots

- **Programación por Guiado: Guiado Activo**

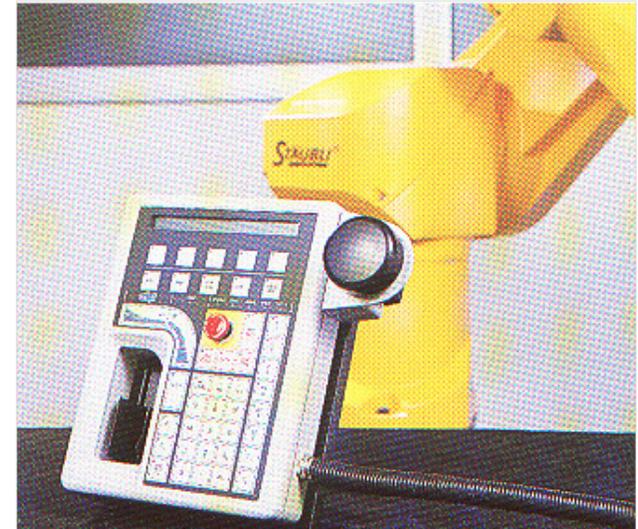
- ▶ Se utiliza el propio sistema de accionamiento del robot, controlado desde una botonera o un JoyStick.

- ASEA, Cincinnati Milacron.

- Se registran únicamente la posiciones del robot que se indiquen expresamente



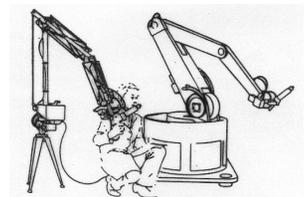
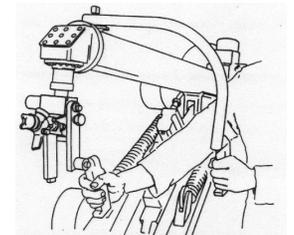
Fig. 2
El panel de control KUKA (KCP) con ratón 6-D integrado





Métodos de Programación de robots

- **Programación por Guiado: Inconvenientes**
 - ▶ Necesidad de disponer del robot y su entorno durante la programación de la aplicación.
 - ▶ Elevado coste económico.
 - ▶ Mayor riesgo de accidentes.
 - ▶ Necesidad de parar la instalación para cambiar la aplicación a ejecutar.
 - ▶ Normalmente no se genera una documentación escrita y comentada de los programas realizados.
 - ▶ Se dificulta la modificación y depuración de grandes programas.





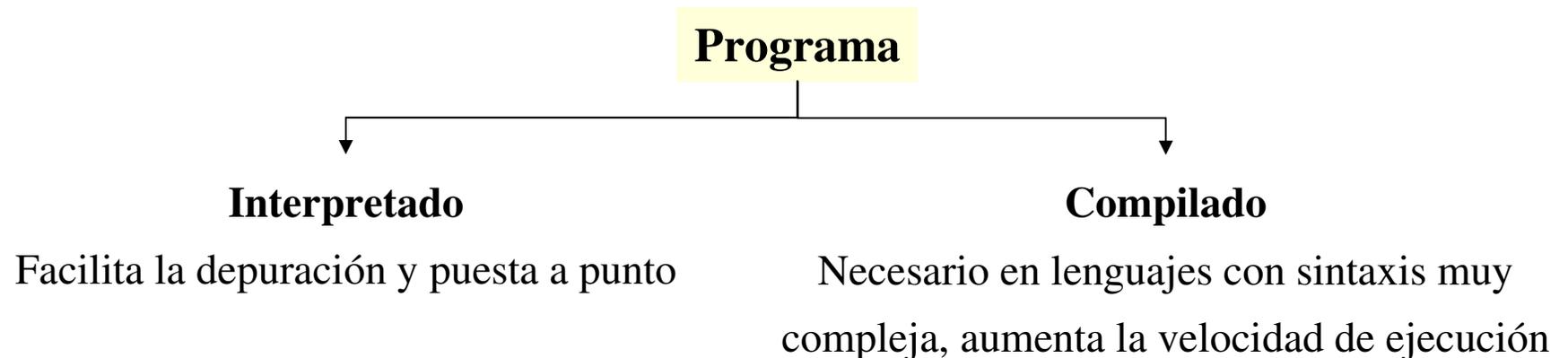
Métodos de Programación de robots

● **Programación Textual**

▶ Se basa en la existencia de un lenguaje formal de programación para indicar los comandos al robot.

▶ Programa

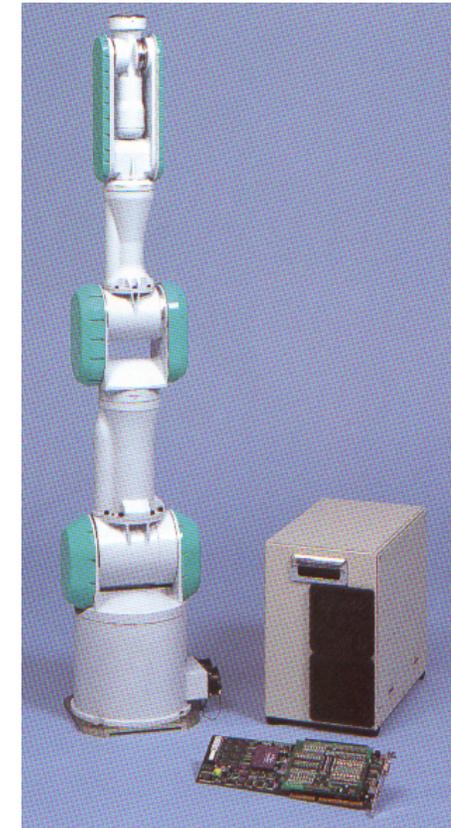
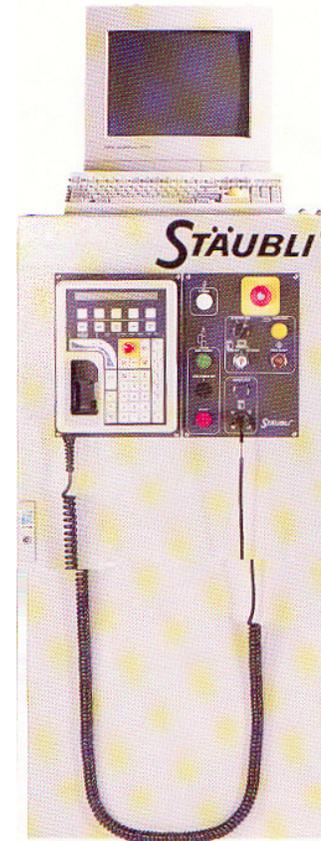
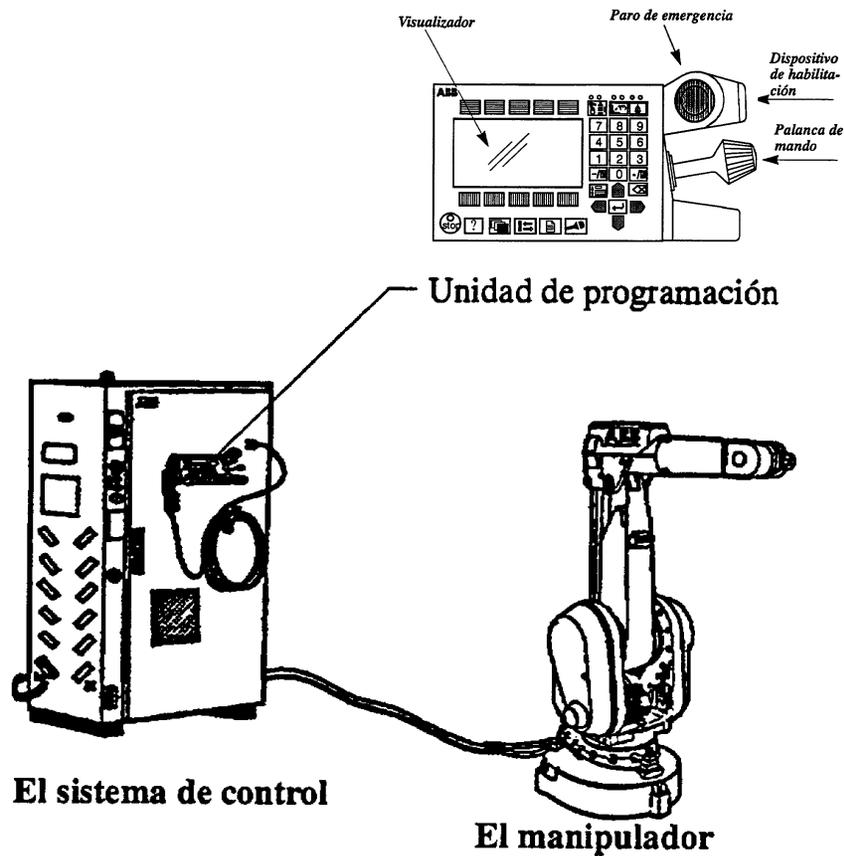
- Secuencia de ordenes
- Editadas y escritas por el usuario
- Ejecutadas por el robot
- Modos en que se procesa un programa





Métodos de Programación de robots

● Programación Textual

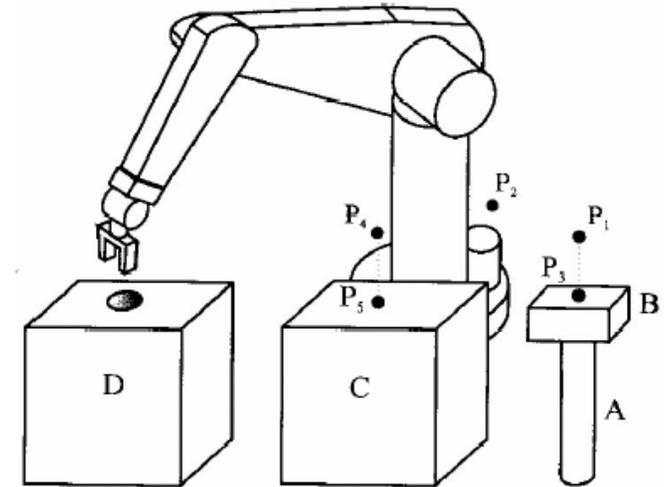




Métodos de Programación de robots

● Programación Textual

Dado el robot y el entorno de la figura se pretende situar La pieza A, sobre la que se apoya la pieza B, en el interior de la pieza D.



Las instrucciones hacen referencia a las acciones que debe ejecutar el robot

Nivel Robot

```
Move P1
Abrir_pinza
Wait 0.5
Set Input1
```

ABB RAPID, V+,
LM, IBM AML,
AL, VAL II

Las instrucciones hacen referencia al estado en que deben quedar los objetos manipulados

Nivel Objeto

```
...
Situar B sobre C
Situar A dentro de D
...
```

MIT LAMA,
IBM
AUTOPASS,
RAPT

Las instrucciones hacen referencia a un objetivo (total o parcial) a conseguir

Nivel Tarea

```
Ensamblar A con D
```



Requerimientos del sistema de programación

- ▶ Entorno de programación
- ▶ Modelado del entorno
- ▶ Tipos de datos
- ▶ Manejo de E/S: digitales y analógicas
- ▶ Control de movimiento
- ▶ Control del flujo de ejecución del programa



Requerimientos del sistema de programación

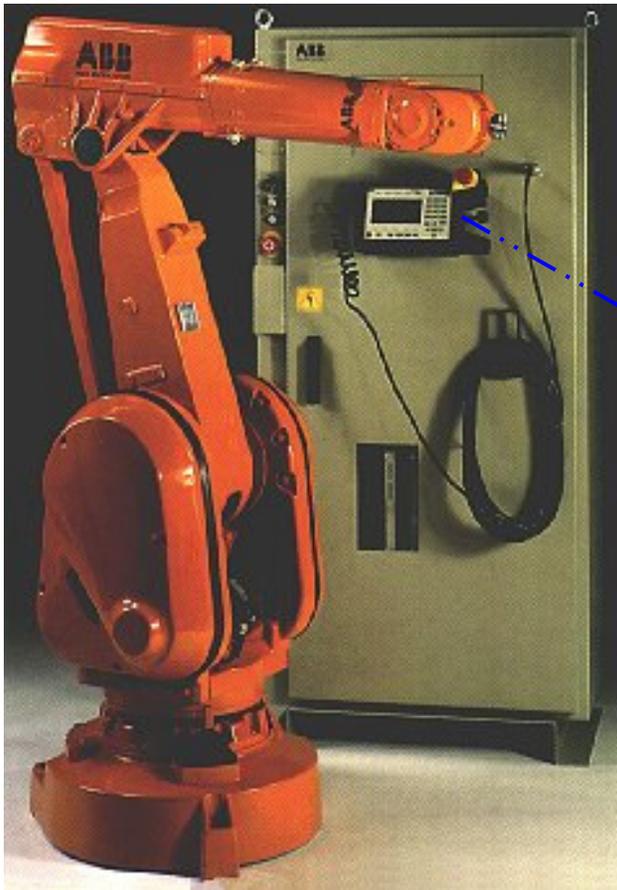
- **Entorno de programación**

- ▶ Su objetivo es facilitar la programación de tareas.
- ▶ La interacción con el entorno conlleva que la depuración de los programas sea un proceso de prueba y error.
- ▶ En la mayoría de los casos son interpretados pudiendo realizar una ejecución paso a paso de lo programado.
- ▶ Los sistemas actuales son multitarea permitiendo el control simultáneo de varios robots y otros sistemas.

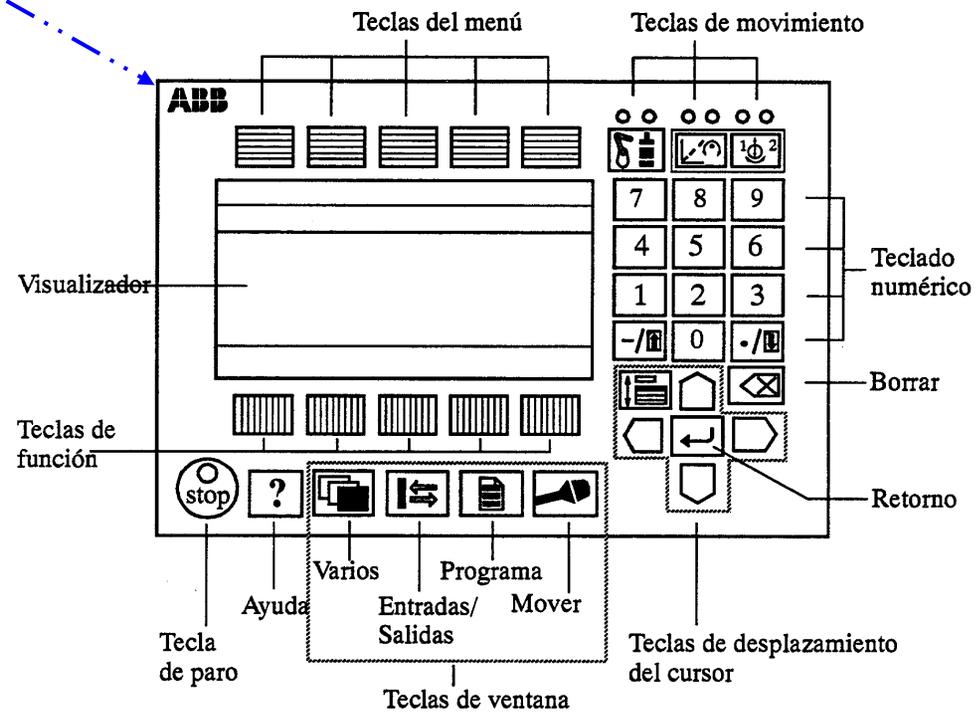


Requerimientos del sistema de programación

- Entorno de programación: **Programación por guiado**



- Paleta de programación
- Menús guiados
- Depuración de programas





Requerimientos del sistema de programación

- Entorno de programación: Programación Textual



- RAPID SyntaxChecker
(Analizador sintáctico fuera de línea)

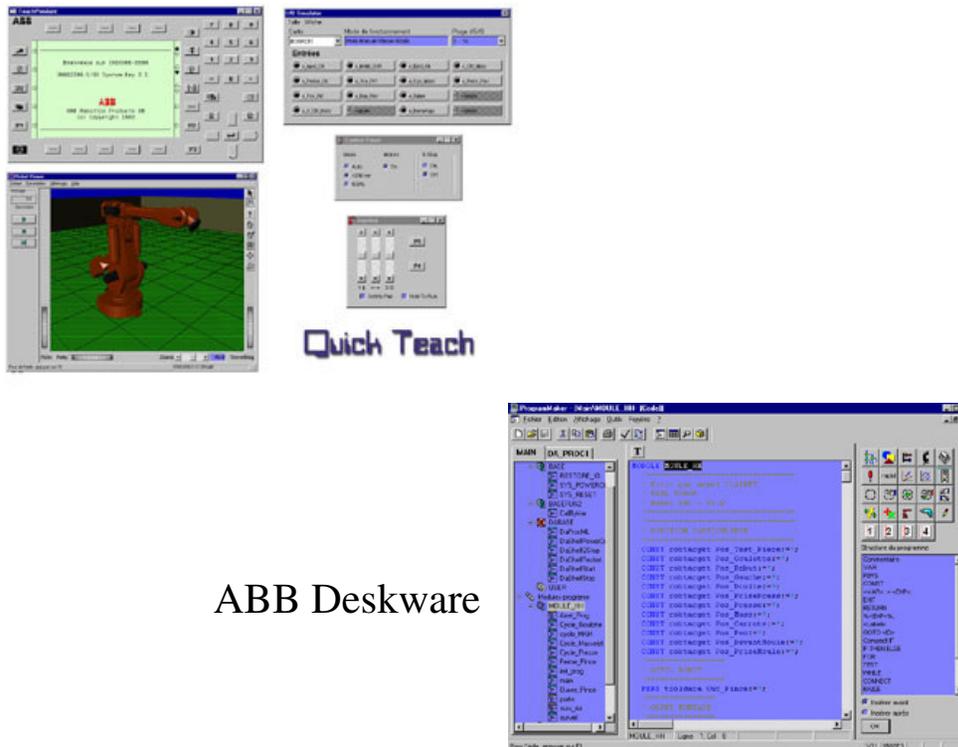
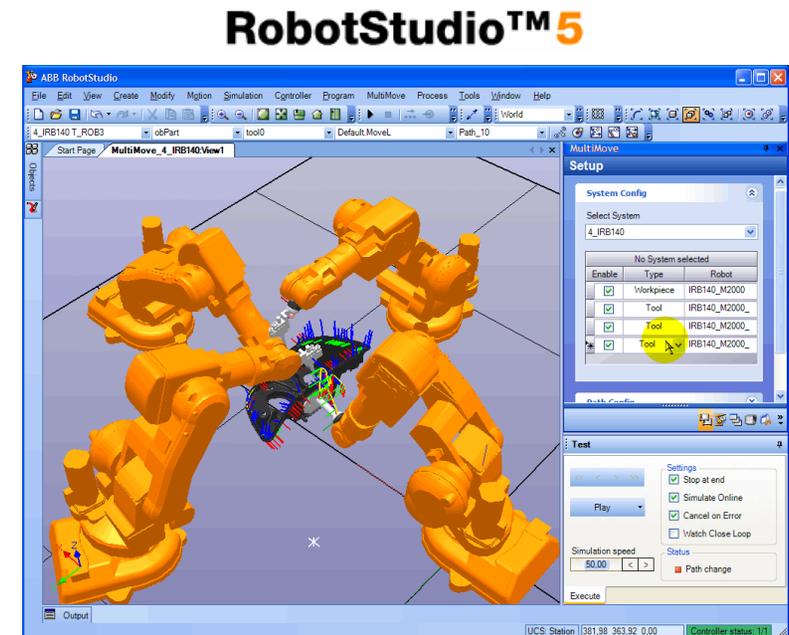


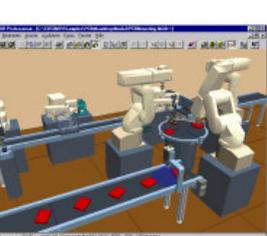
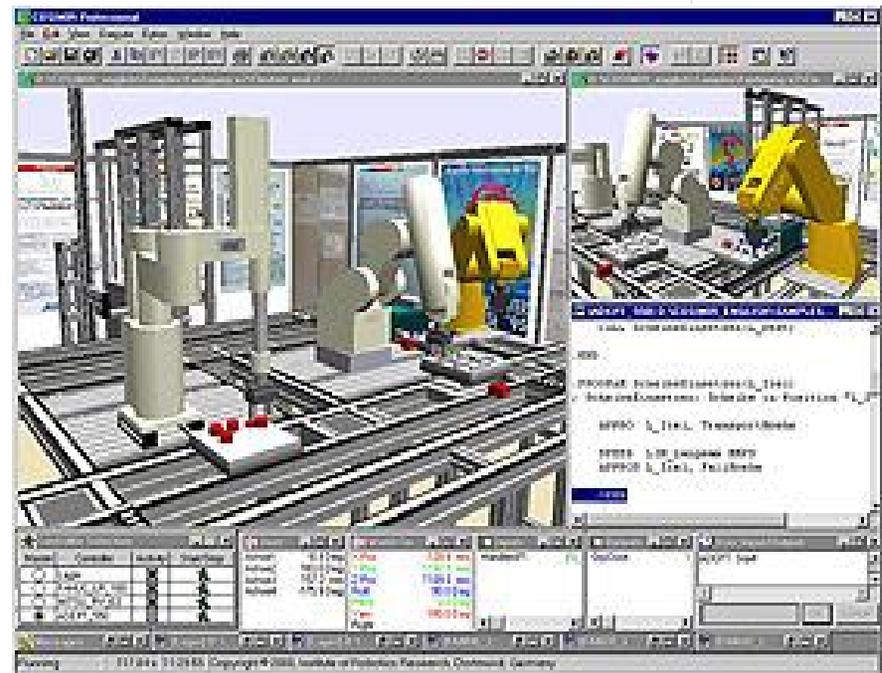
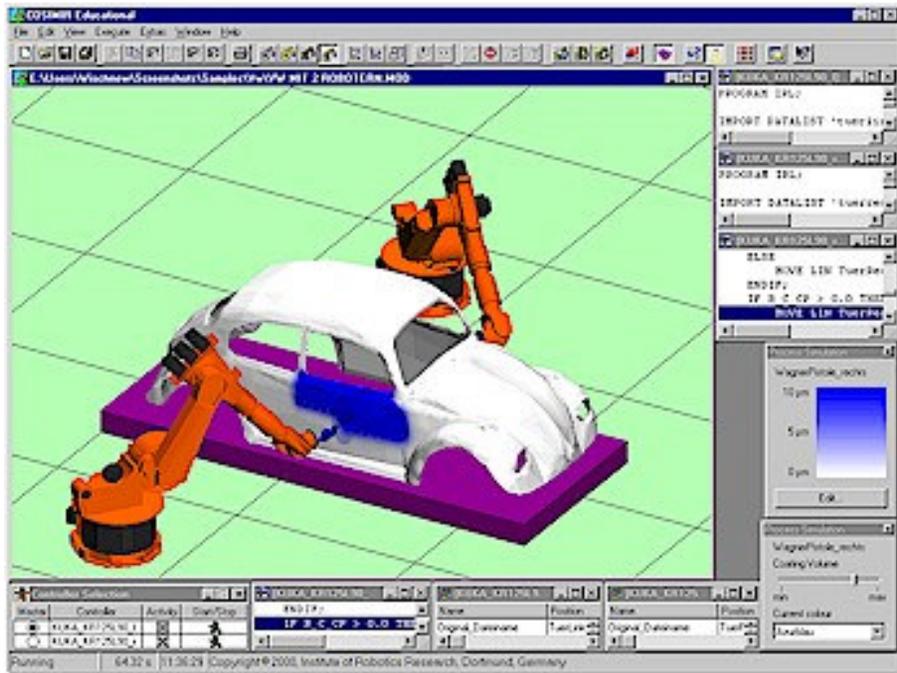
ABB Deskware





Requerimientos del sistema de programación

- Entorno de programación: **Programación Textual**



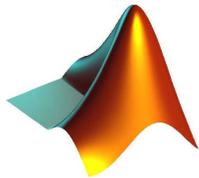
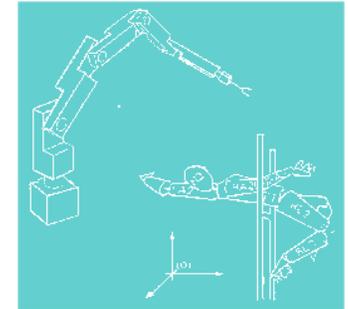


Requerimientos del sistema de programación

- Entorno de programación: **Programación Textual**

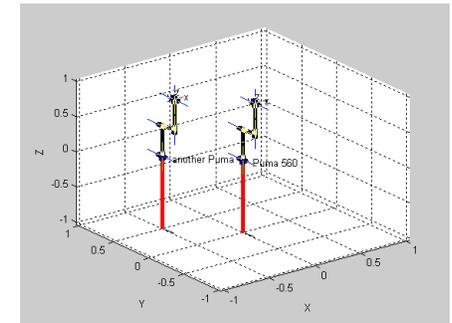
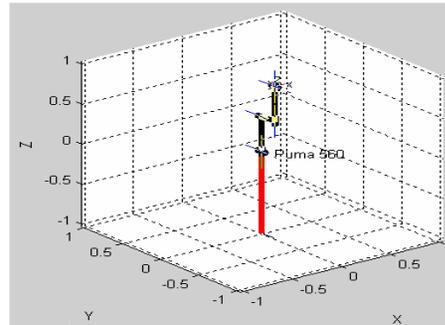


SPACELIB© es una librería para la cinemática y dinámica 3D de sistemas de cuerpos rígidos.

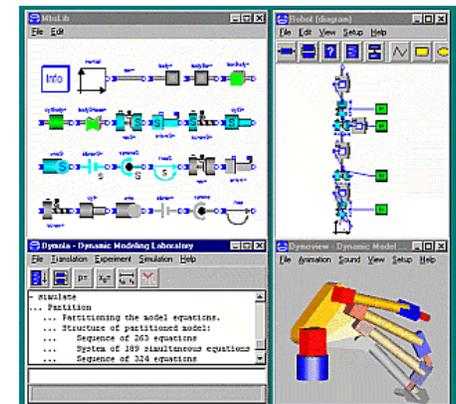


Matlab Robotics Toolbox

Se puede usar para la cinemática y la dinámica de los manipuladores y para la generación de trayectorias



Dymola (Dynamic Modeling Laboratory) es basado en Modelica ® que da acceso único a librerías desarrolladas por expertos de todo el mundo.





Requerimientos del sistema de programación

● Modelado del Entorno

► Es la representación que tiene el programa del robot de los objetos con los que trabaja:

☛ Características geométricas

Posición y orientación en los sistemas más clásicos.

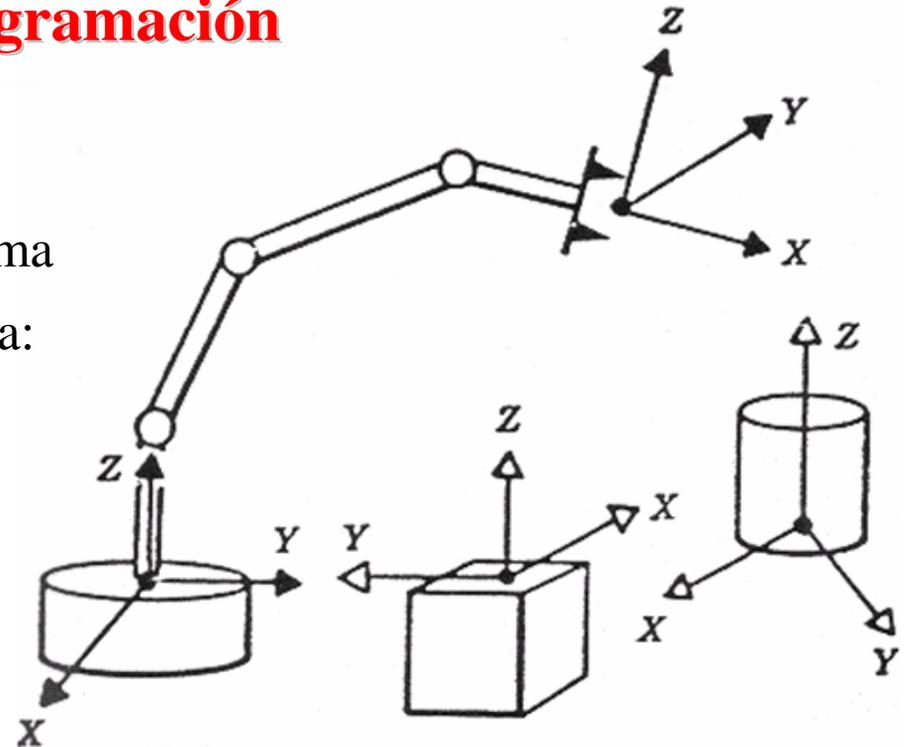
- Asignando a cada objeto un sistema de referencia solidario.
- En coordenadas articulares (q_1, \dots, q_n)
- En coordenadas cartesianas ($X, Y, Z, Q_1, Q_2, Q_3, Q_4$)

☛ Características físicas

Pesos e inercias para definir las cargas

☛ Características dimensionales

Formas y dimensiones (modelado 3D)





Requerimientos del sistema de programación

- **Tipos de Datos**

- ▶ **Datos convencionales simples**

Números (Enteros, Reales), Booleanos (verdadero/falso), Cadenas de caracteres

- ▶ **Datos compuestos**

Vectores (datos del mismo tipo), Estructuras (datos de distinto tipo)

- ▶ **Datos para representar la posición y orientación**

Coordenadas internas del robot (articulares), Coordenadas externas (o del usuario)

- ▶ **Datos para representar cargas**

Posición y orientación, Orientación de los ejes principales de inercia, Masa y momentos de inercia.

- ▶ **Datos para representar herramientas**

Posición y orientación de su punto de trabajo, Carga



Requerimientos del sistema de programación

● **Manejo de Entradas / Salidas**

▶ **Para comunicar y sincronizar el robot con otras máquinas o procesos de fabricación**

☞ Mediante señales binarias de entrada/salida

- Instrucciones de activación y desactivación de salidas
- Lectura de las entradas
- Comienzo y fin de ciclo automático
- Comienzo y fin de ejecución automático
- Generación de interrupciones con tres niveles:

▶ **Interrupciones Movimiento protegido o monitorizado**

☞ Tres niveles de interrupción

- Nivel 1: Se finaliza el ciclo principal en ejecución
- Nivel 2: Se finaliza el movimiento en curso
- Nivel 3: Parada de emergencia lo más rápido posible

⊖ La rutina de tratamiento de la interrupción se ejecutará cuando el robot ya esté parado.



Requerimientos del sistema de programación

- **Manejo de Entradas/Salidas: Interrupciones**

Main ()

```
1.- Move P1
2.- Move P2
3.- Grip On
4.- Wait 0.5
5.- Move P3
6.- Grip Off
7.- Move P4
8.- Set Output1
9.- Wait 2
10.- ~ ~ ~ ~ ~
```

TRAP Input 1

```
1.- Reset Output1
2.- Reset Output2
3.- Move P100
4.- Write "Reparar Pinza"
5.- Wait Input20
5.- ~ ~ ~ ~ ~
```



Requerimientos del sistema de programación

● **Manejo de Entradas / Salidas**

▶ **Comunicaciones más sofisticadas**

☞ **Punto a punto con una conexión serie**

- Carga y descarga de programas
- Monitorización y supervisión
- Impresora línea serie

☞ **Buses de campo (PROFIBUS, CAN BUS, etc.)**

☞ **Redes de área local (LAN)**

☞ **Redes de área extensa (WAN)**



Requerimientos del sistema de programación

● **Manejo de Entradas / Salidas**

- ▶ **Otra función** de la E/S es la integración de la información aportada por sensores.
 - ☞ **Modificar la trayectoria: sensores de F/P**
 - ☞ **Obtener la identidad y posición de objetos y sus características: sistema de visión artificial, láser, etc.**
 - ☞ **Detección de obstáculos: ultrasonidos, láser, infrarrojos, inductivos, capacitivos, etc.**

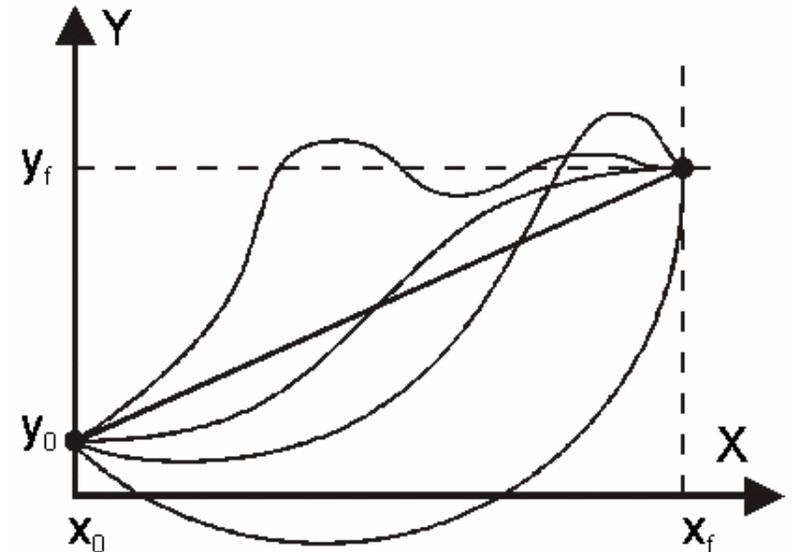


Requerimientos del sistema de programación

● Control del movimiento del robot

▶ Especificación del movimiento del robot

- ☞ Punto de destino (posición y orientación)
- ☞ Tipo de trayectoria espacial: articulares, recta, recta modificada, arco, círculo, etc.
- ☞ Velocidad media del recorrido
 - En la propia instrucción
 - Definición global
- ☞ Precisión en los puntos de paso
- ☞ Condiciones del movimiento
 - Interrupción: movimiento protegido o monitorizado
 - Modificación: movimiento acomodaticio ante los valores indicados por sensores externos





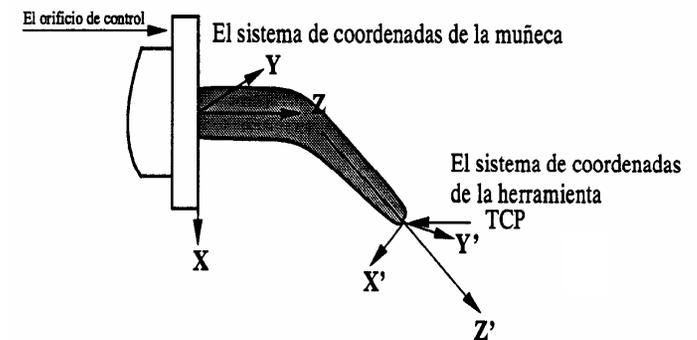
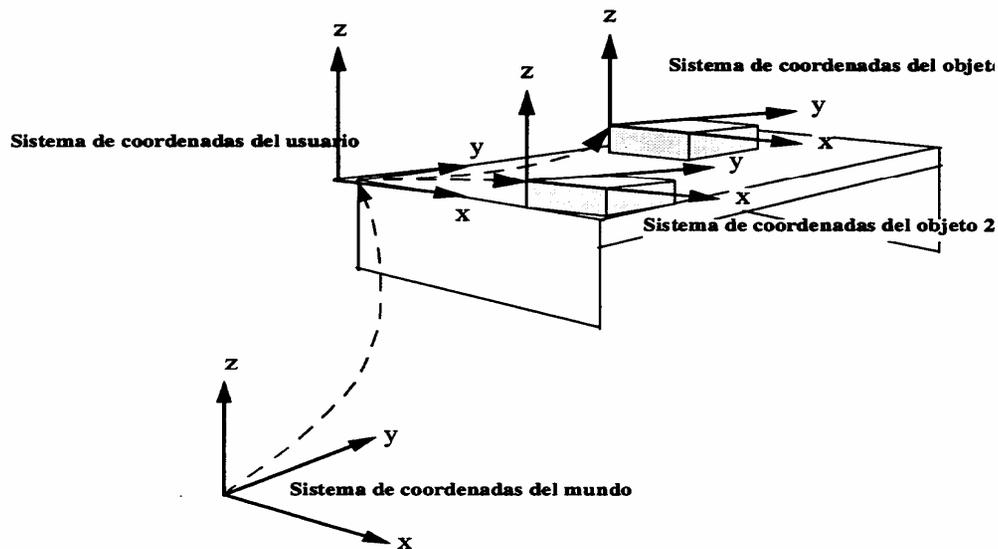
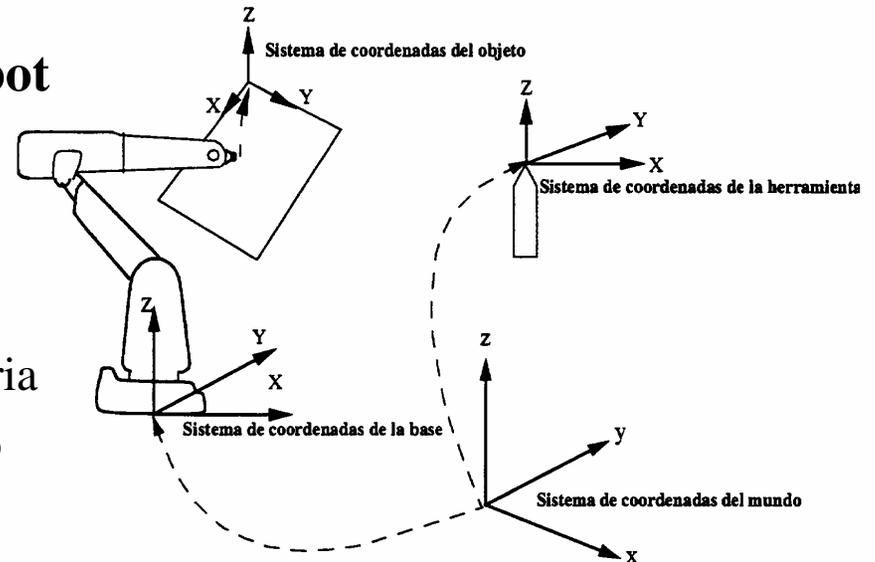
Requerimientos del sistema de programación

● Control del movimiento del robot

▶ Especificación del movimiento del robot

👉 Sistema de coordenadas utilizado

- Base: referida al sistema del mundo
- Mundo
- Herramienta: en el extremo o estacionaria
- Objeto: referido a un sistema de usuario





Requerimientos del sistema de programación

- **Control del flujo de ejecución**

- ▶ Permitir al programador especificar el flujo de operaciones del robot en función de una serie de condiciones lógicas.

- ✚ **Tradicionalmente**

- GOTO etiqueta,[condición]

- Salto incondicional

- Salto condicionado

- Subprogramas

- Sin paso de parámetros

- Todas las variables globales



Requerimientos del sistema de programación

● **Control del flujo de ejecución**

▶ Permitir al programador especificar el flujo de operaciones del robot en función de una serie de condiciones lógicas.

■ **Instrucciones de condición**

IF ... THEN ... ELSE

Selección por casos: CASE

Bucles: FOR, WHILE, REPEAT

Procesamiento en paralelo, multitarea

Trabajar con varios procesos, paso de mensajes, sincronización, temporizaciones, etc.

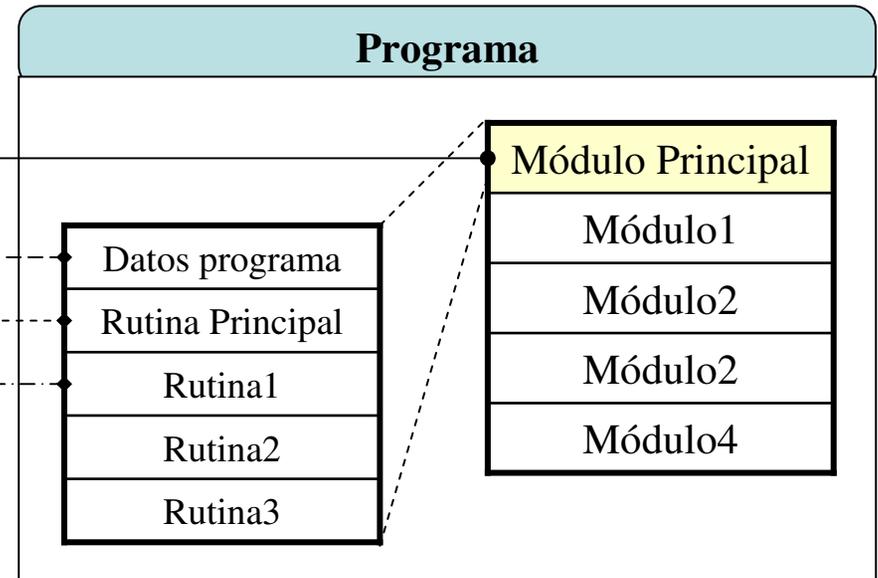
Tratamiento de interrupciones



Lenguajes de programación

● **RAPID (Robotics Application Programming Interactive Dialogue)**

```
%%%  
VERSION:1  
LANGUAGE:ENGLISH  
%%=  
MODULE Module1 ←  
  
  declaraciones ←  
  
  PROC procl() ←  
    . . .  
  ENDPROC  
  
  PROC main() ←  
    . . .  
    . . .  
  ENDPROC  
  
ENDMODULE
```



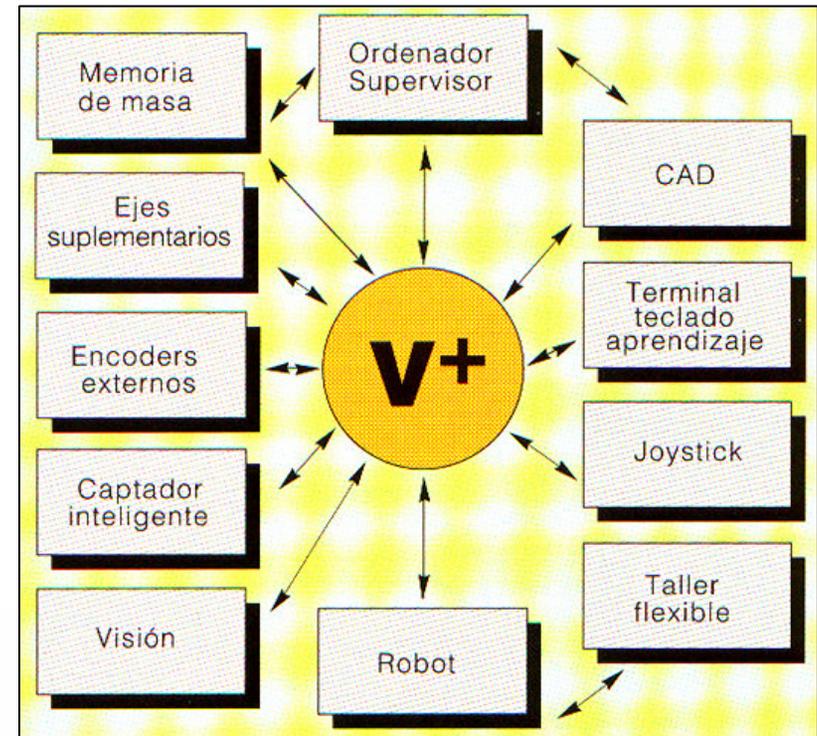
Más información:
Práctica 3
ABB RAPID



Lenguajes de programación

- **VAL (Victor Assembly Language)**

- Desarrollado por Victor Scheinman in 1976 para UNIMATION en sus robots de la serie PUMA
- En 1984 se amplió al VAL II
- Recientemente al VAL+ en su serie RX

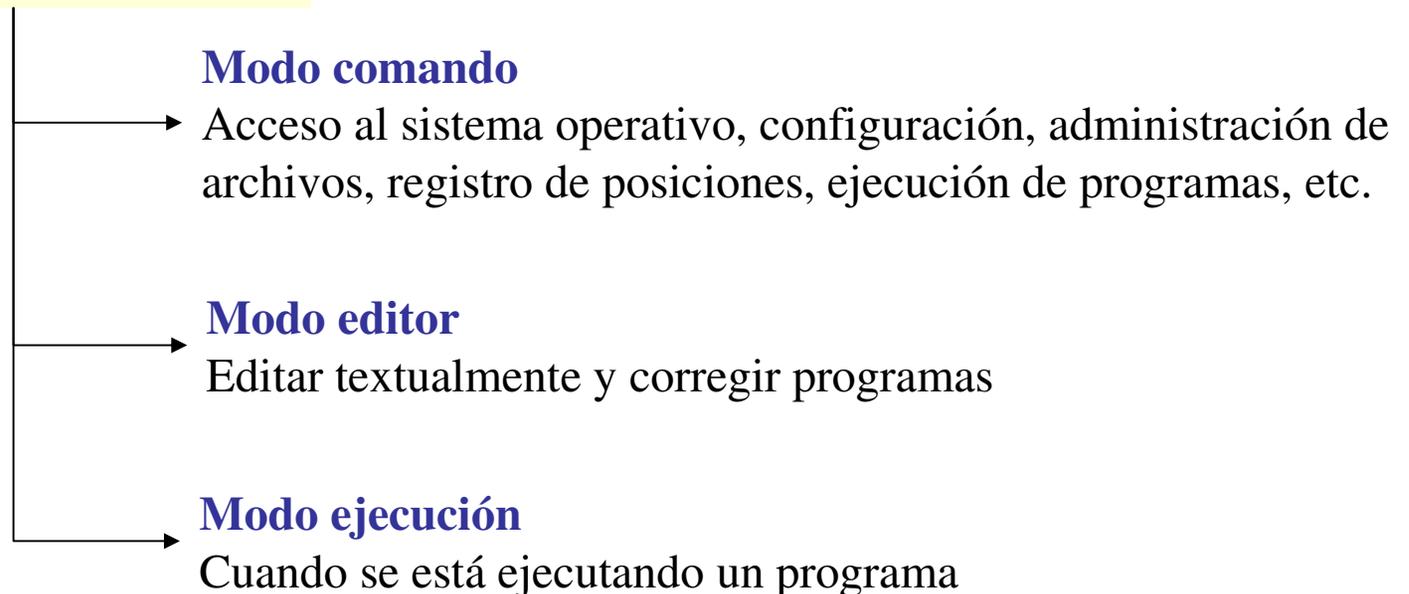




Lenguajes de programación

- **VAL (Victor Assembly Language)**
 - ▶ Todo el sistema se maneja desde una consola.

Modos de funcionamiento VAL II





Lenguajes de programación

• VAL (Victor Assembly Language): Ejemplo

- El robot se encuentra esperando en un punto de reposo (punto A) la activación de la entrada digital 1.
- En cuanto la entrada digital 1 pasa al estado UNO, el robot se aproxima al punto B para coger la pieza.
- El robot lleva la pieza al punto C pasando por una zona segura (punto D).
- El ciclo finaliza en el punto de reposo.

```
PROGRAM EJEMPLO
10 OPENI
20 SPEED 100 MMPS ALWAYS
30 MOVE #A
40 WAIT SIG(1001)
50 SPEED 80 MMPS
60 APPRO B,50
70 MOVES #B
80 BREAK
90 CLOSE
100 SPEED 80 MMPS
110 DEPARTS B,50
120 MOVE D
130 SPEED 80 MMPS
140 APPRO C,50
150 MOVES #C
160 OPENI
170 DEPARTS 50
END
```



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 7 – Miércoles 10 de Mayo 2006

Programación de Robots



Automatización Industrial - II

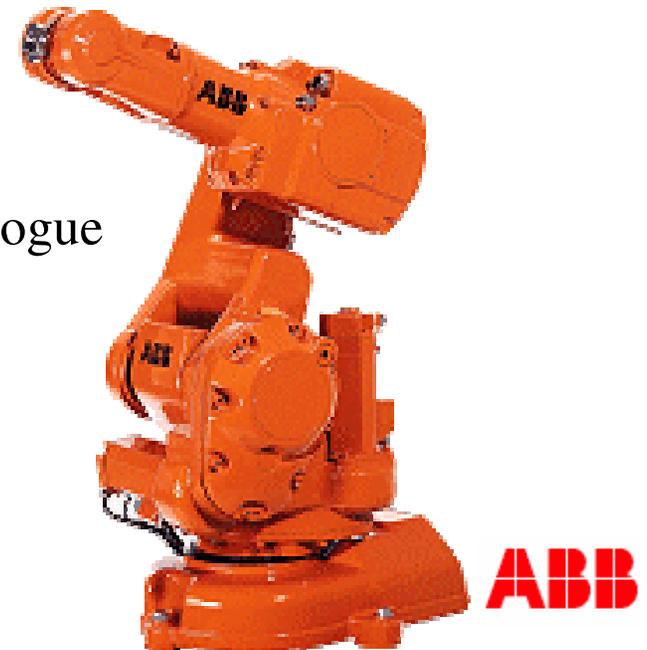
56 – 10569 2º Cuatrimestre 2006

Práctica 3– Lunes y Miércoles 8 & 10 de Mayo 2006



Lenguaje RAPID

Robotics Application Programming Interactive Dialogue

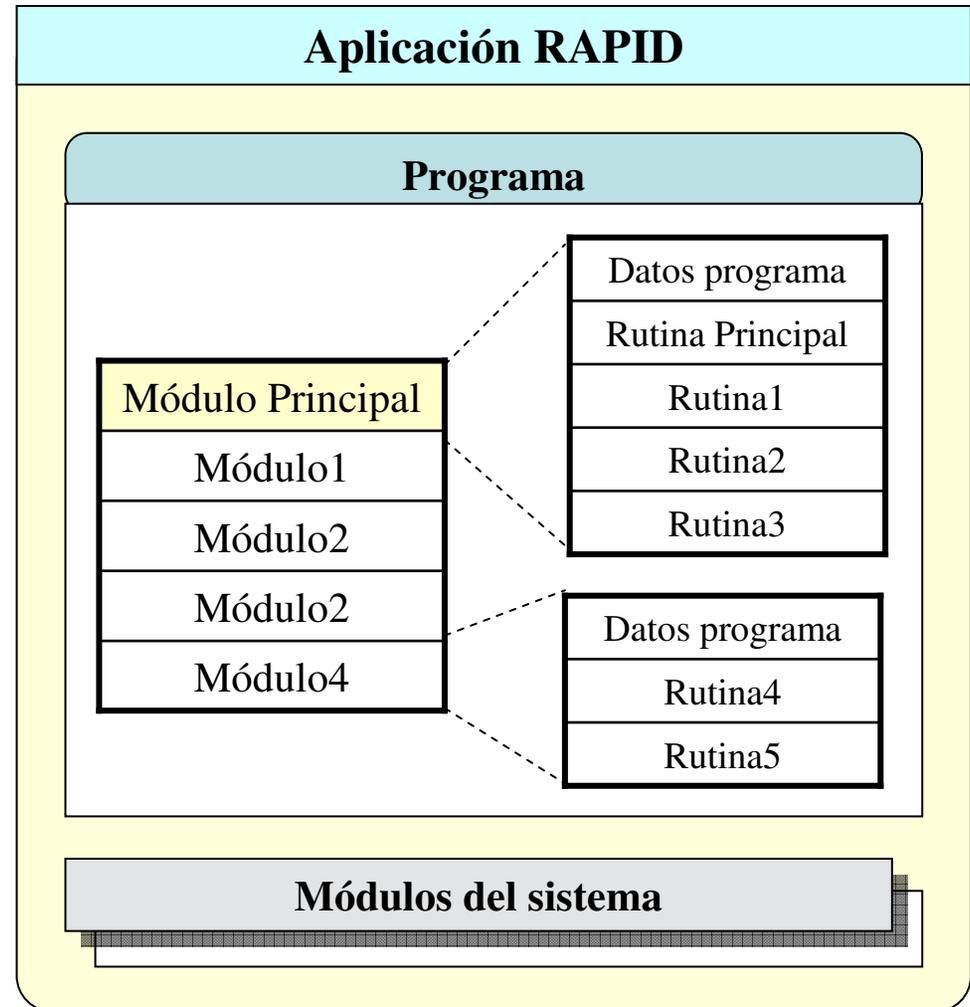




Lenguaje RAPID

● Estructura del lenguaje

- ▶ RAPID es un lenguaje de programación textual de alto nivel desarrollado por la empresa ABB.
- ▶ Una aplicación RAPID consta de un programa y una serie de módulos del sistema.





Lenguaje RAPID

● Programa RAPID

► El programa es una secuencia de instrucciones que controlan el robot y en general consta de tres partes:

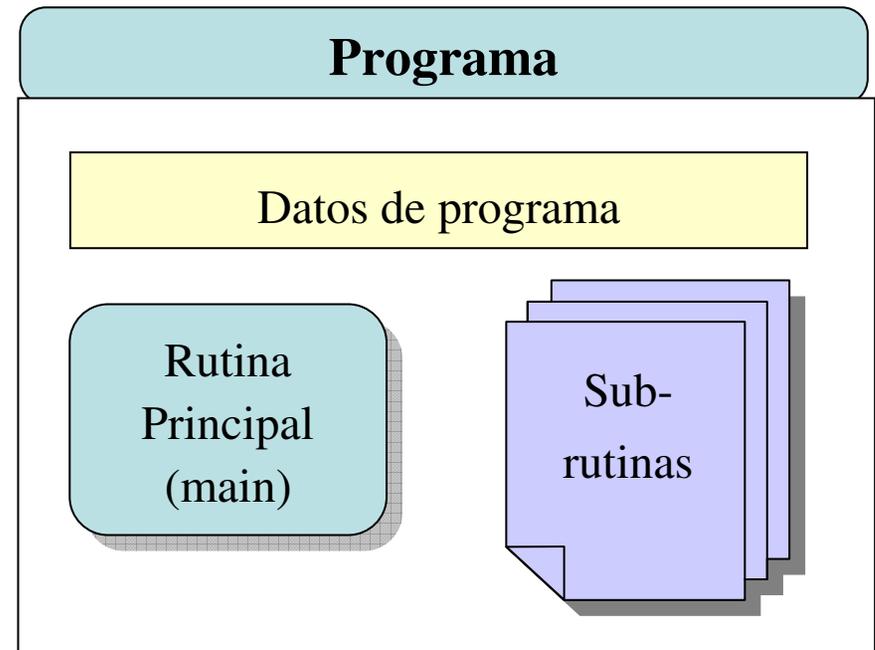
☛ **Una rutina principal (main):**
Rutina donde se inicia la ejecución.

☛ **Un conjunto de sub-rutinas:**

Sirven para dividir el programa en partes más pequeñas a fin de obtener un programa modular.

☛ **Los datos del programa:**

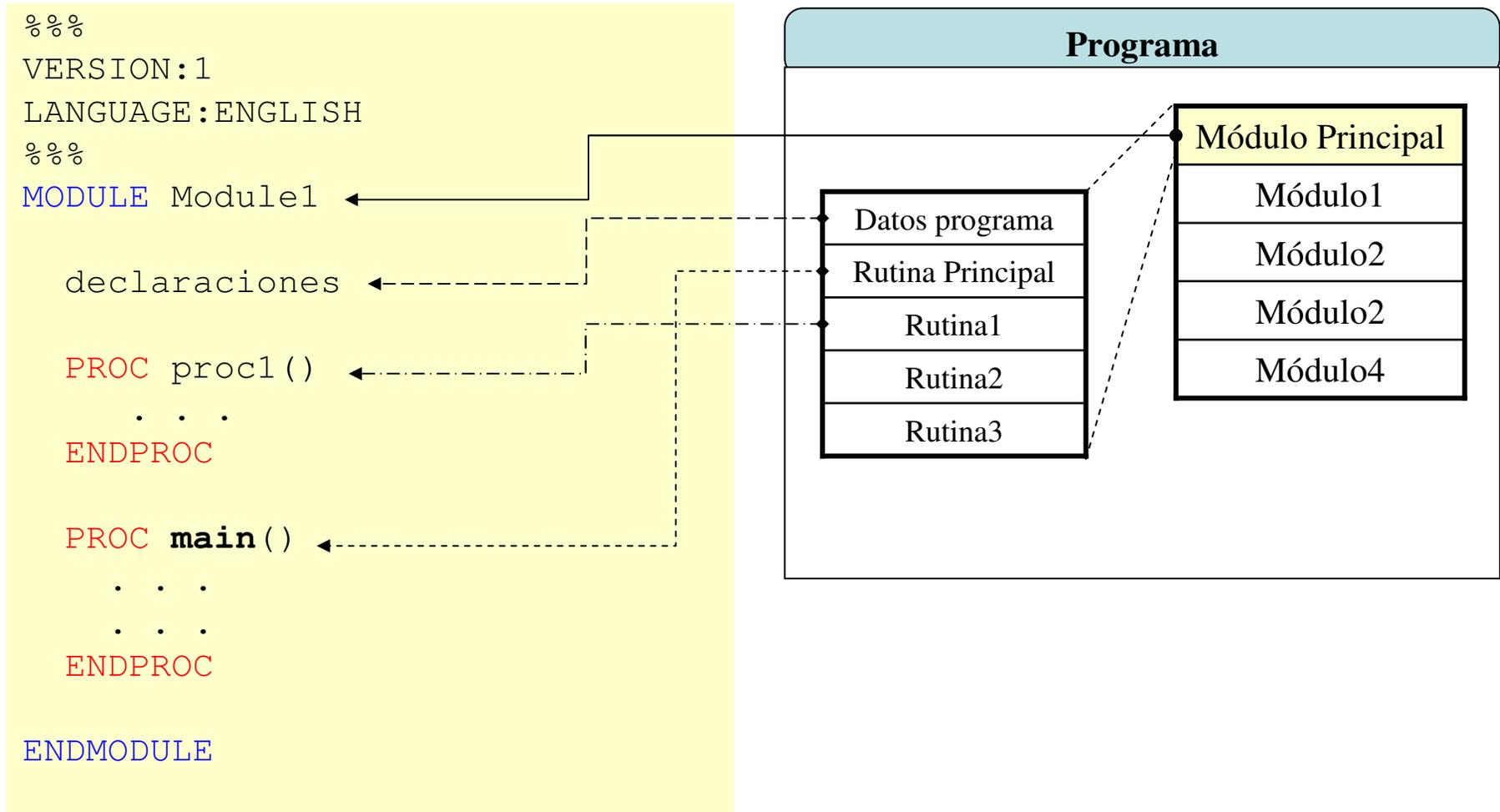
Definen posiciones, valores numéricos, sistemas de coordenadas, etc.





Lenguaje RAPID

● Programa RAPID





Lenguaje RAPID

● **Programa RAPID**

```
%%%  
VERSION:1  
LANGUAGE:ENGLISH  
%%%  
MODULE EJEMPLO  
  CONST robtarget A:=[[0,0,0],[0,0,0,0],[0,-1,0,0], [9E+09,...]];  
  
  CONST tooldata pinza:= [TRUE, [[0,0,0],[1,0,0,0]],  
                           [0,[0,0,0],[1,0,0,0],0,0,0]];  
  
  PROC cerrar_pinza()  
    Set spinza;  
  ENDPROC  
  
  PROC coger_pieza()  
    MoveJ B1,v100,z5,pinza;  
    MoveL B,v80,fine,pinza;  
    cerrar_pinza;  
  ENDPROC
```



Lenguaje RAPID

● **Programa RAPID**

```
PROC main()  
  CONST dionum listo:=1;  
  abrir_pinza;  
  WHILE TRUE DO  
    MoveJ A,v100,fine,pinza;  
    WaitDI econtrol,listo;  
    coger_pieza;  
    MoveL B1,v80,z5,pinza;  
    MoveJ D,v100,z100,pinza;  
    MoveJ C1,v100,z5,pinza;  
    MoveL C,v80,fine,pinza;  
    abrir_pinza;  
    MoveL C1,v80,z5,pinza;  
  ENDWHILE  
ENDPROC  
ENDMODULE
```



Lenguaje RAPID

● Elementos básicos

▶ Identificadores:

Permiten nombrar módulos, rutinas, datos y etiquetas.

Ejemplo:

```
MODULE nombre_módulo  
PROC      nomre_rutina ()  
VAR pos   nombre_dato; nombre_etiqueta:
```

- El primer carácter es siempre una letra.
- Longitud máxima 16.
- Diferencia entre mayúsculas y minúsculas.

▶ Palabras reservadas:

AND	BACKWARD	CASE	CONNECT	CONST	DEFAULT	DIV
DO	ELSE	ELSEIF	ENDFOR	ENDFUNC	ENDIF	ENDMODULE
ENDPROC	ENDTEST	ENDTRAP	ENDWHILE	ERROR	EXIT	FALSE
FOR	FROM	FUNC	GOTO	IF	INOUT	LOCAL
MOD	MODULE	NOSTEPIN	NOT	OR	PERS	PROC
RAISE	READONLY	RETRY	RETURN	STEP	TEST	THEN
TO	SYSMODULE	TRAP	TRUE	VAR	VIEWONLY	WHILE
WITH	XOR					



Lenguaje RAPID

● **Elementos básicos**

▶ **Espacios y caracteres de fin de línea:**

RAPID es un lenguaje **sin formatos**, en consecuencia los espacios pueden utilizarse en cualquier parte excepto en: identificadores, palabras reservadas, valores numéricos.

Los identificadores, las palabras reservadas y los valores numéricos deberán estar separados entre sí por un espacio, un carácter de fin de línea o un tabulador

▶ **Comentarios:**

Sirven para facilitar la comprensión del programa, ocupan una línea entera comenzando con el símbolo **!**, finaliza con un carácter de fin de línea.

```
! Esto es un comentario
```

▶ **Valores de cadena:**

Secuencia de caracteres entre comillas.

```
"Esto es una cadena"
```



Lenguaje RAPID

● **Los Datos**

▶ Los datos a manejar pueden ser definidos como:

Constantes: (**CONS**) representen datos de un valor fijo a los que no se puede reasignar un nuevo valor.

Variables: (**VAR**) son datos a los que se les puede asignar un nuevo valor durante la ejecución del programa.

Persistentes: (**PERS**) se trata de variables en las que cada vez que se cambia su valor durante la ejecución del programa, también se cambia el valor de su inicialización.



Lenguaje RAPID

● **Los Datos**

- ▶ Los datos se pueden definir según la cantidad de memoria que se necesita para almacenarlo:

Atómicos: En ellos solo se guarda un dato. No se puede dividir en otros más sencillos.

Registros: Es un tipo de dato en el que se guardan de una forma ordenada más de un dato. En lenguaje C sería un tipo de dato similar a las estructuras.



Lenguaje RAPID

● Tipos de Datos: Atómicos

- ▶ **num:** Se usa para los valores numéricos, ya sean enteros o reales

Ejemplo:

```
VAR num flujo := 0;
flujo := 2.34;
```

Valores válidos: 5 0.37 0.1E-5 -12.34

- ▶ **bool:** Se usa para designar valores lógicos. (verdadero/falso)

Valores posibles: TRUE y FALSE.

```
VAR bool <identificador>:= <valor>
      <valor>: TRUE / FALSE
      <expresión lógica>
```

Ejemplo:

```
VAR bool abrir:=TRUE;
abrir:=FALSE;
abrir:= reg1 > 1;
```

- ▶ **string:** Se usa para guardar cadenas de caracteres, que pueden tener como máximo 80 incluidas las comillas “ que son las que delimitan la cadena.

Ejemplo:

```
VAR string text;
text:= "Arranque del sistema";
```



Lenguaje RAPID

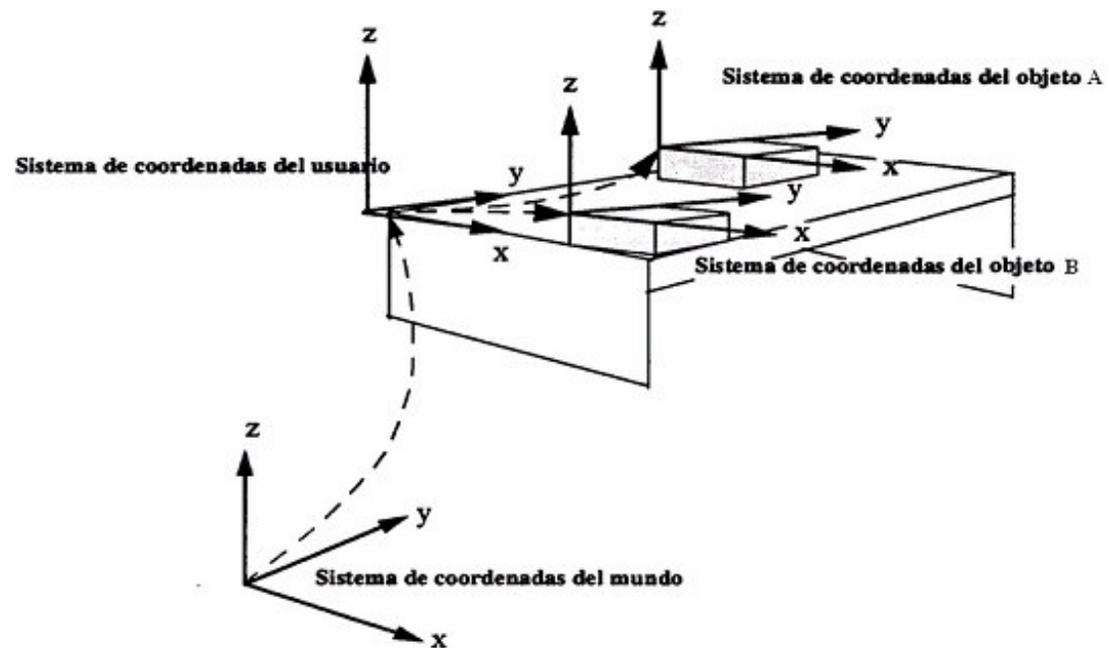
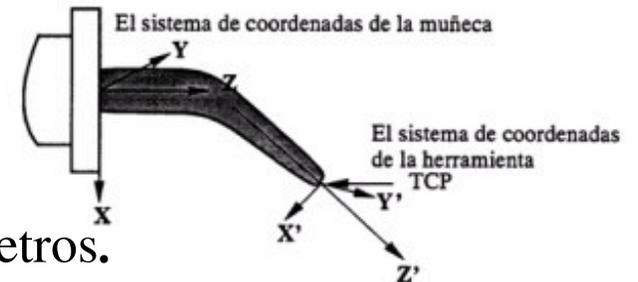
Tipos de Datos: Registros

- **pos:** Representar posiciones sólo X, Y y Z en milímetros.

x es de tipo num.

y es de tipo num.

z es de tipo num.



Ejemplo:

```
VAR pos posicion1;  
posicion1 := [500, 0, 940];  
posicion1.x := posicion1.x + 50;
```

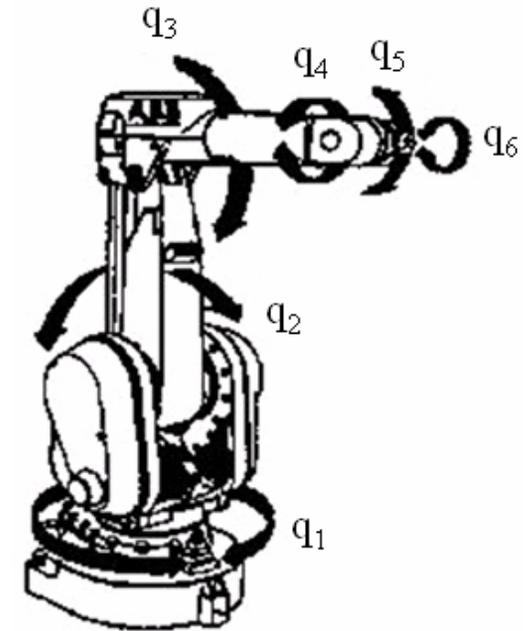


Lenguaje RAPID

- **Tipos de Datos: Registros**

► **orient:** Es un dato de tipo registro que guarda la orientación de algún elemento, como por ejemplo la de la herramienta. ABB usa cuaternios para almacenar la orientación.

(q1, q2, q3, q4,q5,q6) de tipo num.



► **pose:** Se usa para cambiar de un sistema de coordenadas a otro. Está compuesto por un dato pos y otro de tipo orient.

trans pos
rot orient

Ejemplo:

```
VAR pose pos1;  
pos1 := [[500, 100, 800], [1, 0, 0, 0]];  
pos1.trans := [650, -230, 1230];  
pos1.trans.y := -23.54;
```



Lenguaje RAPID

● Tipos de Datos: Registros

- ▶ **confdata:** Permite definir las configuraciones de los ejes del robot.

cf1: Cuadrante utilizado del eje 1.

cf4: Cuadrante utilizado del eje 4.

cf6: Cuadrante utilizado del eje 6.

Ejemplo:

```
VAR confdata conf10 := [1, -1, 0]
```

- ▶ **loaddata:** Sirve para describir la carga colocada en la muñeca del robot.

mass: peso de la carga en kilogramos.

cog: centro de gravedad de la carga.

aom: orientación de los ejes de inercia en el centro de gravedad.

ix, iy, iz: momentos de inercia de la carga alrededor de los ejes x, y, z en kgm^2 .

Ejemplo:

```
VAR loaddata pieza := [5, [50, 0, 50], [1, 0, 0, 0], 0, 0, 0];
```



Lenguaje RAPID

● Tipos de Datos: Registros

▶ **tooldata**: Describe las características de una herramienta.

robhold : Tipo bool que define si el robot sujeta la herramienta o no.

tframe : Sistema de coordenadas de la herramienta

Posición del TCP (x,y,z)

Orinetación. (q1,q2,q3,q4)

tload : Carga de la herramienta

Peso

Centro de gravedad (x,y,z)

Ejes de momento de la herramienta (q1,q2,q3,q4)

Momento de inercia de los ejes (x,y,z).

Ejemplo:

```
PERS tooldata pinza:=[TRUE, [[97,0,220],  
[0.924,0,0.383,0]],5, [-23,0,75], [1,0,0,0],0,0,0]]
```



Lenguaje RAPID

- **Tipos de Datos: Registros**

- ▶ **robtarget:** Sirve para definir la posición del robot y de sus ejes externos.

trans : Posiciones (x, y, z)
rot : Orientación de la herramienta.
robconf : Configuración de los ejes.
extax : posición de los ejes externos

Ejemplo:

```
VAR robtarget punto1;  
punto1 := [[500, 100, 800], [1, 0, 0, 0]];  
punto1.trans := [650, -230, 1230];  
punto1.trans.y := -23.54;
```

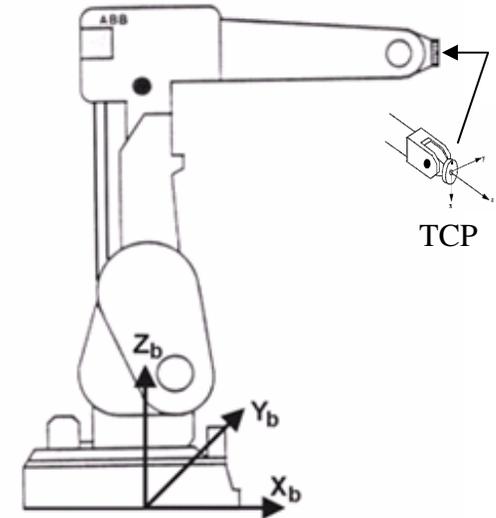


Lenguaje RAPID

● Tipos de Datos: Registros

► **speeddata**: Para especificar la velocidad a la que se moverán los ejes externos del robot.

- v_tcp** : Velocidad del TCP mm/s.
- v_ori** : Velocidad de reorientación del TCP grados/s.
- v_leax** : Velocidad de los ejes externos lineales en mm/s.
- v_reax** : Velocidad de los ejes externos rotativos grados/s.



Ejemplo: `VAR speeddata vmedia:=[1000, 30, 200, 15];`

Datos predefinidos:

v5 [5,5,50,5]
v10,v20,.....,v80
v100, v150, v200, v300,.....,v800
v1000, v1500, v2000, v2500, v3000
vmax [5000, 500, 5000, 500]



Lenguaje RAPID

● **Tipos de Datos: Registros**

- ▶ **zonedata:** Para especificar como debe terminarse una posición.

finep : Punto de paro o de paso bool
pzone_tcp : Radio de zona del TCP en mm.
pzone_ori : Tamaño de zona de reorientación mm.
pzone_eax : Zona de los ejes externos mm.
zone_ori : Tamaño de la zona de reorientación grados.
zone_leax : Tamaño de la zona ejes externos mm.
zone_reax : Tamaño zona ejes rotativos externos grados.

Ejemplo: `VAR zonedata trayec := [FALSE, 25, 40, 50, 5, 35 10];`

Datos predefinidos:

`z1 [1, 1, 1, 0.1, 1, 0.1]`

`z5, z10, z15, z20,, z100`

`z150`

`z200[200, 300, 300, 30, 300, 30]`



Lenguaje RAPID

- **Módulos**

- ▶ **Encabezado de archivo:**

Puede estar formado de diferentes datos y rutinas.

Uno de los módulos contiene el procedimiento de entrada, un procedimiento global de entrada llamado `main`.

- ▶ **Módulos del sistema:**

Sirven para definir datos y rutinas normales del sistema, como por ejemplo las herramientas.



Lenguaje RAPID

● **Módulos**

▶ **Declaración:**

```
MODULE      <nombre_módulo> [<Lista de atributos>]
              <Lista declaración de datos>
              <Lista declaración rutina>

ENDMODULE
```

```
[<Lista de atributos>]      :
```

SYSMODULE	:	Módulo del sistema.
NOSTEPIN	:	No se podrá entrar durante ejecución paso a paso.
VIEWONLY	:	No podrá ser modificado.
READONLY	:	No podrá ser modificado pero sí sus atributos.



Lenguaje RAPID

● Rutinas

▶ Tres tipos:

➤ Procedimientos:

```
PROC <nombre procedimiento> ( Lista de parámetros )  
  <Lista de declaraciones de datos>;  
  <Lista de instrucciones>;  
  ERROR <lista instrucciones>;  
ENDPROC
```

➤ Funciones:

```
FUNC <tipo valor dato> ( Lista de parámetros )  
  <Lista de declaraciones de datos>;  
  <Lista de instrucciones>;  
  RETURN dato;  
  ERROR <lista instrucciones>;  
ENDFUNC
```

➤ Interrupciones:

```
TRAP <nombre trap>  
  <Lista de declaraciones de datos>;  
  <Lista de instrucciones>;  
  ERROR <lista instrucciones>;  
ENDTRAP
```



Lenguaje RAPID

- **Expresiones del lenguaje**

Las expresiones se utilizan para evaluar un valor y así poder asignarlo a una variable o utilizarlo como argumento de una instrucción o de una rutina. Según el tipo de valor que devuelve la expresión se distinguen dos tipos:

► **Aritméticas:** Devuelven un valor numérico si operan con variables de tipo num y una cadena si operan con cadenas de caracteres.

Utilizan los operadores aritméticos: *, +, -, /, DIV (división entera), MOD (resto)

Ejemplo:

```
perimetro = 2 * 3.14 * radio  
"IN" + "PUT"
```



Lenguaje RAPID

- **Expresiones del lenguaje**

- ▶ **Lógicas:** Devuelven un valor de tipo bool

Utilizan los operadores lógicos: <, >, <>, =, <=, >=, AND, OR, NOT, XOR

Ejemplo:

```
DInput (di1) = 1 Doutput (do3) = 0  
num1 < num2;  
nombre1 = nombre2;  
Doutput (do1) = 0 AND pos1.x > 100
```



Lenguaje RAPID

- **Instrucciones: Movimiento**

Para mover el robot hay tres instrucciones:

```
MoveJ Punto, Velocidad, Zona, Herramienta
```

Se mueve el robot hacia un punto usando coordenadas articulares.

Cuando no tiene que seguir ninguna trayectoria determinada.

```
MoveL Punto, Velocidad, Zona, Herramienta
```

Se mueve el robot hacia un punto usando la línea recta.

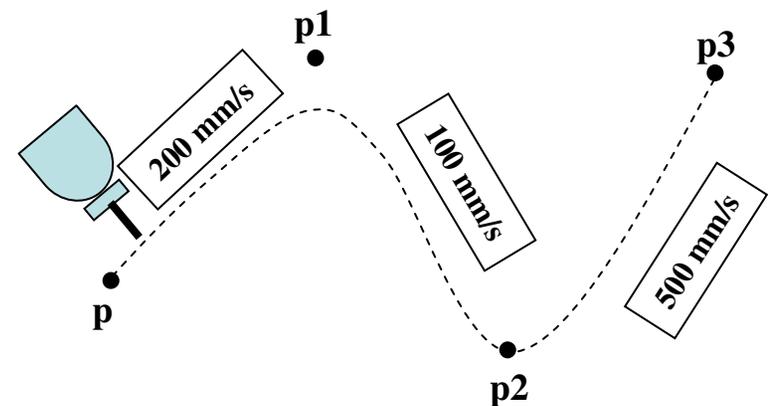
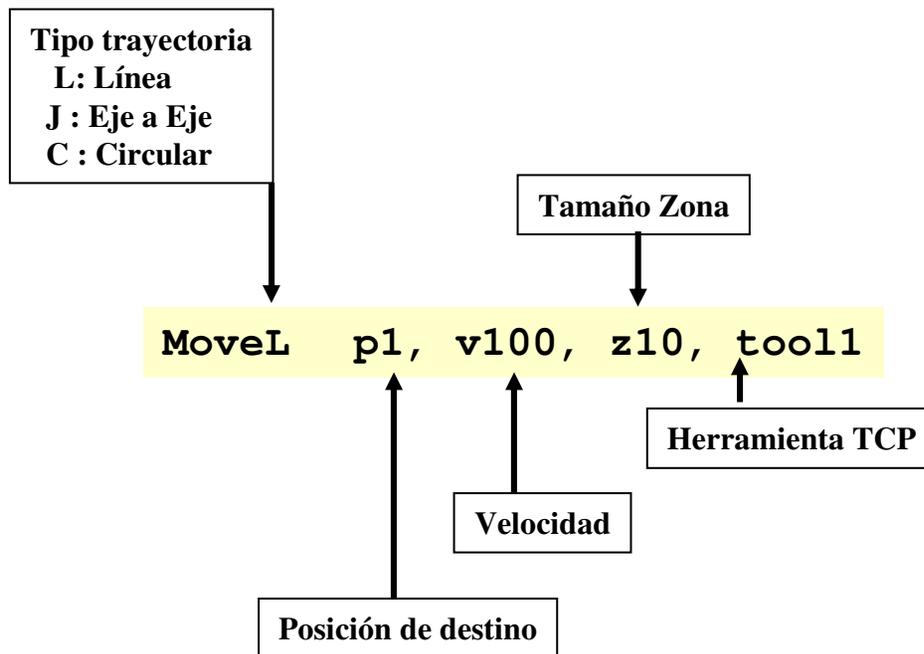
```
MoveC Punto_Circulo, Punto_Destino, Velocidad, Zona, Herramienta;
```

Se mueve el extremo del robot hacia el punto de destino pasando por el punto del círculo trazando un arco de circunferencia.



Lenguaje RAPID

● Instrucciones: Movimiento

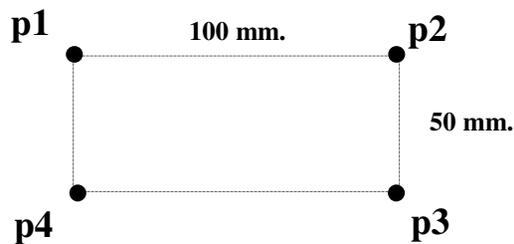
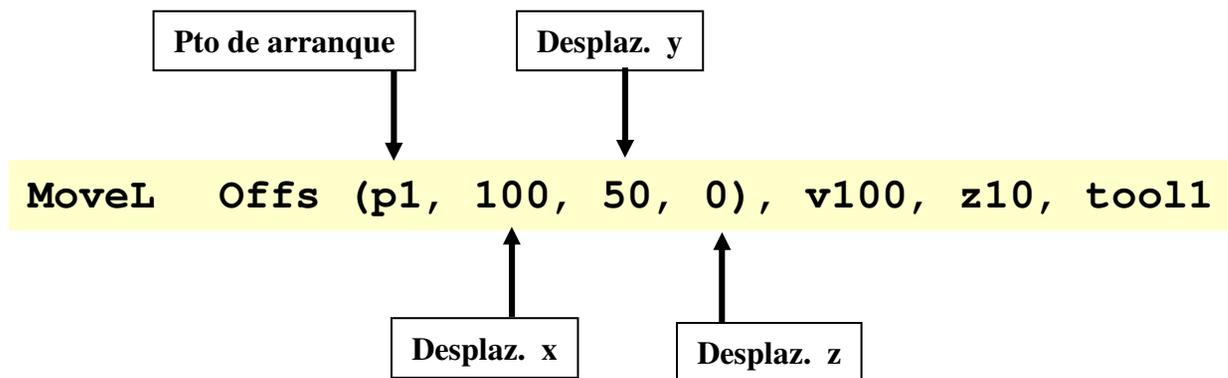


```
MoveL p1, v200, z10, herram1
MoveL p2, v100, fine, herram1
MoveJ p3, v500, fine, herram1
```



Lenguaje RAPID

● Instrucciones: Posicionamiento (Programación con desplazamiento)



```
MoveL p1, v200, fine, herram1
MoveL Offs (p1, 100, 0, 0), v100, fine, herram1
MoveL Offs (p1, 100, 50, 0), v100, fine, herram1
MoveL Offs (p1, 0, 50, 0), v100, fine, herram1
MoveL p1, v100, fine, herram1
```

La función `Offs` sirve para desplazar una posición del robot. Devuelve el dato de la posición desplazada de tipo `robtarg`.

`Offs(punto, OffestX, OffestY, OffestZ)`



Lenguaje RAPID

- **Instrucciones: entrada/salida**

```
Set señal;
```

Sirve para colocar el valor de la señal de la salida digital a uno.

```
Reset señal;
```

Sirve para poner una señal de salida digital a cero

```
SetDO señal, valor;
```

Sirve para cambiar el valor de una señal de salida digital

```
DInput ( di1) / DOutput (do2)
```

Lectura de Entradas / Salidas digitales

Ejemplo: **SetDO** do1, 1 ! Activación =1 Desactivación = 0



Lenguaje RAPID

- **Instrucciones: Condición de espera**

WaitDI	di, 1	!Esperar hasta que se active una señal digital
WaitTime	0.5	!Esperar cierto tiempo
WhileUntil		!Esperar hasta que se cumpla cierta condición



Lenguaje RAPID

- **Control de Flujo: Compact IF**

Ejecutar una instrucción sólo si se cumple una condición.

```
IF <condición> Instrucción;
```

- **Control de Flujo: IF**

Diferentes instrucciones se ejecutan si se cumple la condición.

```
IF <condición>           THEN  
    Instrucciones;  
ELSE  
    Instrucciones;  
ENDIF
```

- **Control de Flujo: FOR**

```
FOR <contador> FROM VI TO VF [ STEP Incremento ] DO  
    Instrucciones;  
ENDFOR
```



Lenguaje RAPID

- **Control de Flujo: WHILE**

```
WHILE <condición>      DO  
    Instrucciones;  
ENDWHILE
```

- **Control de Flujo: TEST**

```
TEST <dato>  
    CASE valor1, valor2,..., valor(n-1):  
        rutinal;  
    CASE valor n:  
        rutinax;  
    DEFAULT  
        instrucciones;  
ENDTEST
```

- **Control de Flujo: GOTO**

```
GOTO Etiqueta
```



Lenguaje RAPID

● Juego de instrucciones del RAPID

:=	Asignar un valor
Abs()	Obtener el valor absoluto
AInput()	Leer el valor de una señal de entrada analógica
AccSet	Reducir la aceleración
Add	Sumar un valor numérico
Clear	Borrar un valor
ClkStart	Iniciar un reloj para la toma de tiempos
ClkStop	Parar un reloj para la toma de tiempos
comment	Comentario
CompactIF	Si se cumple una condición, entonces... (una instrucción)
ConfJ	Controlar la configuración durante movimiento articular
ConfL	Monitoriza la configuración del robot durante movimiento en línea recta
Decr	Decrementar en 1
EXIT	Terminar la ejecución del programa
FOR	Repetir un número de veces
GetTime()	Leer el valor de la hora actual como valor numérico
GOTO	Ir a una nueva instrucción
GripLoad	Definir la carga del robot
HoldMove	Interrumpir el movimiento del robot
IF	Si se cumple una condición, entonces...; de otra manera...
Incr	Incrementar en 1
InvertDO	Invertir el valor de una salida digital
label	Nombre de una línea
LimConfL	Definir la desviación permitida en la configuración del robot
MoveC	Mover el robot en movimiento circular
MoveJ	Movimiento articular del robot
MoveL	Movimiento del robot en línea recta
Offs()	Desplazamiento de la posición del robot
Open	Apertura de un fichero o de un canal serie



Lenguaje RAPID

● Juego de instrucciones del RAPID

Present()	Comprobar que se utiliza un parámetro opcional
ProcCall	Llamada a un nuevo procedimiento
PulseDO	Generar un pulso en una señal digital de salida
RAISE	Llamada a un manejador de errores
RelMove	Continuar con el movimiento del robot
Reset	Reset de una salida digital
RETRY	Recomenzar tras un error
RETURN	Terminar la ejecución de una rutina
Set	Set de una salida digital
SetAO	Cambiar el valor de una salida analógica
SetDO	Cambiar el valor de una salida digital
SetGO	Cambiar el valor de un grupo de salidas digitales
SingArea	Definición de la interpolación alrededor de puntos singulares
Stop	Parar la ejecución de un programa
TEST	Dependiendo del valor de la expresión...
TPERase	Borrar el texto de la paleta de programación
TPReadFK()	Leer las teclas de función de la paleta de programación
TPWrite	Escribir en la paleta de programación
VelSet	Cambiar la velocidad programada
WaitDI	Esperar hasta el set de una entrada digital
WaitTime	Esperar un tiempo determinado
WaitUntil	Esperar hasta que se cumpla una condición
WHILE	Repetir mientras ...
Write	Escribir en un fichero de caracteres o en un canal serie
WriteBin	Escribir en un canal serie binario



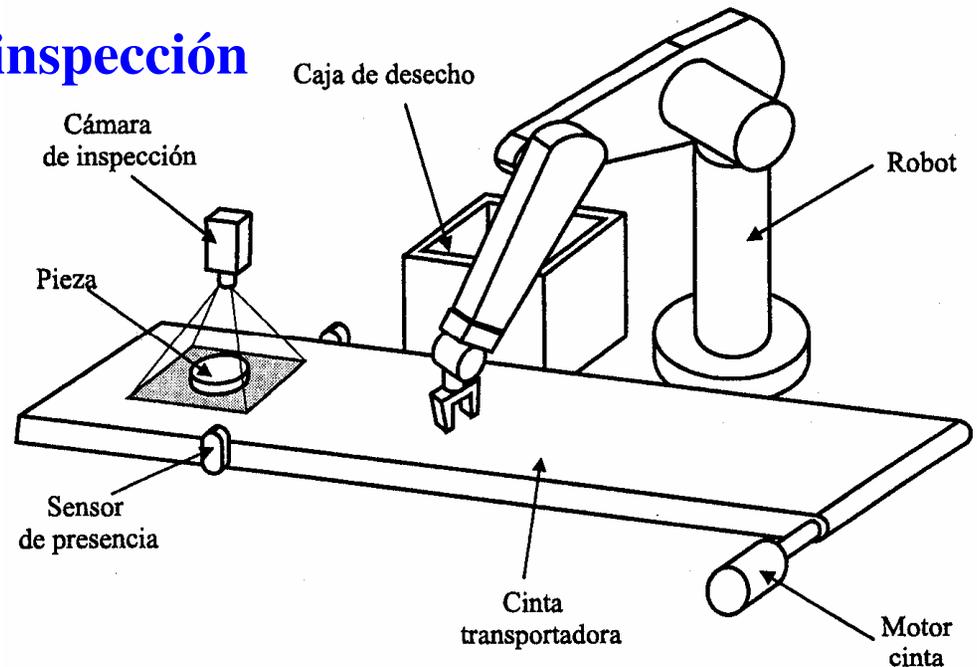
Lenguaje RAPID

● Ejemplo: Célula robotizada de inspección

▶ El robot retira de una cinta transportadora aquellas piezas identificadas como defectuosas.

☞ La operación se desarrolla como sigue:

1. El robot se encuentra en **espera** hasta la llegada de una **señal** indicando la existencia de una pieza defectuosa sobre la cinta transportadora.
2. El robot procede entonces a **parar la cinta** y a **coger la pieza** y a **depositarla** en un almacén de piezas defectuosas.
3. El propio robot se encarga de **activar** de nuevo el movimiento de la cinta una vez la pieza ha sido cogida.
4. Tras la operación, el robot **vuelve a su posición inicial** y se repite de nuevo el ciclo.





Lenguaje RAPID

● Célula de inspección: Definición de variables

herramienta: una variable de tipo *tooldata* que representa una pinza en el extremo del robot para la manipulación de piezas.

carga: una variable de tipo *loaddata* para definir la carga a transportar por la pinza.

```
PERS tooldata herramienta:= [FALSE, [[97, 0, 223],  
                                     [0.924, 0, 0, 0.383, 0]], [5, [-23, 0, 75], [1, 0, 0, 0], 0, 0, 0]]  
PERS loaddata carga:= [5, [50, 0, 50], [1, 0, 0, 0], 0, 0, 0];
```



tooldata

robhold : Tipo bool que define si el robot sujeta la herramienta o no.
tframe : Sistema de coordenadas de la herramienta
Posición del TCP (muñeca del robot) (x,y,z)
Orientación (q1,q2,q3,q4)
tload : Carga de la herramienta
Peso
Centro de gravedad (x,y,z)
Ejes de momento de la herramienta (q1,q2,q3,q4)
Momento de inercia de los ejes (x,y,z).



Lenguaje RAPID

● Célula de inspección: Definición de variables

```
VAR signaldo pinza           !señal de activación de pinza
VAR signaldo activar_cinta   !señal de activación de cinta
VAR signaldi pieza_defectuosa !señal de pieza defectuosa
VAR signaldi terminar        !señal de terminar programa
```

Es necesario definir una configuración inicial en la que el robot espera la señal que le indica que puede recoger la pieza defectuosa.

```
VAR robtargt conf_espera := [[600, 500, 225], [1, 0, 0, 0], [1, 0, 0, 0],
                               [9E9, 9E9, 9E9, 9E9, 9E9, 9E9]];
```



robtargt

trans : Posiciones (x, y, z)
rot : Orientación de la herramienta.
robconf : Configuración de los ejes.
extax : posición de los ejes externos



Lenguaje RAPID

- **Célula de inspección: Rutinas de control de la pinza**

```
PROC Cogrer ()  
  Set pinza      !Cerrar la pinza activando la señal digital pinza  
  WaitTime 0.3   !Esperar 0,3 segundos  
  GripLoad carga !Señalar que la pieza está cogida  
ENDPROC
```

```
PROC Dejar ()  
  Reset pinza    !Abrir la pinza  
  WaitTime 0.3   !Esperar 0,3 segundos  
  GripLoad LOAD0 !Señalar que no hay pieza cogida  
ENDPROC
```



Lenguaje RAPID

- **Célula de inspección: Rutina de coger la pieza de la cinta**

```
PROC Coger_pieza ()
  MOVEJ *,VMAX,z60,herramienta !Mov. en articulares con poca precisión
  MOVEL *,V500,z20,herramienta !Mov. Línea recta con precisión
  MOVEL *,V150,FINE,herramienta !Bajar con precisión máxima
  Coger !Coger la pieza
  MOVEL *,V200,z20,herramienta !Subir con la pieza cogida
ENDPROC
```

- **Célula de inspección: Rutina de dejar la pieza**

```
PROC Dejar_pieza ()
  MOVEJ *,VMAX,z30,herramienta !Mover hacia almacén piezas dañadas
  MOVEJ *,V300,z30,herramienta
  Dejar !Dejar la pieza
ENDPROC
```



Lenguaje RAPID

- **Célula de inspección: Rutina de ir a la posición de espera**

```
PROC Ir_posicion_espera ()  
  MOVEJ conf_espera, VMAX, z30, herramienta  !Mover a posición inicial  
ENDPROC
```



Lenguaje RAPID

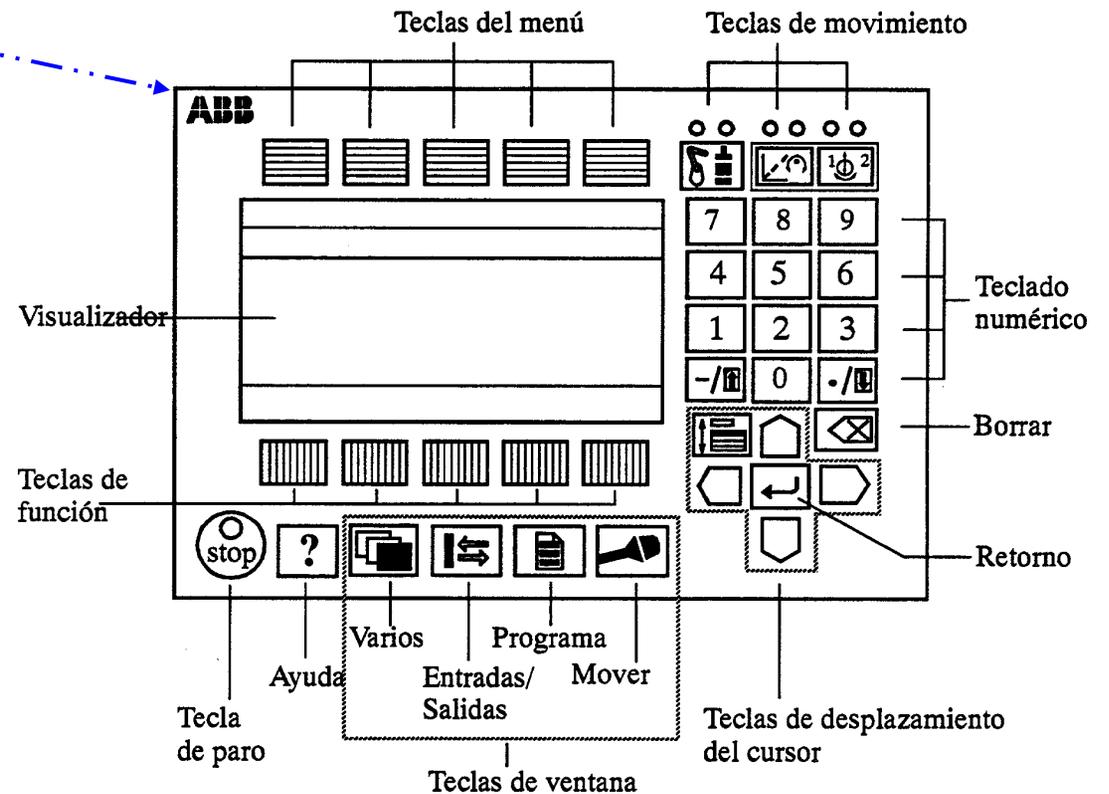
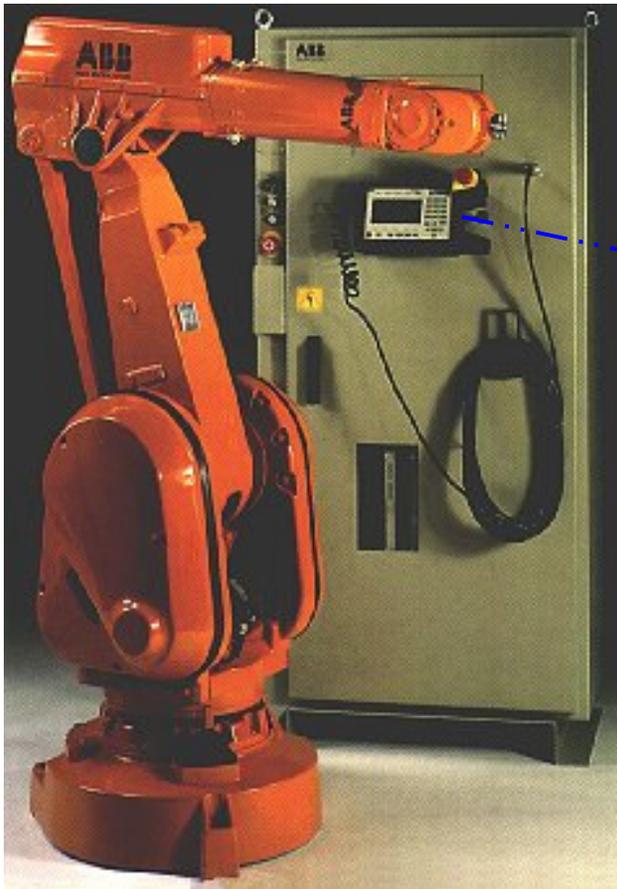
● **Célula de inspección: Programa principal**

```
PROC main()  
  Ir_posicion_espera;           !Mover a posición de espera  
  WHILE Dinput(terminar)=0 Do  !Esperar la señal de terminar  
    IF Dinput(pieza_defectuosa)=1 THEN !Esperar la señal de pieza  
                                     !defectuosa  
    SetDO activar_cinta,0;        !Parar cinta  
    Coger_pieza                   !Coger la pieza  
    SetDO activar_cinta,1;        !Activar señal de cinta  
    Ir_posicion_espera;          !Mover a posición de espera  
  ENDIF  
ENDWHILE  
ENDPROC
```



Lenguaje RAPID

- Entorno de programación





Lenguaje RAPID

- Entorno de programación: **ABB**

RAPID SyntaxChecker
(Analizador sintáctico fuera de línea)

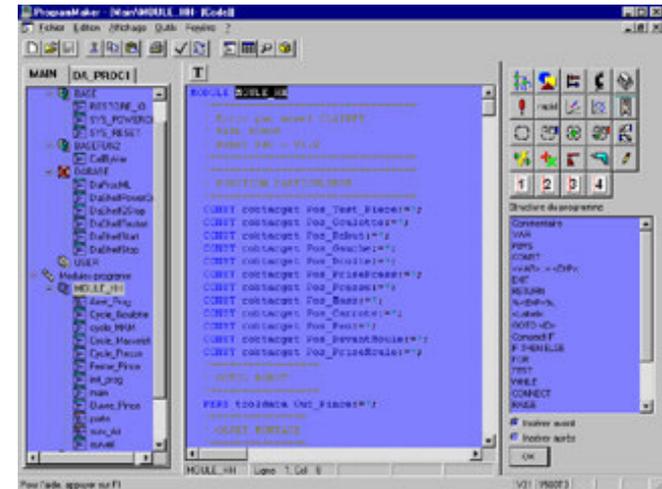
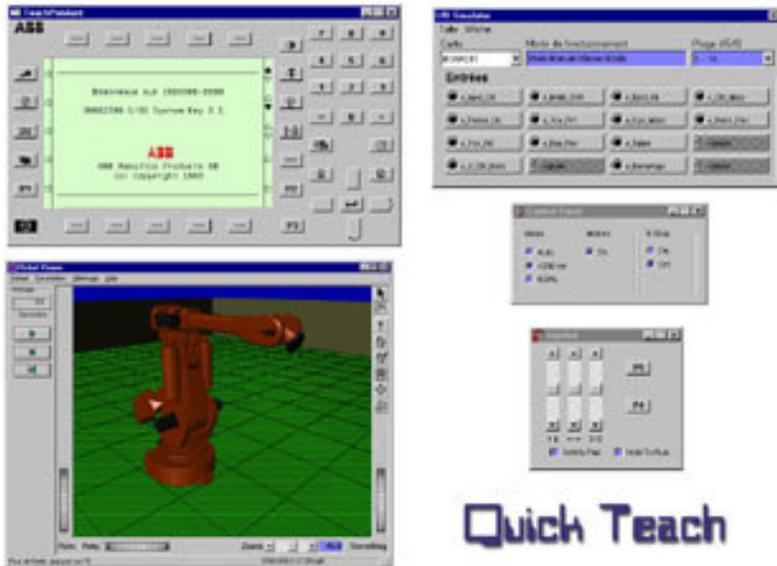
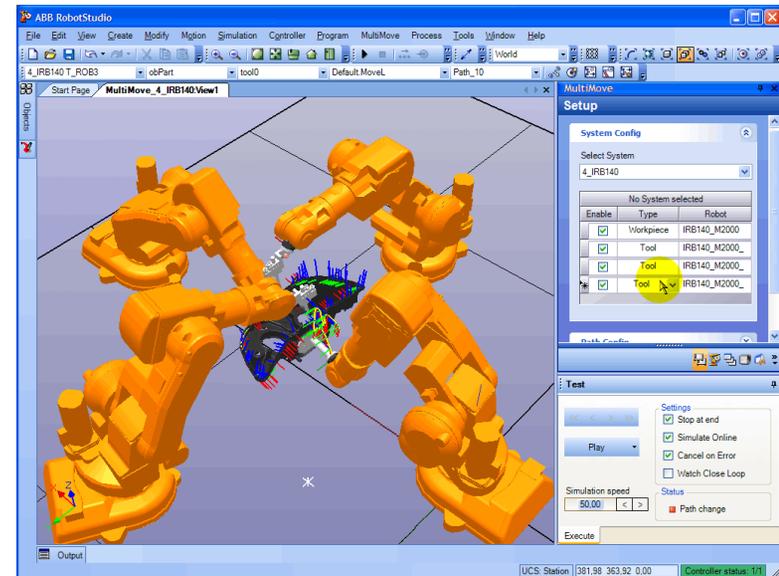


ABB Deskware



Quick Teach

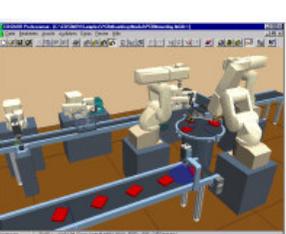
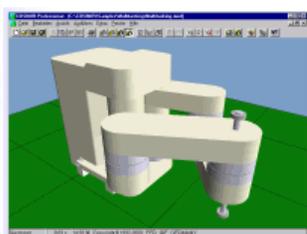
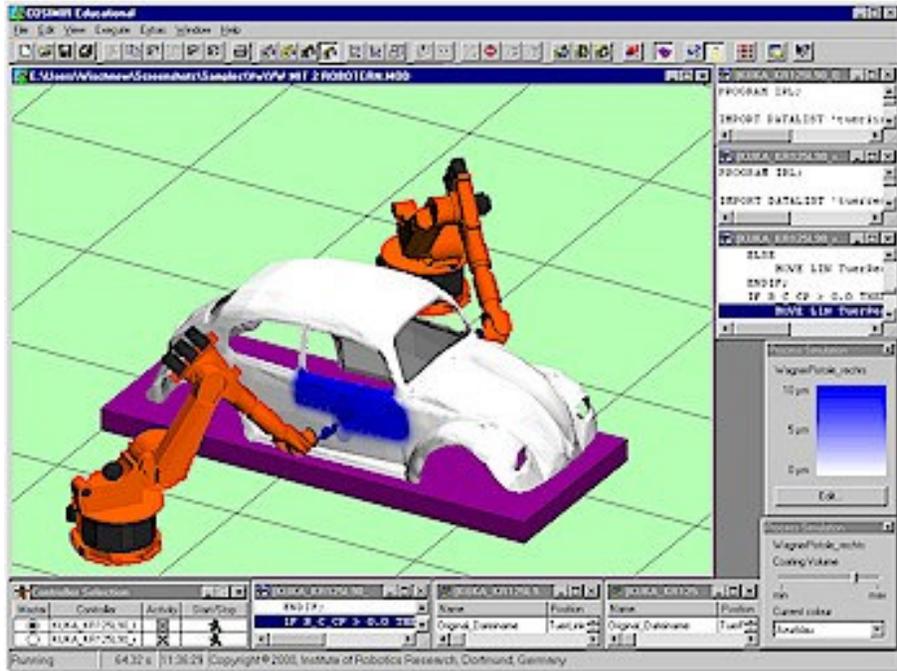


RobotStudio™ 5



Lenguaje RAPID

- Entorno de programación: **FESTO COSIMIR®**





Lenguaje RAPID



● **FESTO COSIMIR ®**

Demo >>

Name	Position	Orientation	Rel. to Object
HomePosition	855.0,0.0,1370.0	-180.0,180,R,A,N	
GreifeDeckel	915.0,-50.0,906.3	-91.0,-180,R,A,N	
OberhalbDeckel	915.0,-50.0,1016.3	-91.0,-180,R,A,N	
Zwischenposition	312.9,599.7,814.7	-91.0,-180,R,A,N	
LoeseDeckel	-2.4,1193.2,15.6	-91.0,-180,R,A,N	
Zwei2	192.1,756.3,602.6	-91.0,177,R,A,N	

```

MoveJ HomePosition, v1000, z10, tool10;

! initialize the robot
InitRobot;

! Hole Greifer
ret := ChangeGripper (GRIPPER);

! Deckel holen
MoveL OberhalbDeckel, v1000, z10, actualtool;
ret := OpenGripper();
MoveL GreifeDeckel, v100, fine, actualtool;
ret := CloseGripper();
WaitTime 2;
MoveL OberhalbDeckel, v100, z10, actualtool;

WaitTime 1;
MoveL behindgripes, v200, fine, gripperflange;
actualcpnum:=GRIPPER;
actualtool:=gripper1;

! illegal gripper number
DEFAULT:
MoveJ savechangeupos, v200, fine, gripperflange;
RETURN FALSE;

ENDTEST

! move to safe position
MoveJ savechangeupos, v1000, fine, gripperflange;

! action finished
RETURN TRUE;
  
```

Stopped | 126.81 s | 18:32:27 | Copyright © 1992-2000 · EFR · IRF · GERMANY



Automatización Industrial - II

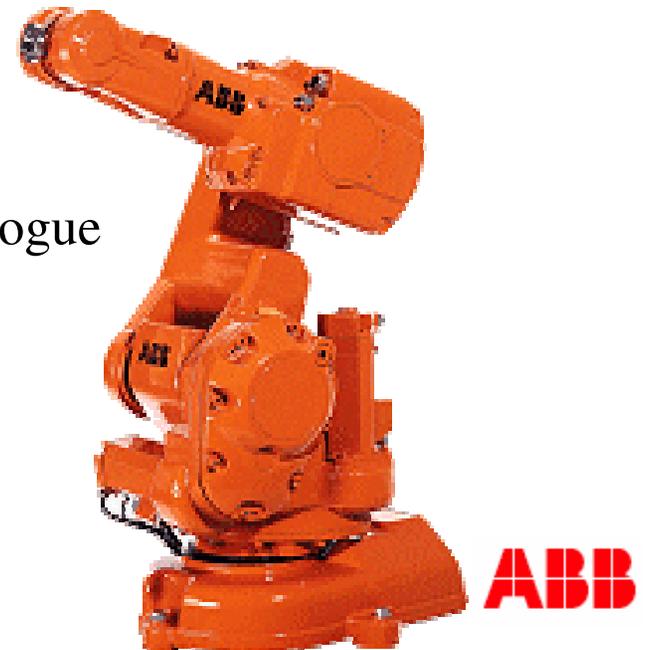
56 – 10569 2º Cuatrimestre 2006

Práctica 3– Lunes y Miércoles 8 & 10 de Mayo 2006



Lenguaje RAPID

Robotics Application Programming Interactive Dialogue





Práctica 3 – Plazo de entrega Miércoles 31 de mayo 2006

El lenguaje RAPID

- ❶ Describir cual son los parámetros de los siguientes tipos de datos en RAPID:

pos: Representar posiciones

tooldata: Describe las características de una herramienta.

confdata: Permite definir las configuraciones de los ejes del robot

loaddata: Sirve para describir la carga colocada en la muñeca del robot.

robtarget: Sirve para definir la posición del robot y de sus ejes externos

- ❷ Describir cual son los parámetros de las siguientes instrucciones en RAPID:

MoveJ, movimiento en coordenadas articulares

MoveL, movimiento en línea recta

MoveC, movimiento circular

WaitTime, espera por tiempo

WaitUntil, espera por condición

WaitDI, espera por entrada digital

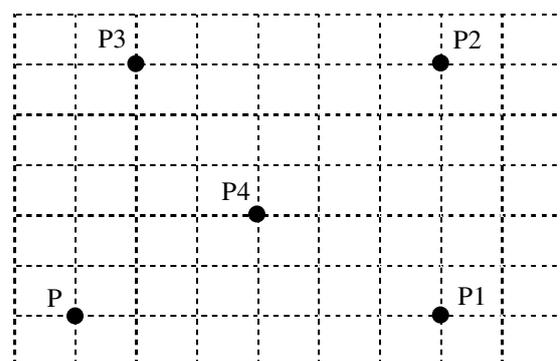
- ❸ Siendo la escala de la retícula 5 cm. y estando el extremo del robot en el punto p. Dibuja la trayectoria que sigue el extremo del robot luego de ejecutar las siguientes instrucciones.

```
MoveL p1, v200, z5, herram1
```

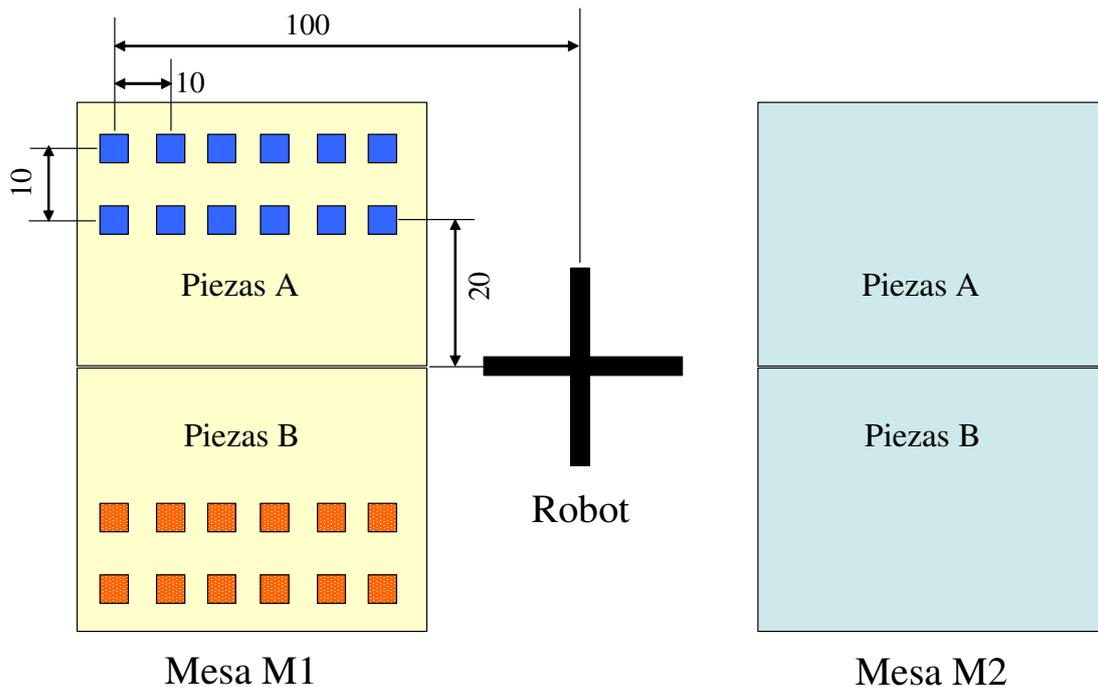
```
MoveL p2, v100, z10, herram1
```

```
MoveL p3, v500, z15, herram1
```

```
MoveL p4, v50, fine, herram1
```



④ Se pretende programar un robot que realiza la operación de paletización, transportando las piezas de la mesa M1 a la mesa M2 (ver figura). Las masas están situadas de forma simétrica respecto al robot. Cada una de las mesas esta dividida en dos partes, una para las piezas de tipo A y otra para las piezas de tipo B. hay en total 12 piezas de cada tipo. La diferencia entre las piezas A y B es que deben ser manipuladas por diferentes tipos de pinzas. Las de tipo A con una pinza cuya longitud, en la dirección Z de la pinza, es de 10cm. y las de tipo B es de 20 cm. en la misma dirección. Para agarrar la pieza, la pinza debe ponerse a 10 cm. sobre ella, es decir en la dirección $-Z$ del sistema del mundo.



Las instrucciones de programación que se pueden utilizar son:

- **MoveJ** ($x,y,q1,q2,q3,q4,prec,vel$), instrucción de movimiento del robot en coordenadas del robot, donde (x,y,z) son las coordenadas, ($q1,q2,q3,q4$) son los cuaternios, ($prec$) es la precisión de posicionamiento, (vel) es la velocidad de movimiento.
- **TCP**(x,y,z), instrucción de cambio del TCP del robot, donde (x,y,z) define la geometría de la herramienta en el sistema de la herramienta.
- **Open**, abrir la pinza
- **Close**, cerrar la pinza
- **Wait** (t), esperar (t) segundos
- **X=X+inc**, instrucción de actualización de la variable X mediante la suma de un incremento (inc). Es aplicable a todas las variables
- **For i=1 to N**, instrucción que controla el bucle en un programa. Para terminar el bucle se usa la instrucción EndFor



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 8 – Jueves 11 de Mayo 2006

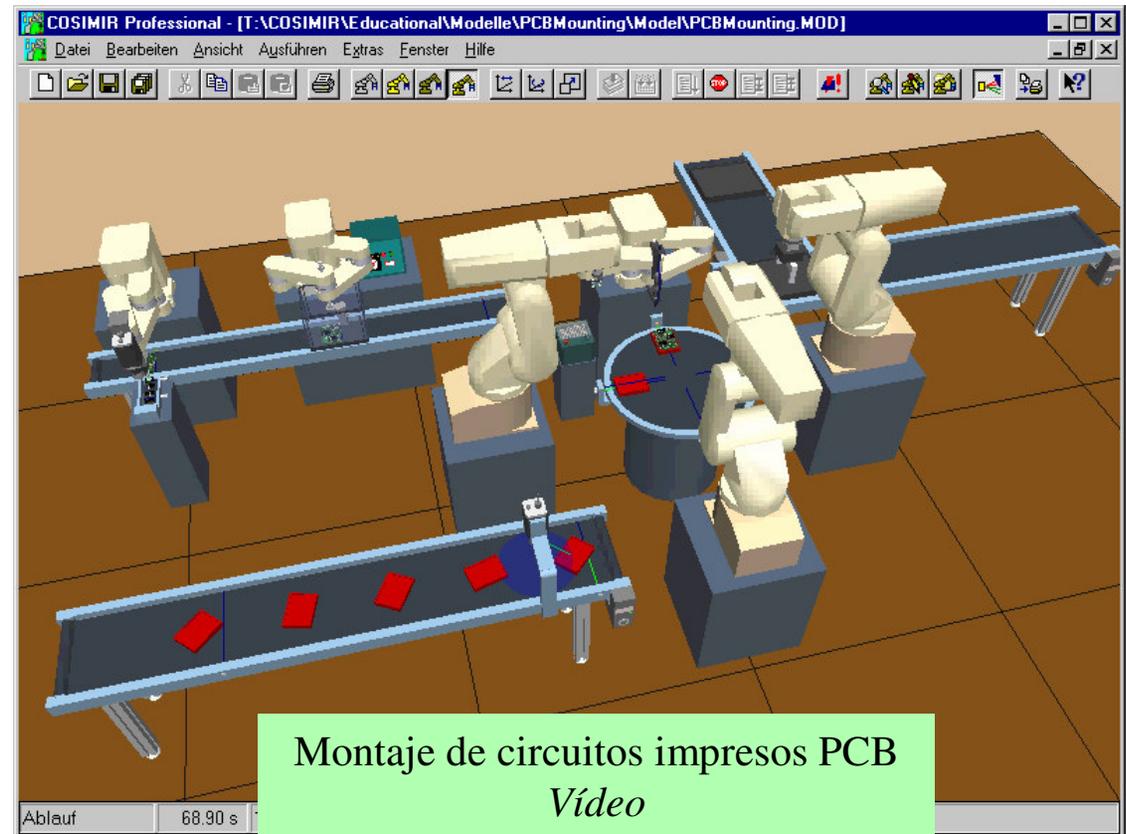
Criterios de Implantación de un Robot Industrial



Crterios de Implantación

● El robot industrial

- ▶ no trabaja de forma aislada
- ▶ debe interactuar con otros equipos, máquinas, procesos, etc.
- ▶ Formarán una estructura de fabricación superior denominada Célula de Fabricación Flexible (FMS)



- ▶ El robot es considerado la parte principal de las FMS o las células de trabajo robotizadas.



Crterios de Implantación

- **Aspectos a considerar en el diseño de una célula robotizada**

- ▶ **Lay-out:** esquema de implantación de equipos, máquinas y otros elementos de la planta.



- ▶ **Arquitectura de control:** tanto hardware como software.

- ▶ **Elección de la maquinaria :** en especial el ROBOT

- ▶ **Seguridad:** tratada de forma especial al aparecer maquinas con funcionamiento automático.

- ▶ **Justificación económica** de la implantación

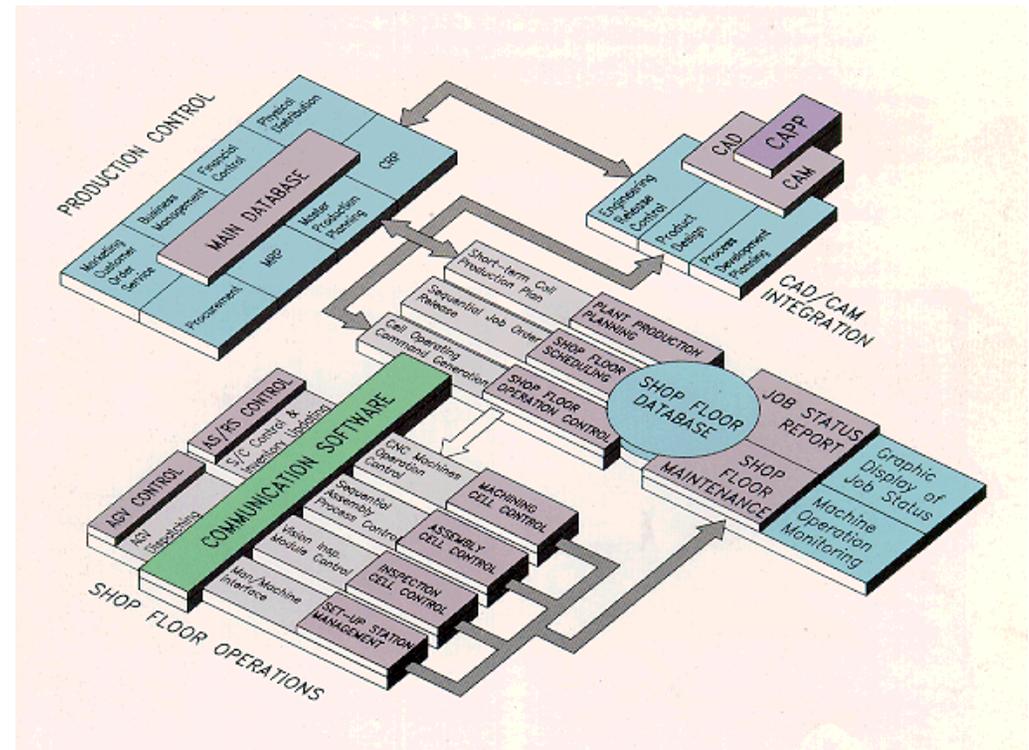


Criterios de Implantación

● Aspectos del proyecto de un sistema robotizado

- ▶ Definición detallada del lay-out.
- ▶ Posible rediseño del producto.
- ▶ Elección de robots necesarios.
- ▶ Definición o diseño de elementos periféricos.

- **Pasivos:** Mesas, alimentadores, utillajes, etc.
- **Activos:** Transportes, manipuladores secuenciales, CN, etc.





Crterios de Implantación

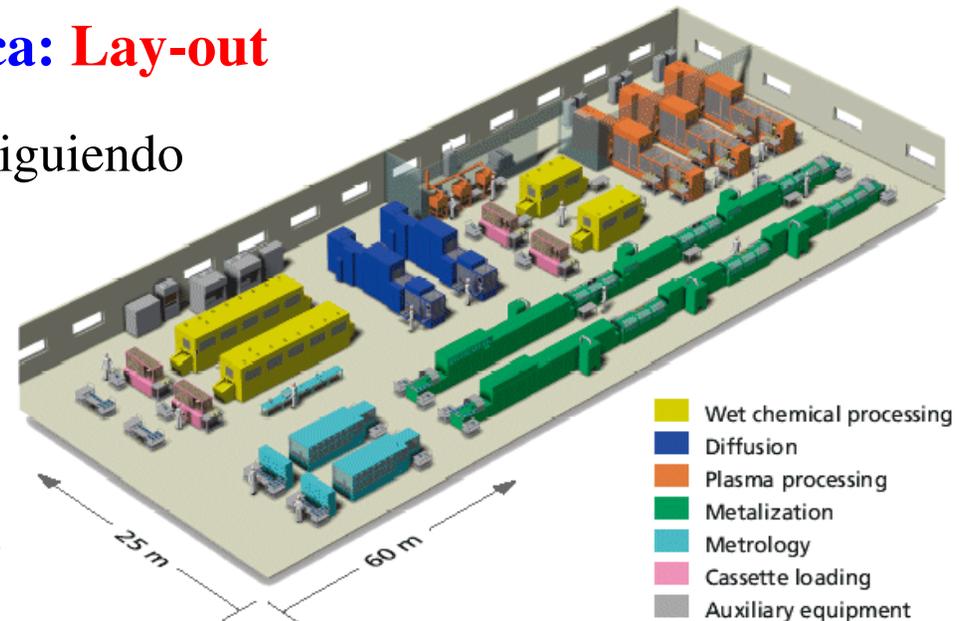
- **Definición de la disposición física: Lay-out**

- ▶ En un proceso de **ajuste iterativo** siguiendo las fases indicadas anteriormente.

- ▶ Utilizando herramientas CAD genéricas.

- ▶ Simuladores de células robotizadas.

- ▶ Simuladores de sistemas de fabricación flexible





Crterios de Implantación

● **Simulación**

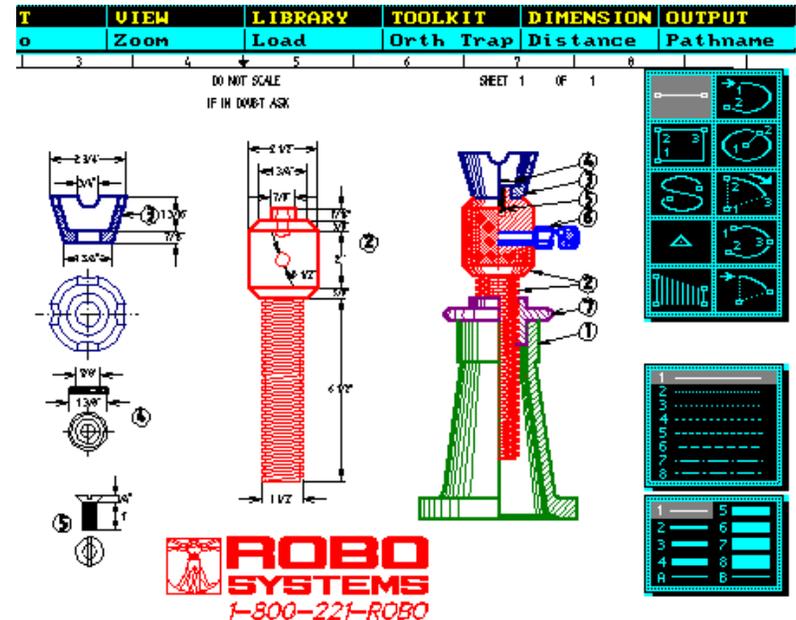
- ▶ Evaluar las diferentes alternativas en cuanto al robot a utilizar y la disposición física de todo el sistema.
- ▶ Dimensionar adecuadamente la célula, informando sobre su productividad, rendimiento y comportamiento ante cambios en la demanda.
- ▶ Ensayar estrategias de control de la célula.
- ▶ ROBOCAD, GRASP, CATIAS, TOROS, PreView



Crterios de Implantación

● Simulación: ROBCAD

- ▶ Es un sistema de ayuda para el diseño, simulación y programación off-line de sistemas automáticos de manufactura.
- ▶ Construir y modificar elementos de la célula de trabajo.
- ▶ Definir las tareas de los mecanismos utilizados.
- ▶ Trasladar el programa al lenguaje del robot.
- ▶ Permite al ingeniero: Verificación, Reducción de los tiempos de diseño, Cambios del diseño, simulaciones a diferentes velocidades, etc.





Criterios de Implantación

● Simulación: GRASP

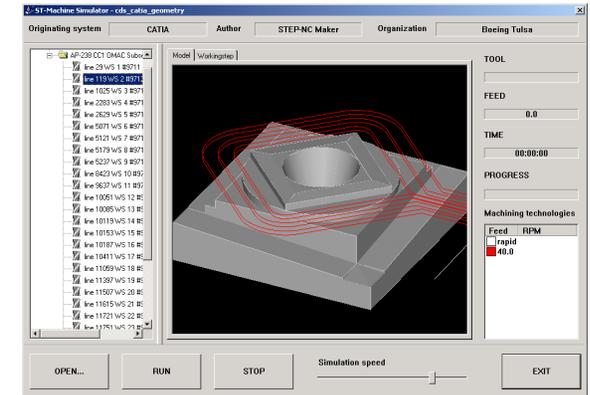
- ▶ Sistema 3D para simulación gráfica que contiene un modelador de sólidos y un modelador cinemático.
- ▶ Genera información de posición y lógica requerida por el robot.
- ▶ Rápida evaluación de diferentes robots.
- ▶ Identificación y eliminación de errores de planificación.
- ▶ Generación de programas off-line de robots.
- ▶ Reducción de riesgo de errores, costos y daños.
- ▶ Visualización interactiva del sistema completo antes de la instalación.
- ▶ Diferentes funciones matemáticas.



Crterios de Implantación

● Simulación: CATIA

- ▶ Funciones similares a los simuladores presentados anteriormente.
- ▶ CATIA, posee una función denominada ROBOT que permite la construcción de nuevos robots con relativa facilidad.
- ▶ Función TASK, permite la programación de los robots por medio de funciones como agarrar y soltar un objeto. Se trata de una función que permite simulación y optimización del trabajo de un robot.



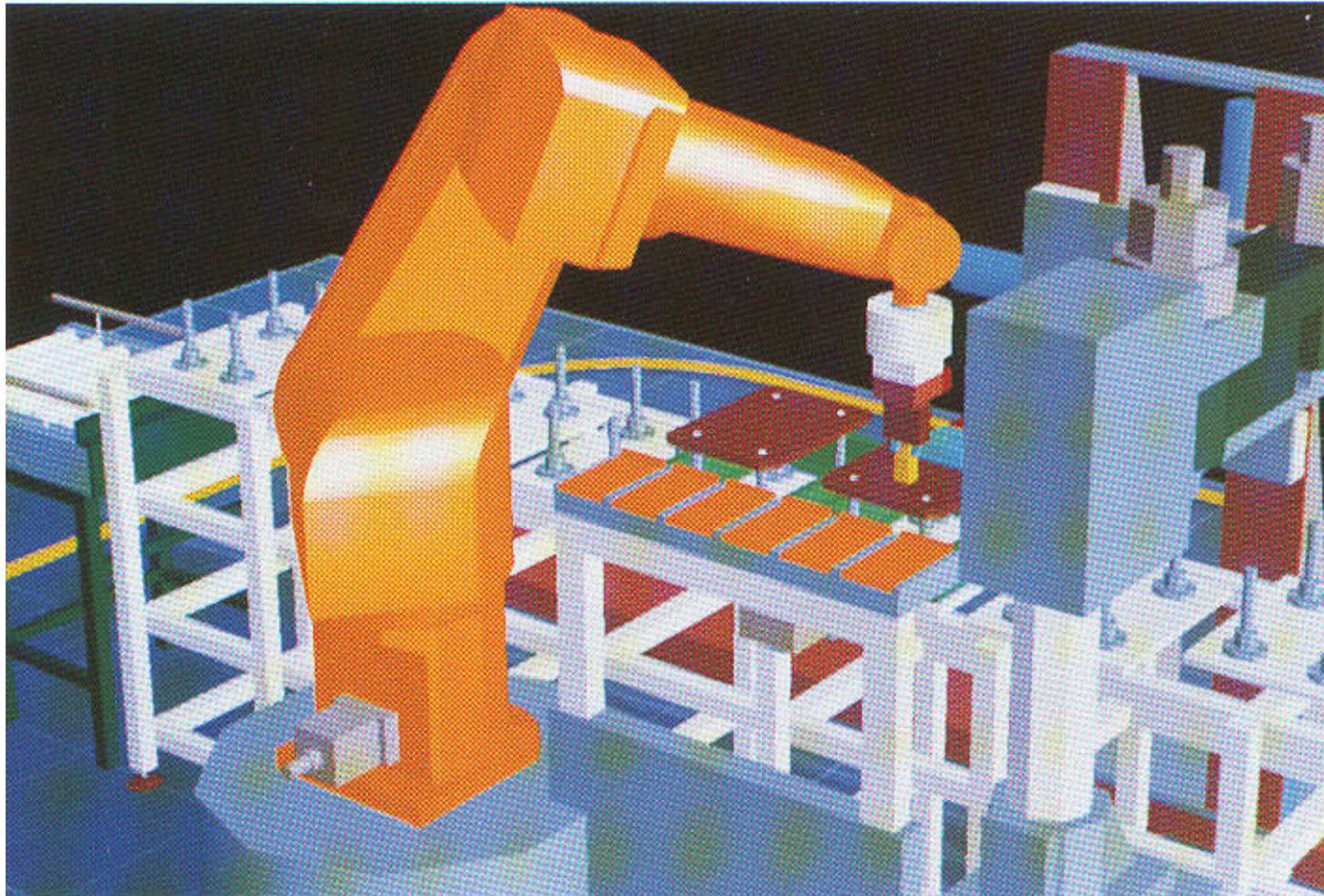
● Simulación: TOROS

- ▶ Desarrollado en la Universidad Politécnica de Madrid.
- ▶ Es un simulador de sistemas Multirobot.



Crterios de Implantación

- **Simulación: Staubli-Unimation**



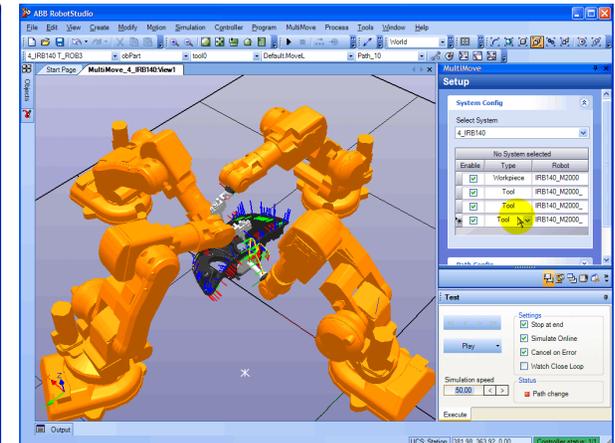
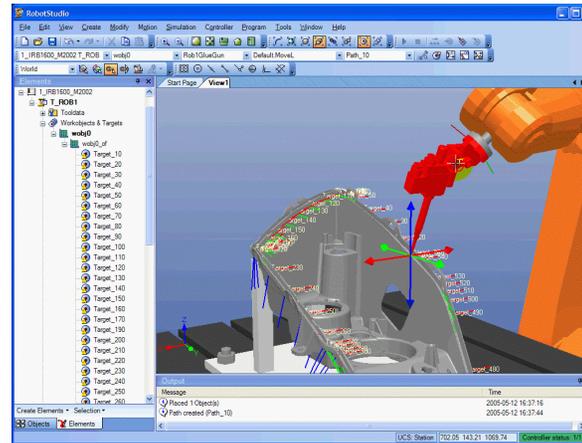
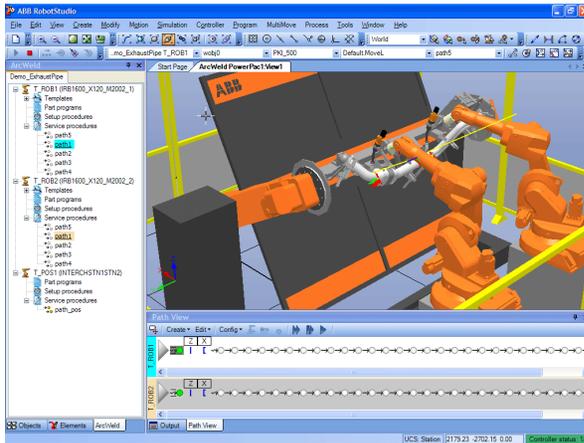


Automatización Industrial -II

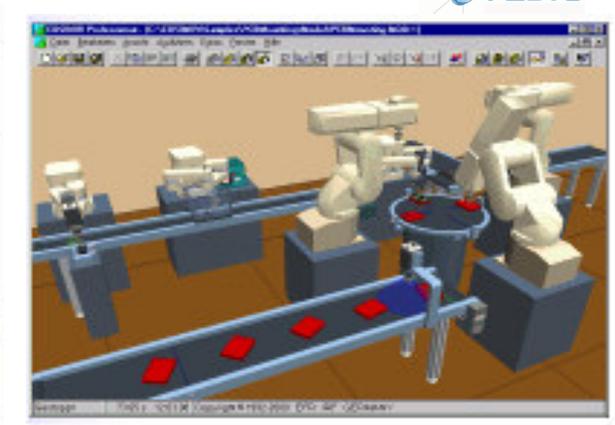
Criterios de Implantación

- **Simulación: ABB RobotStudio**

ABB RobotStudio™ 5



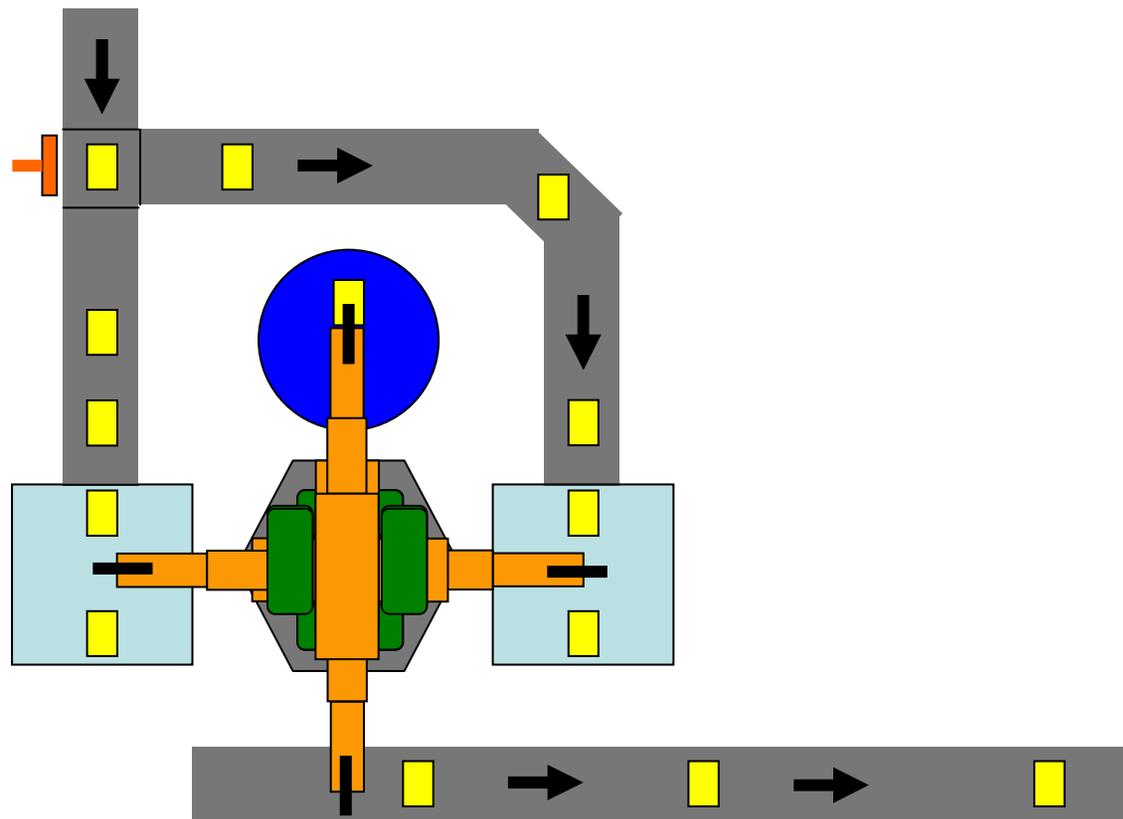
- **Simulación: FESTO COSIMIR**





Criterios de Implantación

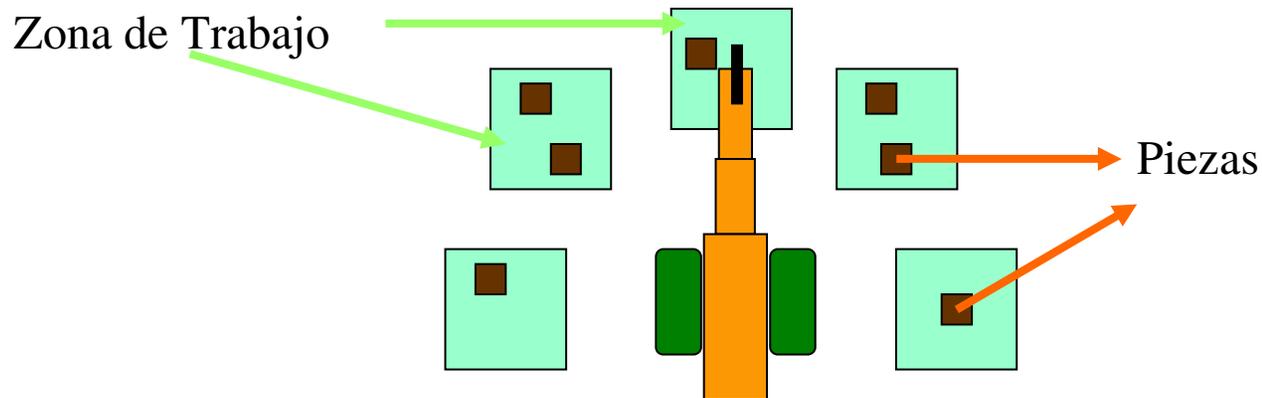
- Disposición del robot en la célula





Criterios de Implantación

- **Disposición del robot en la célula: Robot en el centro de la célula**

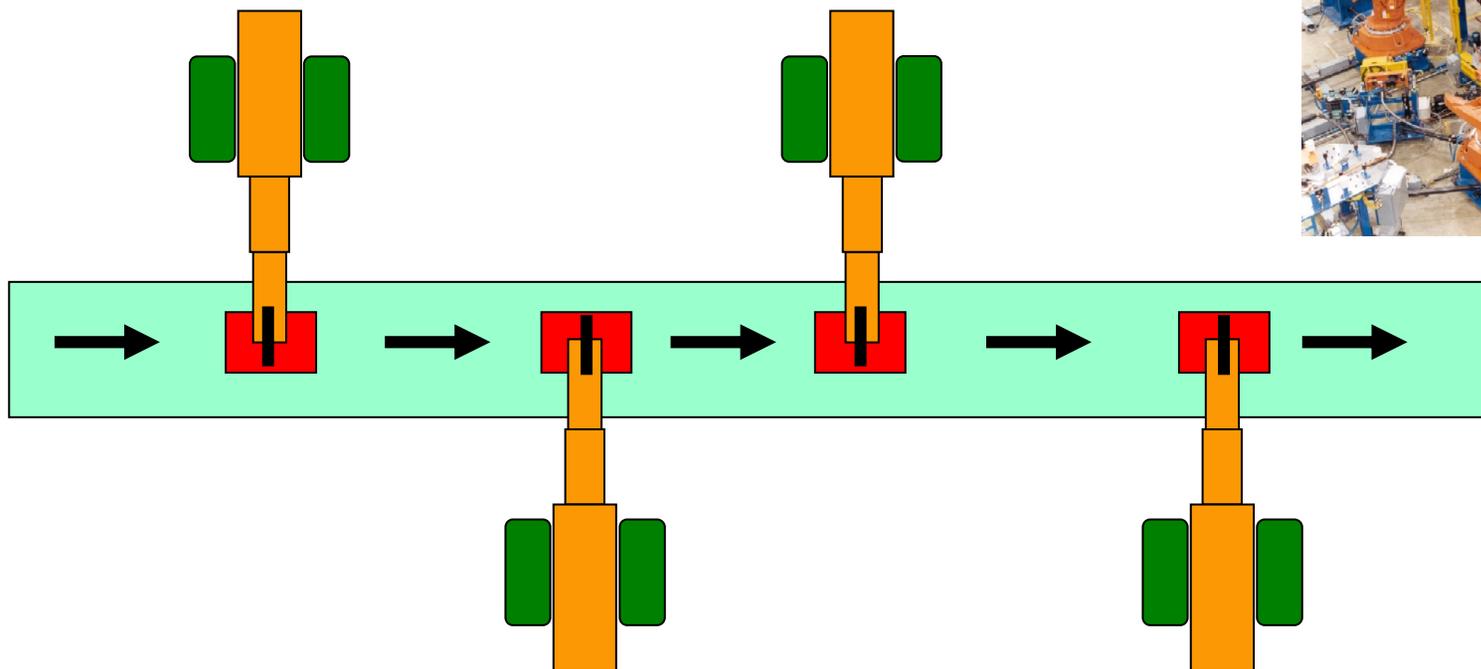


- ▶ El robot se sitúa de modo que quede rodeado por el resto de los elementos que intervienen en la célula.
- ▶ Se trata de una disposición típica para robots de estructura articular polar, cilíndrica o SCARA, en la que se puede aprovechar al máximo su campo de acción, que presenta una forma básica de esfera.
- ▶ Utilizada en aplicaciones donde el robot sirve a una o varias máquinas.
- ▶ Aplicaciones de soldadura al arco, paletización o ensamblado, en las que el robot debe alcanzar diversos puntos de su área de trabajo.



Criterios de Implantación

- **Disposición del robot en la célula: Robot en línea**

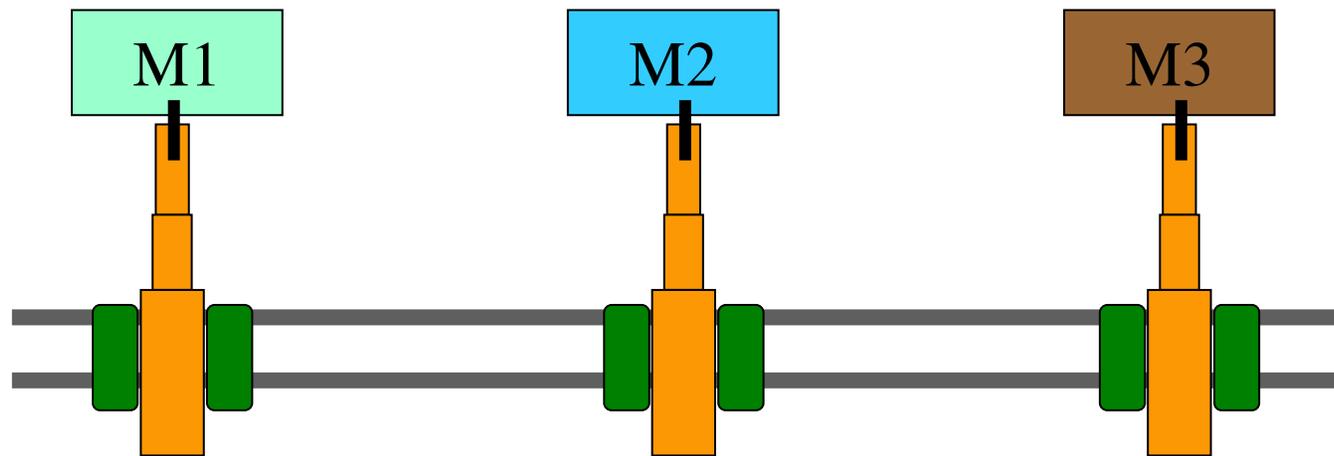


- ▶ Cuando se trabaja sobre elementos que llegan en un sistema de transporte.
- ▶ Líneas de soldadura
- ▶ El transporte podrá ser intermitente o continuo.



Criterios de Implantación

- **Disposición del robot en la célula: Robot móvil**

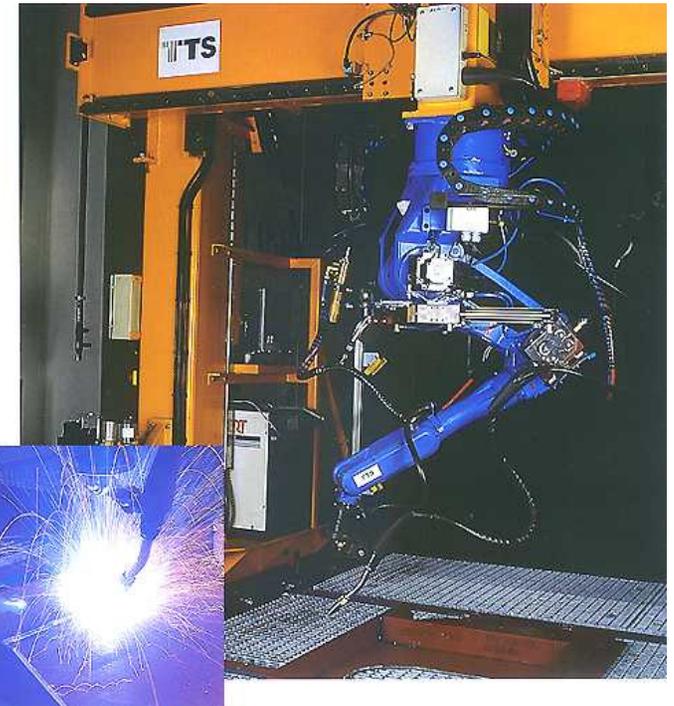


- ▶ Se le añade un grado de libertad adicional al robot ubicándolo sobre una vía que permita su desplazamiento de manera controlada.
- ▶ Permite cubrir un mayor campo de acción (pintura en carrocerías).
- ▶ Seguir el movimiento de la pieza en el caso que ésta se desplace de forma continua.
- ▶ Dar servicio a varias máquinas (carga-descarga).



Criterios de Implantación

- **Disposición del robot en la célula: Robot suspendido**
 - ▶ Robot tipo pórtico en la que queda situado sobre el área de trabajo.
 - ▶ Invertido sobre la célula.
 - ▶ Mejor aprovechamiento del área de trabajo, permitiendo alcanzar puntos sobre su propio eje.
 - ▶ Aplicación de adhesivos, proyección de material (pintura), corte (láser, chorro) y soldadura arco.





Criterios de Implantación

- **Características del sistema de control de la célula**
 - ▶ Control individual de cada una de las máquinas, transportes y demás dispositivos, incluidos los robots que componen la célula.
 - ▶ Sincronización del funcionamiento de los diferentes dispositivos.
 - ▶ Detección, tratamiento y recuperación de situaciones anómalas.
 - ▶ Optimización del funcionamiento, distribuyendo si es posible las funciones de manera dinámica.
 - ▶ Interfaces:
 - Interfaz con el usuario.
 - Interfaz con otras células.
 - Interfaz con un sistema de control superior
 - ▶ Estructura jerarquizada en la que un elemento central (computador, PLC), mantenga la comunicación entre los diferentes controladores.



Criterios de Implantación

● **Selección del Robot**

▶ **Características geométricas.**

- Velocidad nominal máxima
- Aceleración y deceleración
- Área de trabajo (puntos singulares): uso de simuladores
- Grados de libertad
- Errores en el seguimiento de trayectorias
- Errores de posicionamiento

▶ **Características dinámicas**

- Fuerza
 - De agarre.
 - Carga máxima.
 - Control de fuerza par.
- Frecuencia de resonancia.



Crterios de Implantación

● **Selección del Robot**

▶ **Tipo movimientos**

- Movimientos punto a punto.
- Movimientos coordinados.
- Trayectorias continuas.

▶ **Modo de programación**

- Enseñanza (Guiado)
- Textual o ambos

▶ **Tipo accionamiento**

- Eléctrico (c.a. , c.c.)
- Neumático
- Hidráulico.

▶ **Comunicaciones**

- E/S Digitales/Analógicas
- Comunicaciones línea serie
- WEB

▶ **Servicio del proveedor**

Mantenimiento, asistencia técnica, cursos de formación, etc.

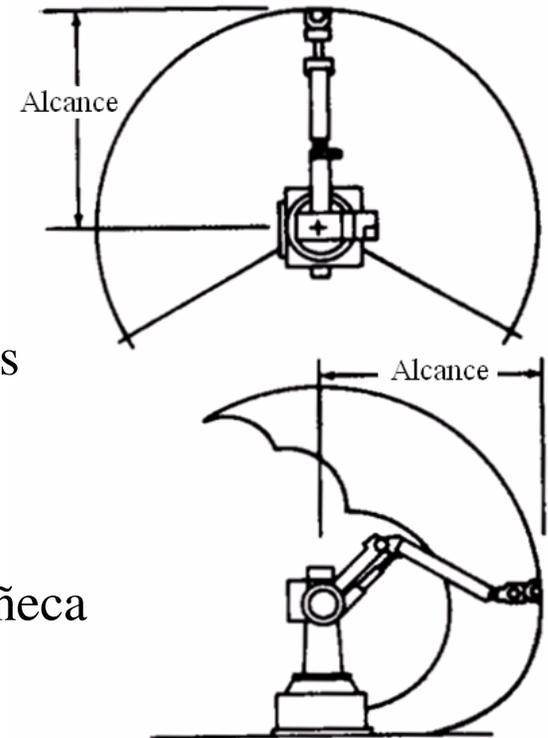
▶ **Coste**



Crterios de Implantación

● Selección del Robot: Área de trabajo

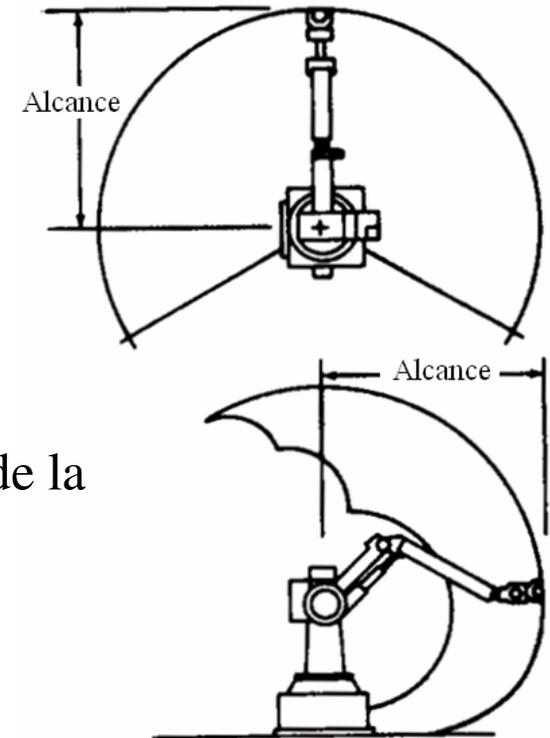
- ▶ Es el volumen espacial donde puede llegar el extremo del robot.
- ▶ Es determinado por el tamaño, forma y tipo de eslabones que integran el robot, así como por las limitaciones impuestas por el sistema de control.
- ▶ Nunca debe utilizarse la herramienta colocada en la muñeca para calcular el área de trabajo.
- ▶ Suele encontrarse en los catálogos mediante un dibujo acotado o en forma numérica mediante el recorrido de cada articulación.
- ▶ Debe elegirse de modo que su área de trabajo le **permita llegar a todos los puntos necesarios para llevar acabo su tarea** (puntos de recogida de piezas, mesa de trabajo, puntos de salida, intercambiador de herramientas, etc..)





Crterios de Implantación

- **Selección del Robot: Área de trabajo**
 - ▶ Considerar que en los puntos más alejados y más cercanos existen limitaciones en la orientación.
 - ▶ Considerar los puntos singulares.
 - ▶ Considerar las posibles colisiones con otros elementos de la célula.

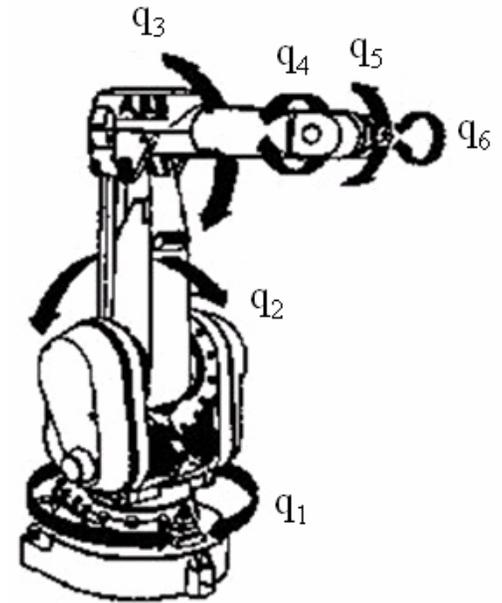




Criterios de Implantación

● Selección del Robot: Grados de libertad

- ▶ El número de grados (GDL) con que cuenta un robot determina su accesibilidad y su capacidad para orientar su herramienta.
- ▶ En general para los robots comerciales el número de grados de libertad coincide con el número de articulaciones.
- ▶ El número de GDL viene determinado por la aplicación.
- ▶ Aplicaciones (*pick and place*, paletizado) en general los objetos se recogen y depositan en planos horizontales, en estos casos un robot con **3 GDL** para posicionar y a lo sumo uno para orientar suelen ser suficientes.
- ▶ Algunos fabricantes proporcionan un número de GDL ampliables en 1 o 2 de manera opcional.
- ▶ Los GDL extras suelen incorporarse en el extremo o la base del robot.

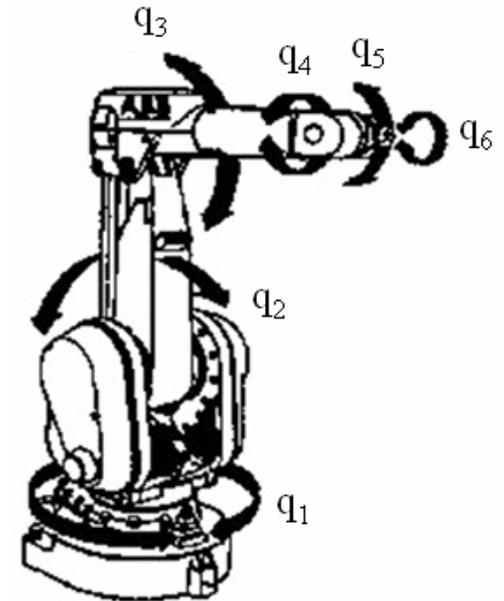




Criterios de Implantación

- **Selección del Robot: Grados de libertad**

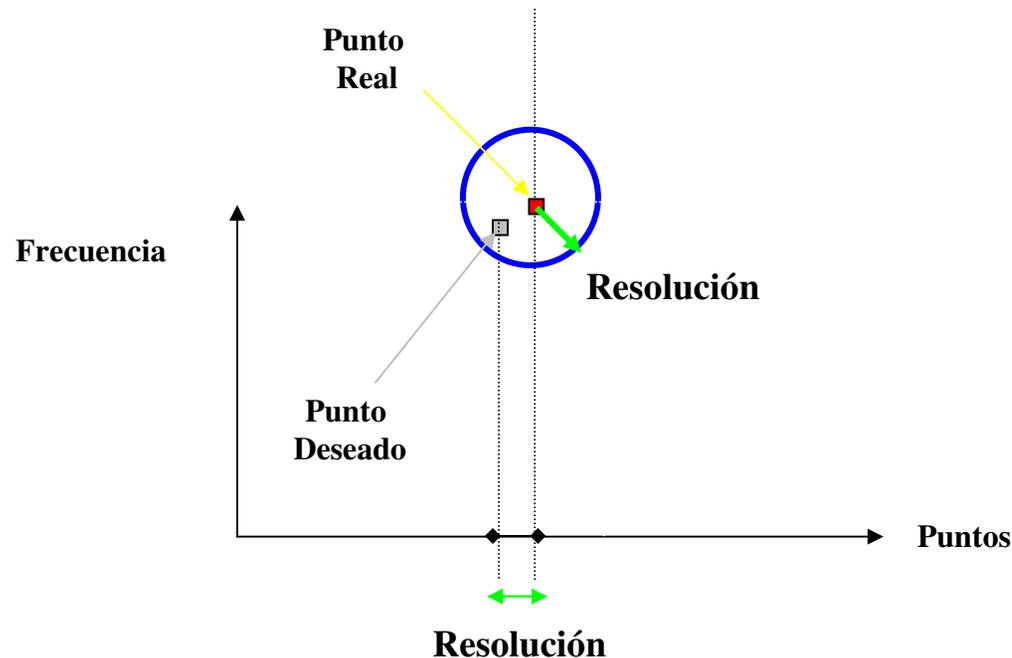
- ▶ El incremento de los GDL lleva implícito un considerable aumento en el costo del robot.
- ▶ En general para los robots comerciales el número de grados de libertad coincide con el número de articulaciones.
- ▶ Aplicaciones como pintura, soldadura de arco y colocación de adhesivos, suelen requerir **6 GDL**.





Criterios de Implantación

- Selección del Robot: **Resolución**

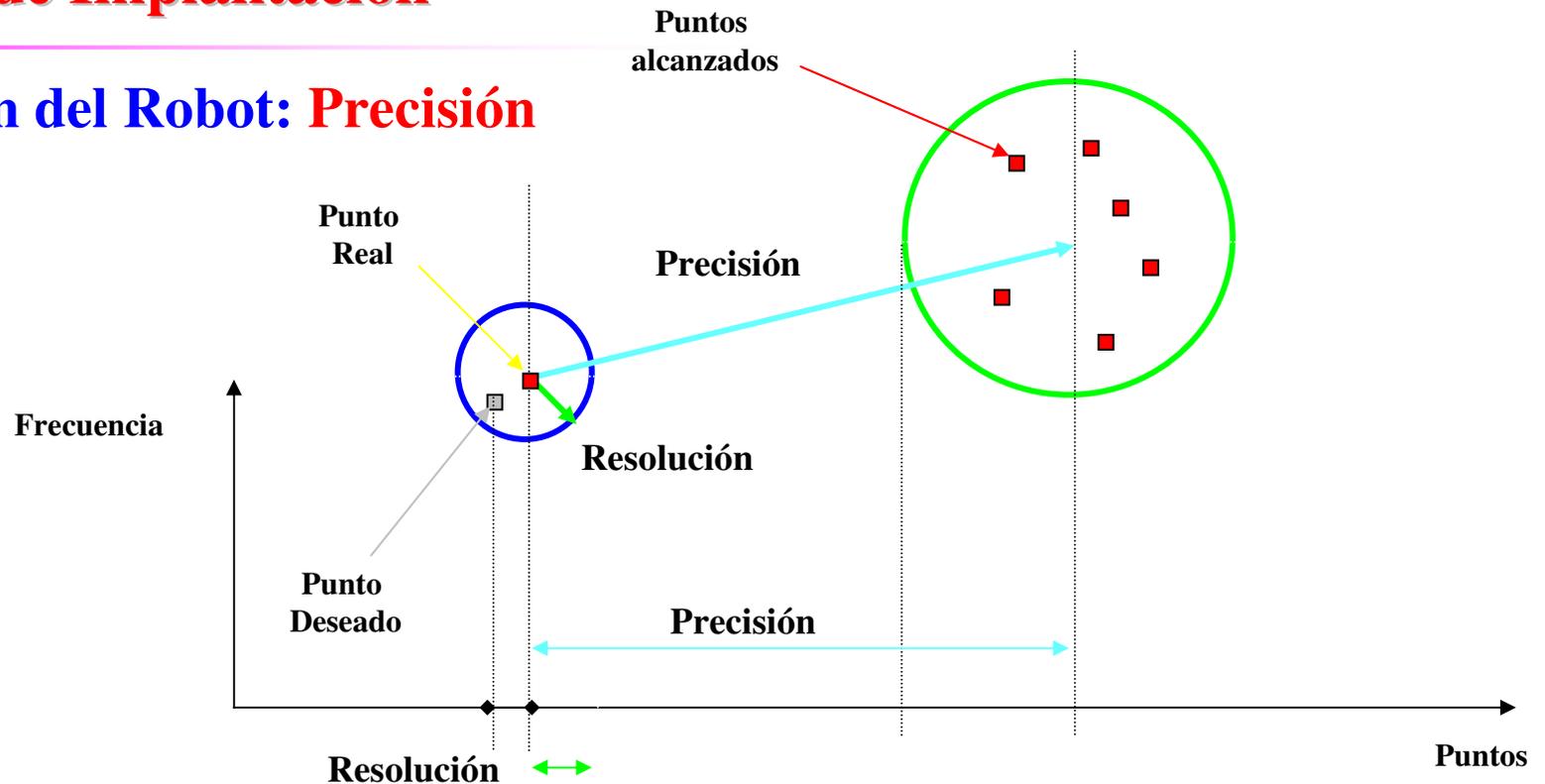


► **Resolución:** Es el mínimo incremento que puede aceptar la unidad de control del robot. Su valor está limitado por la resolución de los captadores de posición y convertidores A/D y D/A.



Criterios de Implantación

● Selección del Robot: Precisión



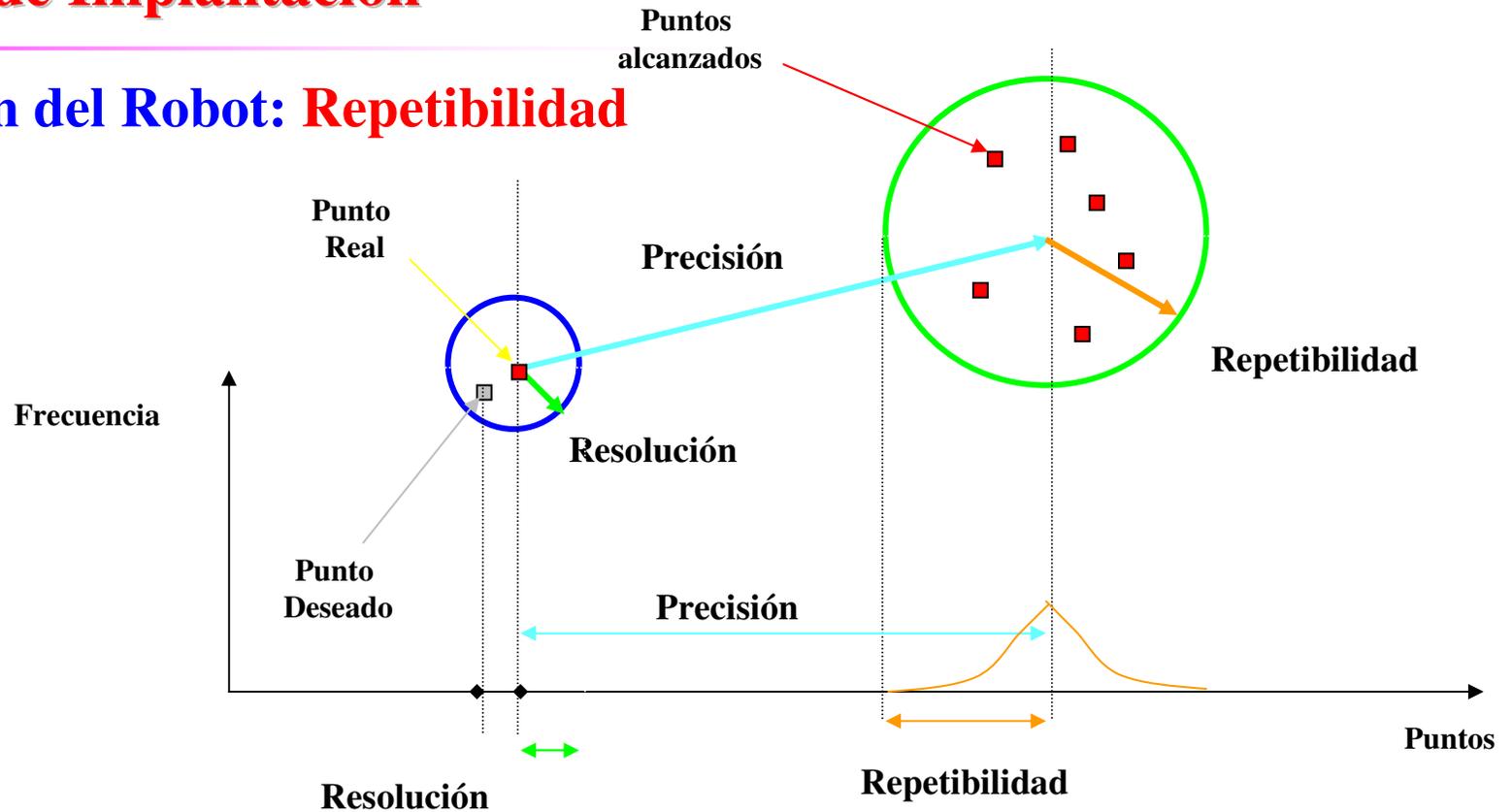
► **Precisión:** Distancia entre el punto programado (normalmente de manera textual) y el valor medio de los puntos realmente alcanzados al repetir el movimiento varias veces con carga y temperatura nominales.

Su origen se debe a errores de calibración, errores de redondeo en el cálculo de la transformación cinemática, térmicos, etc.



Criterios de Implantación

- Selección del Robot: **Repetibilidad**



- ▶ **Repetibilidad:** Radio de la esfera que abarca los puntos alcanzados por el robot tras suficientes movimientos, al ordenarle ir al mismo punto de destino programado, con condiciones de carga y temperatura iguales



Criterios de Implantación

● **Selección del Robot: Velocidad**

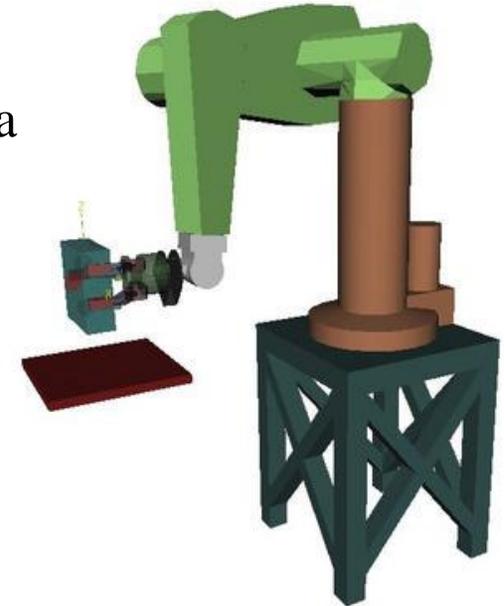
- ▶ La velocidad a la que puede moverse un robot y la carga que transporta, están inversamente relacionados.
- ▶ Suele existir también una relación de orden inverso entre el error de posicionamiento y la velocidad del robot.
- ▶ Puede darse por la velocidad de cada una de sus articulaciones o por la velocidad de su extremo, siendo ésta última más imprecisa.
- ▶ La velocidad es un dato relevante para el cálculo de los tiempos de ciclo, especialmente en aplicaciones de ensamblaje o manipulación.
- ▶ En general los tiempos de arranque y parada son proporcionalmente más significativos que la correspondiente al movimiento a la velocidad nominal.
- ▶ Los movimientos suelen ser cortos, en consecuencia el dato de velocidad nominal (régimen permanente) proporcionado por el fabricante, pocas veces es alcanzado, salvo en movimientos largos.



Crterios de Implantación

- **Selección del Robot: Capacidad de carga**

- ▶ La capacidad de carga de un robot para una determinada tarea está condicionada por el tamaño, la configuración y el sistema de accionamiento del propio robot.
- ▶ Al evaluar la carga a manipular debe considerarse el peso de las piezas a manipular y el propio peso de la herramienta o pinza que emplee el robot.
- ▶ Se considera además, el momento que genera la pieza en el extremo del robot.
- ▶ En general el fabricante proporciona las cargas para las cuales el robot no pierde sus prestaciones dinámicas.





Criterios de Implantación

● **Selección del Robot: Características deseables en un robot según su aplicación**

☞ **Pintura**

- Programación por guiado
- Campo de acción similar al humano
- Estructura antropomórfica
- 6 GDL

☞ **Procesado**

- Sistema de programación
- Campo de acción similar al humano
- Control de trayectoria continua
- 5-6 GDL

☞ **Paletización**

- Elevada capacidad de carga
- Relación grande entre área de trabajo y tamaño del robot
- Control PTP

☞ **Ensamblado**

- Elevada precisión y rapidez
- Campo de acción similar al humano
- Potencia del sistema de programación
- Sistema sensorial



Crterios de Implantación

● Selección del Robot: Características de algunos robots comerciales

Fabricante	Modelo	Aplicación	Configuración	GDL	Alcance horizon. (mm)	Carga máxima (kg)	Veloc. máxima (mm / s)	Repetibilidad (mm)	Accionamiento	Control trayec.
ABB	IRB 2400	Uso general	Angular	6	1542	10	4000	0.08	Eléctrico (c.a.)	PTP CP
	TR 5002	Pintura	Angular	6	2574	5	2000	1.0	Eléctrico (c.a.)	PTP CP
Adept	Adept Three	Uso general	SCARA	4	1070	25	11000	0.025	Directo (c.c.)	PTP CP
DEA	BRAVO 2205	Medición	Cartes.	3	x=2525 y=1185	-	500	0.3	Eléctrico (c.c.)	PTP CP
FANUC	ARC Mate 120 / S-12	Sold. Arco	Angular	6	1605	12	135 (°/s)	0.1	Eléctrico (c.a.)	PTP CP
	L-1000	Corte láser	Polar	5	1200	5	-	0.05	Eléctrico (c.a.)	PTP CP
Kremlin	AKR 3000	Pintura	Angular	7	2500	15	2000	1.0	Hidráulico	CP
KUKA	KR 125	Uso general	Angular	6	2410	125	1000	0.2	Eléctrico (c.a.)	PTP CP
Stäubli	PUMA560	Uso general Investigar	Angular	6	900	4	4700	0.1	Eléctrico (c.c.)	PTP CP
	RX 90	Uso general	Angular	6	900	12	9800	0.02	Eléctrico (c.c.)	PTP CP
Yaskawa	Motoman SK 16-6	Uso general	Angular	6	1775	6	140 (°/s)	0.1	Eléctrico (c.a.)	PTP CP
	Motoman S604	Montaje Manipular	SCARA	4	1950	60	-	0.2	Eléctrico (c.c.)	PTP CP



Criterios de Implantación

● Seguridad

- ▶ El robot tiene un mayor índice de riesgo que otras máquinas automáticas.

Comparación entre un robot industrial y otras máquinas de control numérico

Robot industrial	Máquina convencional de control numérico
Movimiento simultáneo de varios ejes (seis o más).	Normalmente mueve simultáneamente sólo uno o dos ejes.
Movimiento independiente de cada uno de los ejes. Trayectorias complejas.	Movimientos limitados y con trayectorias simples.
Campo de acción fuera del volumen cubierto por la propia máquina. Espacio de trabajo no reconocido fácilmente.	Campo de acción dentro del volumen abarcado por la máquina.
Campo de acción solapado con el de otras máquinas y dispositivos.	Normalmente sin solapamiento del campo de acción.



Criterios de Implantación

● Seguridad

▶ Según el Instituto de Investigaciones de Seguridad en el Trabajo de Tokio, el **90 % de los accidentes** en líneas robotizadas ocurren durante las operaciones de **mantenimiento, ajuste, programación**, etc. Mientras que sólo el **10 %** ocurre durante el **funcionamiento** normal de la línea.

▶ Tipos de accidentes

- Electrocuciones al instalar o reparar los equipos
- Colisión del robot con el operario
- Aplastamiento del operario entre el robot y algún elemento fijo
- Proyección de una pieza o material transportado por el robot



Criterios de Implantación

- **Seguridad**

- ▶ **Causa de accidentes:**

- Trabajo en entornos hostiles
- Alta velocidad
- Mal funcionamiento del sistema de control del robot
- Acceso indebido a la zona de trabajo
- Errores en las etapas de programación y mantenimiento
- Rotura de partes mecánicas
- Sobrecarga del robot
- Herramientas peligrosas (láser, chorro de agua)



Criterios de Implantación

● **Seguridad**

▶ **Medidas de seguridad intrínsecas al robot:**

- Responsabilidad del fabricante
- A tomar en la fase de diseño del robot
 - Supervisión del sistema de control (chequeo interno)
 - Paradas de emergencia
 - Velocidad máxima limitada
 - Detectores de sobre esfuerzo
 - Pulsador de seguridad (en programación y pruebas)
 - Códigos de acceso o llaves
 - Frenos mecánicos adicionales
 - Comprobación de señales de autodiagnóstico



Criterios de Implantación

● **Seguridad**

▶ **Medidas de seguridad a tomar en el diseño de la aplicación:**

- Barreras de acceso a la célula
- Dispositivos de intercambio de piezas
- Movimientos condicionados
- Zonas de reparación
- Condiciones adecuadas en la instalación auxiliar

▶ **Medidas de seguridad durante la instalación y explotación:**

- Abstenerse de entrar en la zona de trabajo
- Señalización adecuada
- Prueba progresiva de los programas
- Formación adecuada



Crterios de Implantación

- **Seguridad**

- ▶ **Normativa legal:**

- Normativa Internacional ISO 10218:1992
- Normativa Americana ANSI/RIA R15.06-1992
- Normativa Europea EN 775



Crterios de Implantación

● **Justificación económica de la implantación**

▶ **Factores económicos y datos básicos:**

☞ **Tipo de instalación**

- Creación de una nueva aplicación

Estudio de diferentes alternativas de automatización

- Automatizar un proceso realizado manualmente

☞ **Costes de inversión**

- Coste de adquisición del robot
- Coste de adquisición de herramientas y equipos especiales
- Coste de ingeniería, instalación y puesta en marcha

☞ **Costes y beneficios de explotación**

- | | |
|-----------------|-------------------------------|
| - Mano de obra | - Aumento de calidad |
| - Mantenimiento | - Incremento de productividad |
| | - Formación y entrenamiento |



Crterios de Implantación

- **Justificación económica de la implantación**

- ▶ **Análisis económico:**

- ▢ **Ventaja del robot: versatilidad o flexibilidad para su uso en distintas tareas**

- Robot y CN como herramientas programables

- Considerar el valor de recuperación o de retorno del coste del robot



Resumen:

- Un robot industrial generalmente no trabaja como un elemento aislado, sino que forma parte de un proceso de fabricación que incluye muchos otros equipos.
 - El robot es considerado la parte principal de la denominada **célula de trabajo robotizada**.
 - El criterio de implantación de una célula robotizada implica factores, que van desde el posible rediseño del producto, hasta la definición detallada del lay-out o plano de implementación del sistema.
 - Además, de la selección más adecuada del robot, habrá que definir, e incluso diseñar, los elementos periféricos pasivos (mesas, alimentadores, utillajes, etc.) o activos (manipuladores secuenciales, máquinas CN, etc.) que intervienen en la célula.
 - La utilización de **herramientas informáticas**, como sistemas CAD, simuladores específicos para robots y simuladores de sistemas de fabricación flexible son de gran utilidad.
-



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 8 – Jueves 11 de Mayo 2006

Criterios de Implantación de un Robot Industrial



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

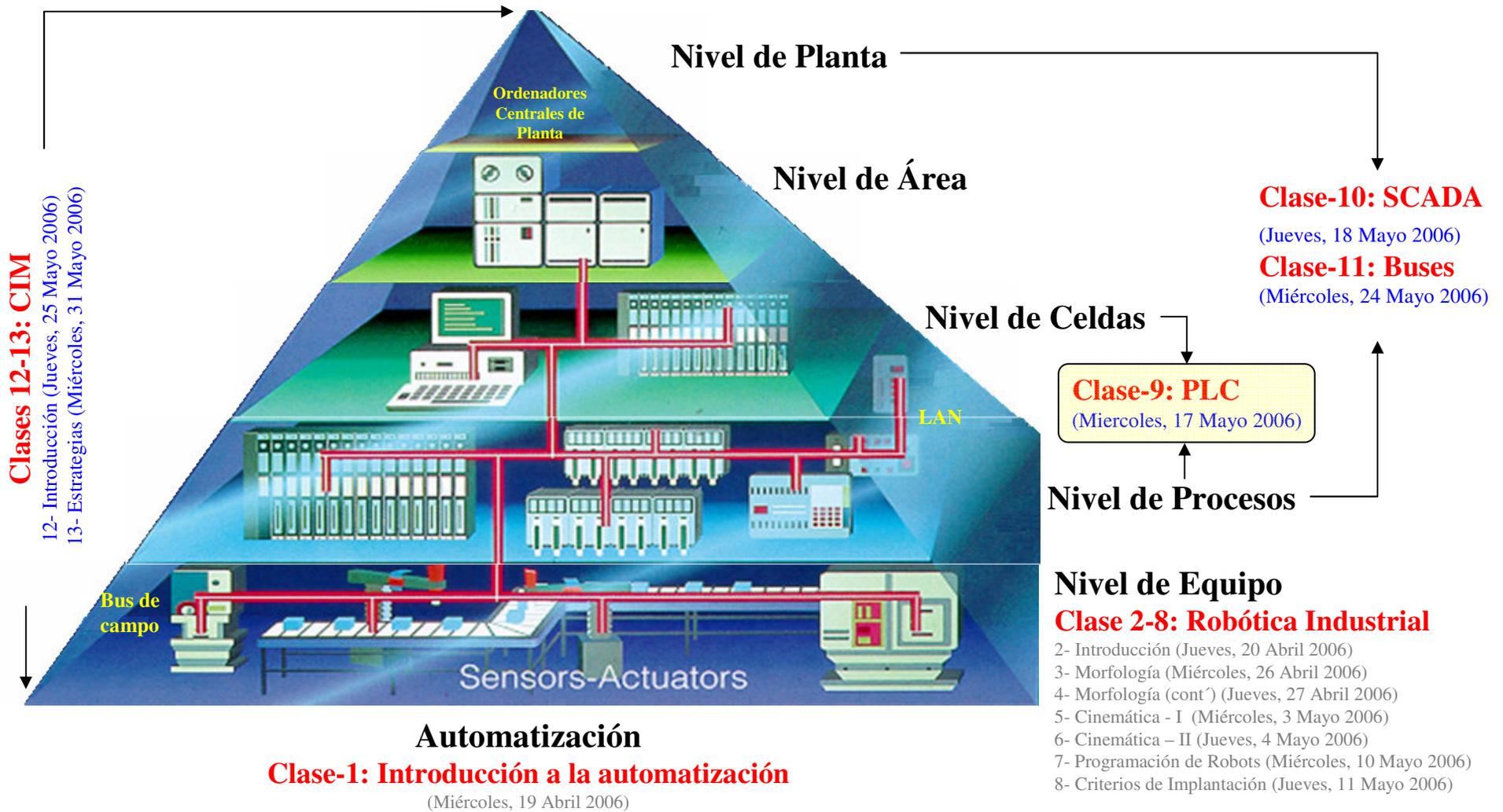
Clase 9 – Miércoles 17 de Mayo 2006

Autómatas Programables



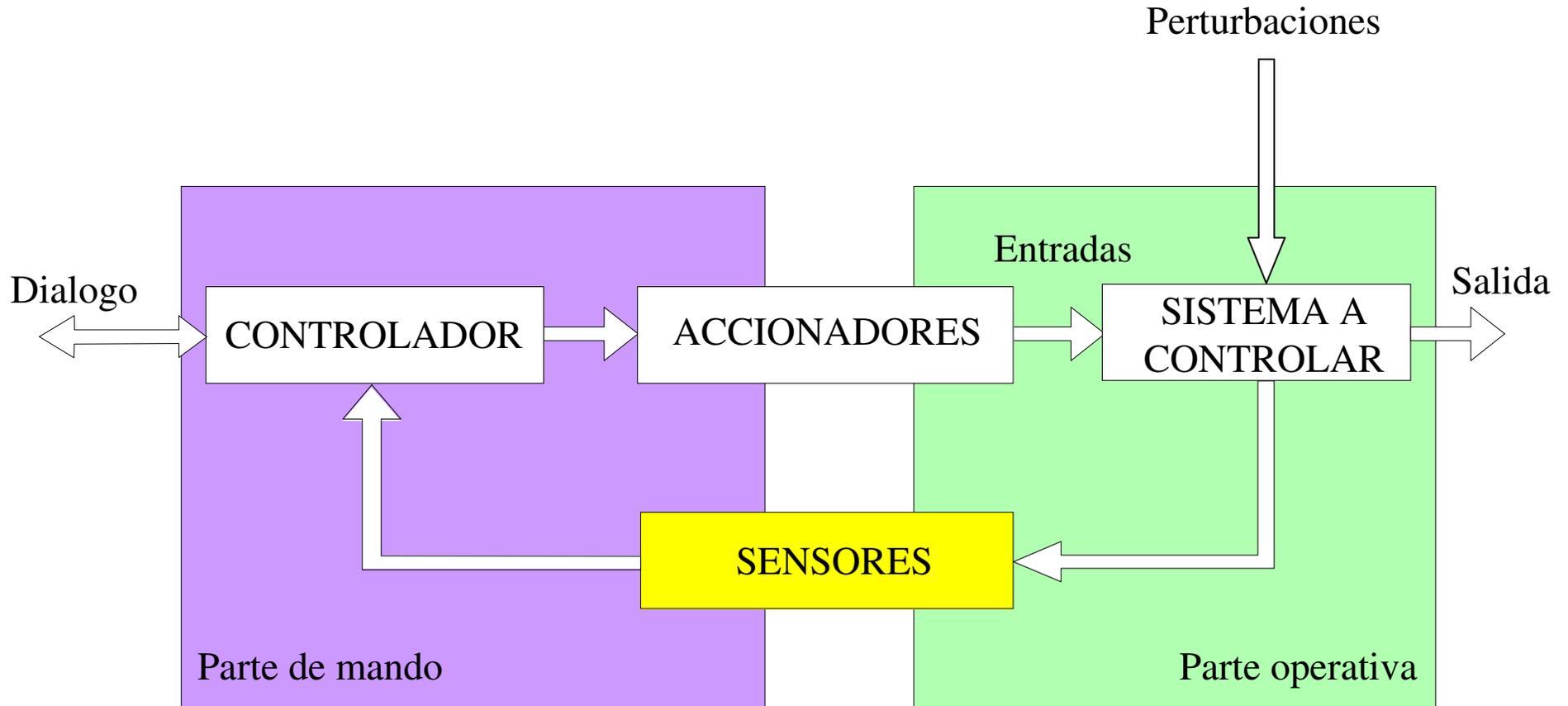
Automatización Industrial -II

Introducción





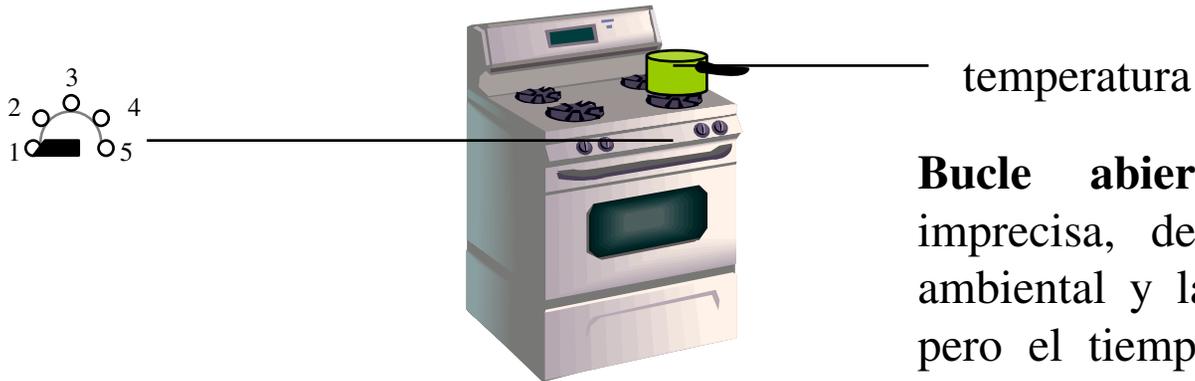
Estructura de los sistemas automatizados



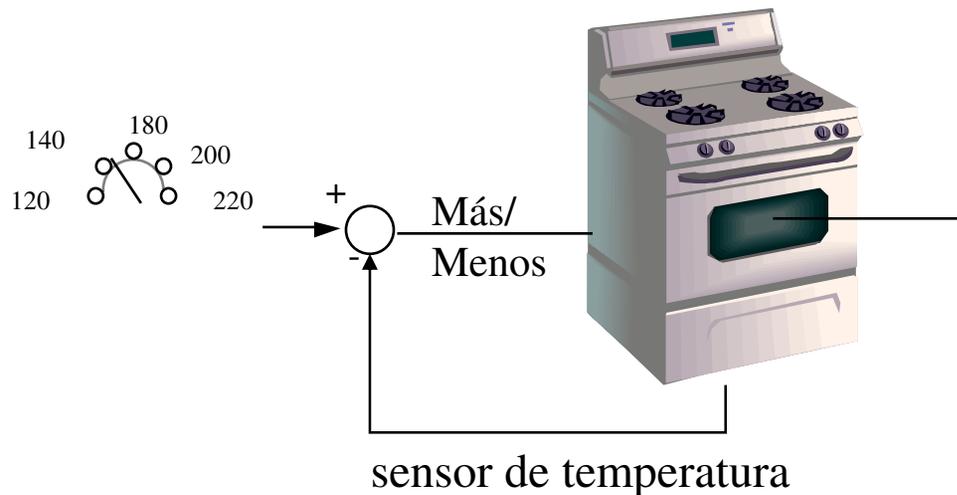


Tecnologías de control

- Control en bucle abierto y control en bucle cerrado



Bucle abierto: la temperatura es imprecisa, depende de la temperatura ambiental y la cantidad que se calienta pero el tiempo de calentamiento puede ser modulado.



Bucle cerrado: la temperatura es controlada, se requiere la medida de la variable de salida (la temperatura)



Tecnologías de control

- **Control en bucle abierto y control en bucle cerrado**

Sistema a bucle abierto

- Adquisición de datos y procesamiento preliminar
- Transferir datos entre la planta y el operador
- Monitorización del estado de la planta
- Grabación de historia de acceso
- Simulación y entrenamiento
- Algoritmo de optimización de procesos

Sistemas a bucle cerrado

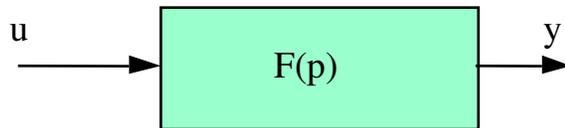
- Protección
- Regulación
- Control secuencial de procesos



Tecnologías de control

- Control continuo y control discreto

Control continuo

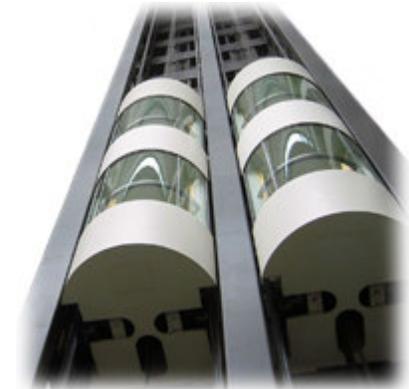
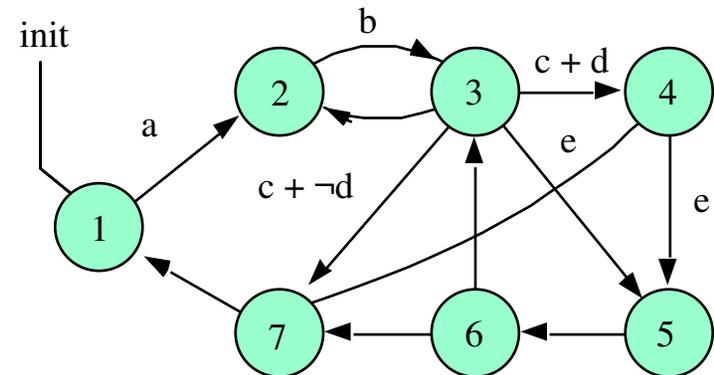


Ejemplos:

Accionadores, hornos, reactores químicos



Control discreto





Sistemas de eventos discretos

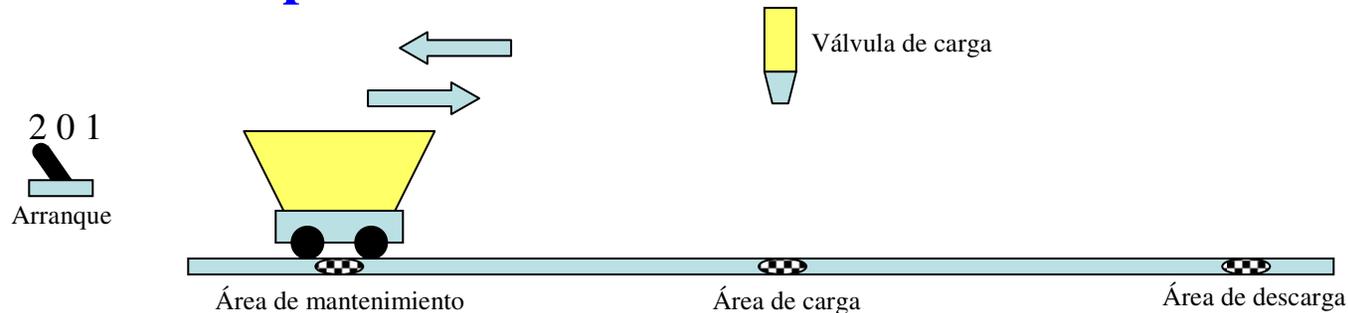


The screenshot shows a software window titled "PLC Lab - Elevator Control.vpd". The interface includes a menu bar (File, Edit, View, Draw, Project, Shape, Simulation, PLC, User Tool, Window, Help), a toolbar with various icons, and a main workspace. On the left, a tree view shows the project structure: Elevator Control, Virtual Plant, Elevator Control, Script, Elevator Control, Tag, Elevator Control, 3D Virtual Plant, Elevator Control, Ladder Diagram, and Elevator Control. The main workspace is divided into three panes: a control panel on the left with buttons for "3rd Floor", "2nd Floor", and "1st Floor"; a central 3D model of an elevator car in a shaft; and a variable declaration table on the right. The variable table lists: %M000 (red square), %M001 (empty), %M002 (empty), %M003 (empty), %M004 (empty), %M005 (empty). A green text box is overlaid on the 3D model with the text "Un ascensor controlado por un PLC" and "Video". The status bar at the bottom shows "Ready", "x= 97 y= 14", "100%", "Simulation", and "NUM".



Representaciones de sistemas de eventos discretos

● Sistema de Transporte de Material



El funcionamiento del sistema es el siguiente:

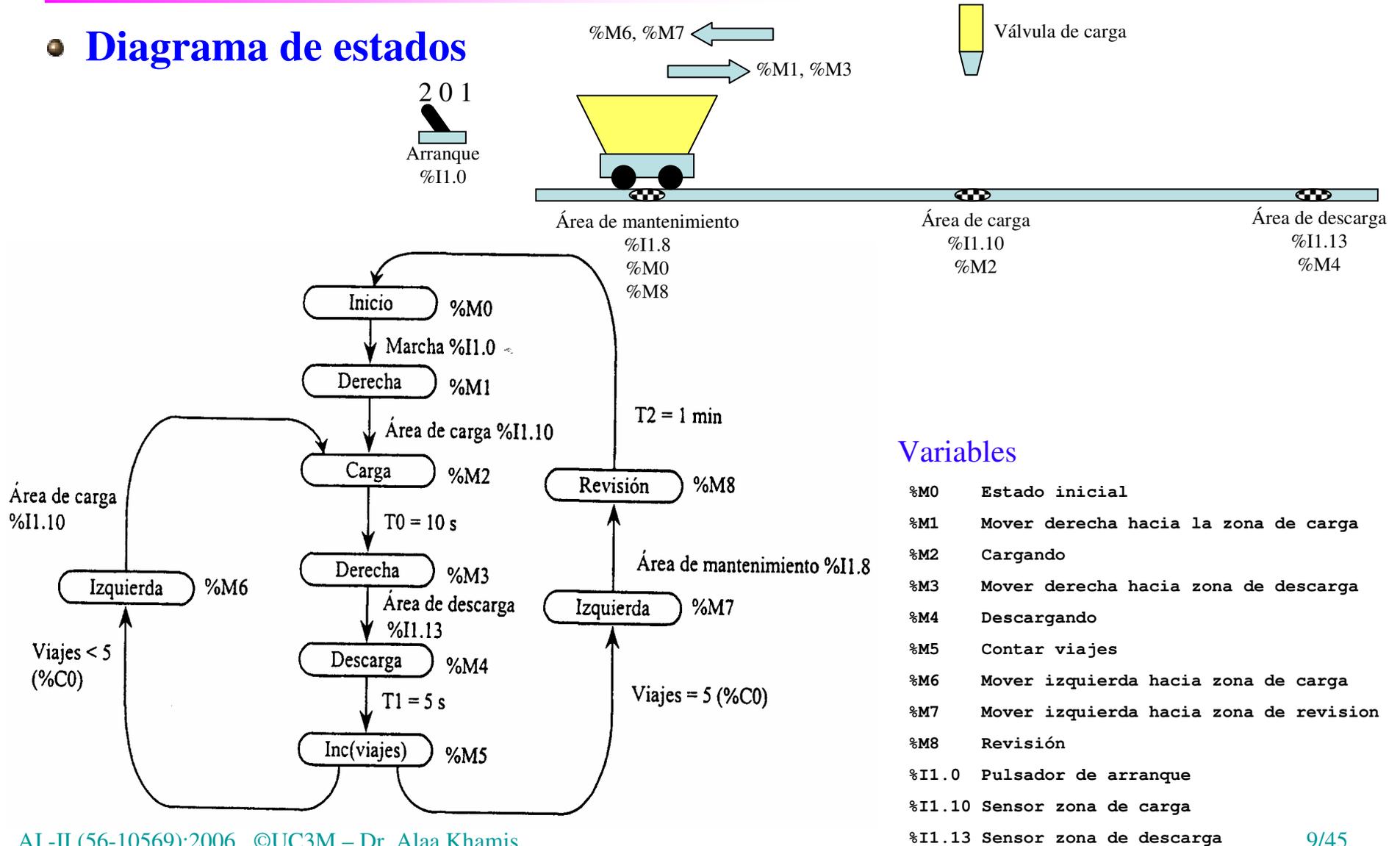
- En el estado inicial la vagoneta se encuentra en el área de mantenimiento.
- El sistema se activa mediante un pulsador.
- Se pone en marcha la vagoneta hacia la derecha hasta llegar a la zona de carga (se detecta mediante un sensor) y se detiene.
- Se abre la válvula de carga, durante 10 segundos, tiempo empleado en llenar la vagoneta.
- Una vez llena se desplaza hacia la zona de descarga donde vacía su contenido en 5 segundos.
- Vuelve a la zona de carga y repite el proceso 5 veces.
- Concluida la quinta descarga, vuelve a la zona de mantenimiento para una inspección de la vagoneta; la revisión dura 1 minuto.
- Terminada la revisión se puede repetir el ciclo actuando sobre el pulsador.
- Durante el proceso permanecerá encendido un piloto indicando el estado activo.

Nota: la carga, descarga y revisión se simulará con luces (leds) en la maqueta.



Representaciones de sistemas de eventos discretos

● Diagrama de estados





Representaciones de sistemas de eventos discretos

● Diagrama funcional

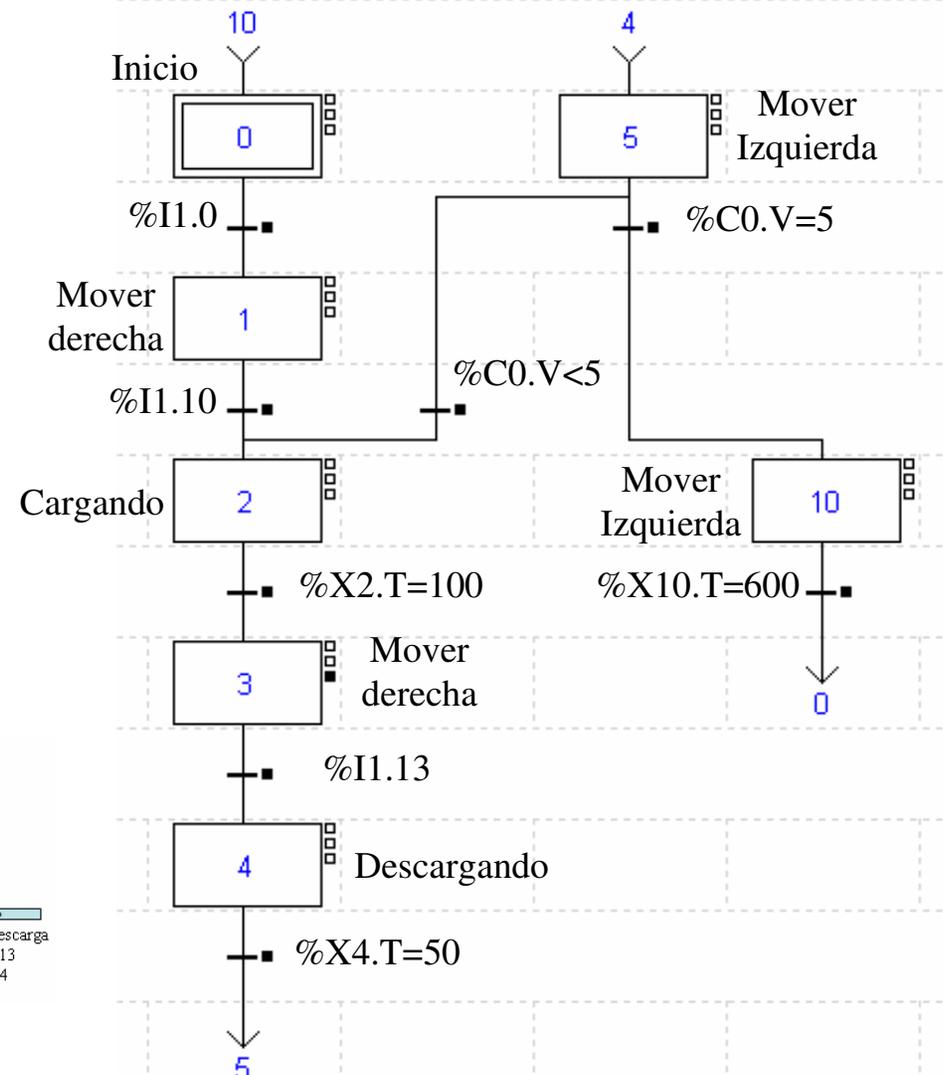
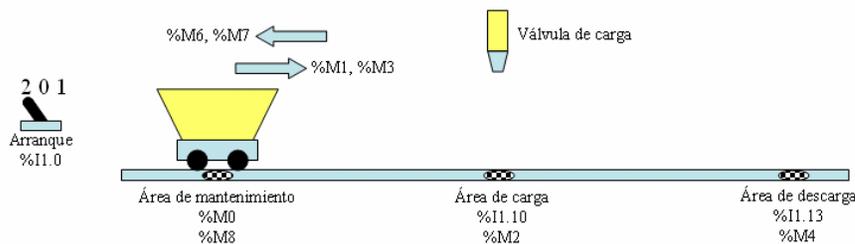
▶ Grafcet: GRAPHe de Commande

Etape-Transition

Norma IEC 848

Relación con las redes de Petri

- Etapas (lugares)
- Transiciones
- Marcado binario





Representaciones de sistemas de eventos discretos

● Redes de Petri

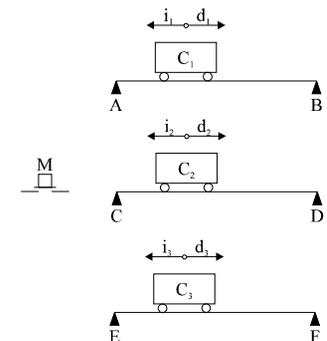
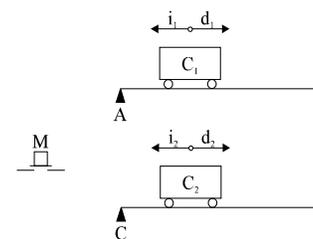
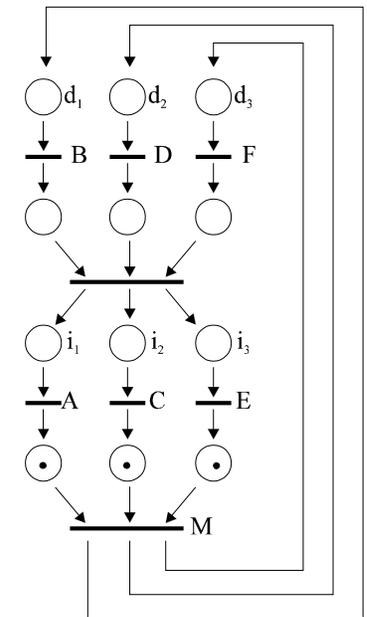
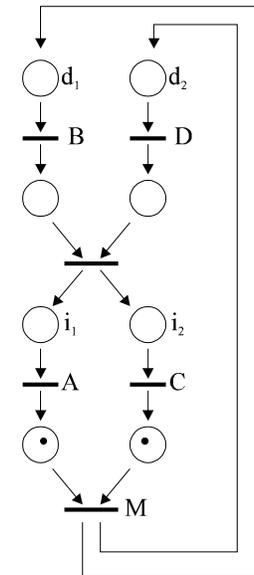
▶ Elementos de una red de Petri

- Lugares
- Transiciones
- Flechas
- Marcas

▶ Los lugares representan acciones

▶ Las transiciones representan eventos

▶ El marcado de la red representa el estado del sistema (marcado binario) debe interactuar con otros equipos, máquinas, procesos, etc.





Control de sistemas de eventos discretos

- **Computadores de uso general vs. autómatas programables**



Ventajas de los computadores de uso general

- Precio
- Capacidad de cálculo
- Interfaz gráfica
- Normalización

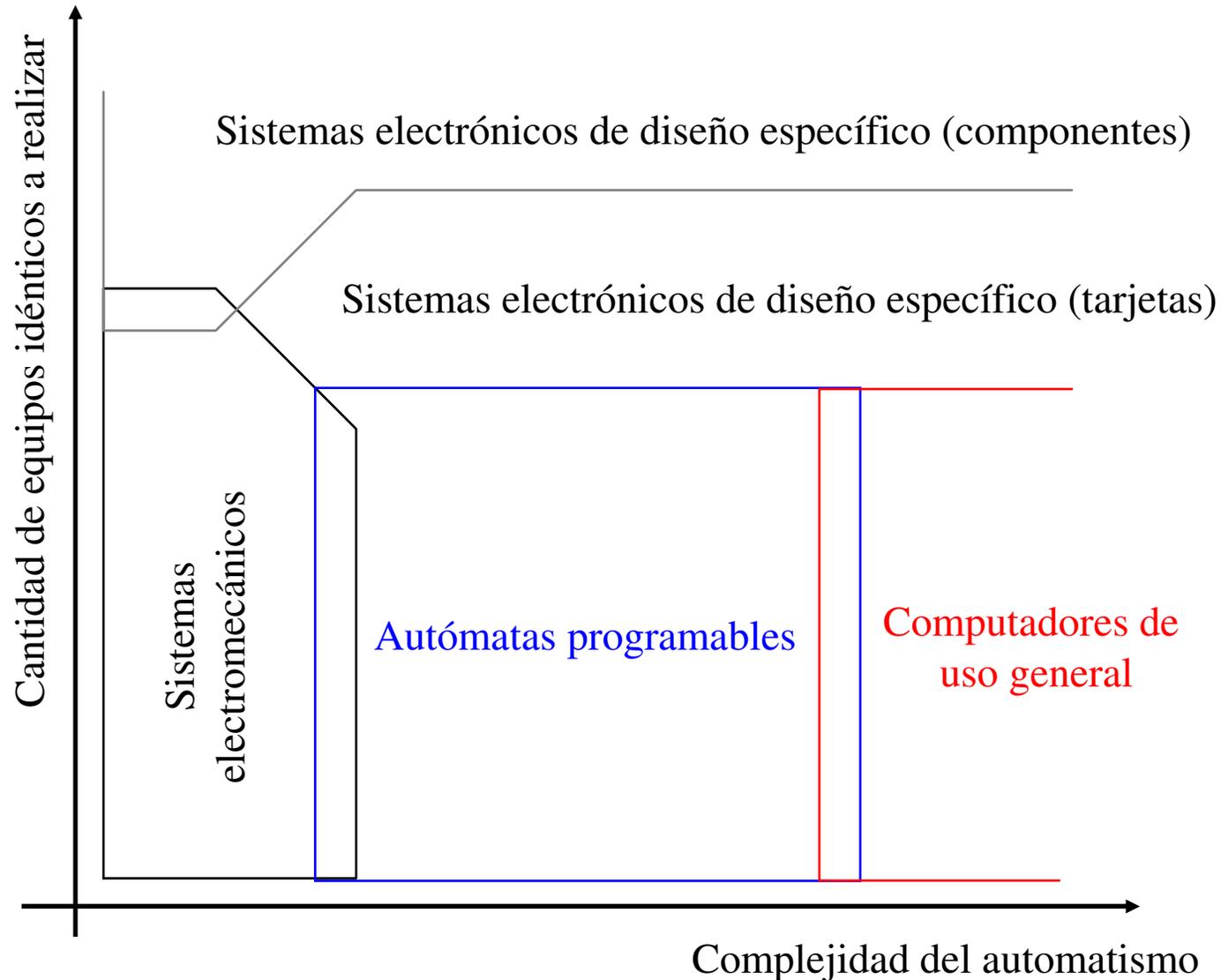


Ventajas de los autómatas programables

- Robustez
- Arquitectura adaptada a la aplicación
- Modularidad y versatilidad entradas /salidas (tipo y numero)
- Software



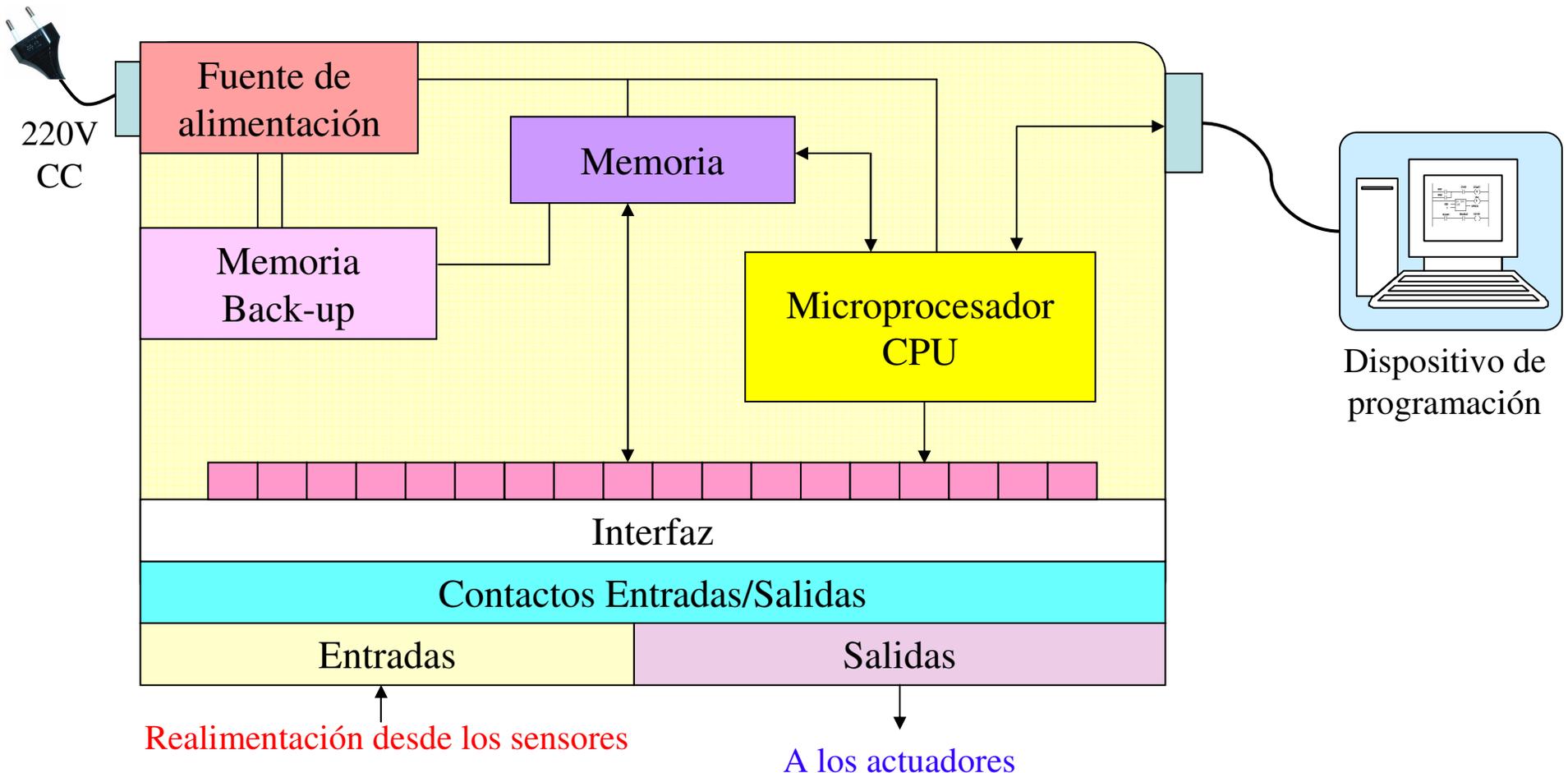
Control de sistemas de eventos discretos





Autómatas Programables

PLC (**P**rogrammable **L**ogic **C**ontroller) significa Controlador Lógico Programable. Un PLC es un dispositivo usado para controlar sistemas de eventos discretos.





Autómatas Programables

● **Componentes del PLC**

- ▶ **Unidad central de procesamiento (CPU):** que constituye el "cerebro" del sistema y toma decisiones en base a la aplicación programada.
- ▶ **Módulos para señales digitales y analógicas (I/O)**
- ▶ **Procesadores de comunicación (CP)** para facilitar la comunicación entre el hombre y la máquina o entre máquinas. Se tiene procesadores de comunicación para conexión a redes y para conexión punto a punto.
- ▶ **Módulos de función (FM)** para operaciones de cálculo rápido.

Existen otros componentes que se adaptan a los requerimientos de los usuarios:

- ▶ **Módulos de suministro de energía**
- ▶ **Módulos de interfaces** para conexión de racks múltiples en configuración multi-hilera



Autómatas Programables

● **Componentes del PLC**

▶ En los **módulos de entrada** pueden ser conectados:

- Sensores inductivos, capacitivos, ópticos
- Interruptores
- Pulsadores
- Llaves
- Finales de carrera
- Detectores de proximidad

▶ En los **módulos de salida** pueden ser conectados

- Contactores
- Electroválvulas
- Variadores de velocidad
- Alarmas



Autómatas Programables



● **Funcionamiento del PLC**

- ▶ El usuario ingresa el programa a través del dispositivo adecuado (un cargador de programa o PC) y éste es almacenado en la memoria de la CPU.
- ▶ La CPU, que es el "cerebro" del PLC, procesa la información que recibe del exterior a través de la interfaz de entrada y de acuerdo con el programa, activa una salida a través de la correspondiente interfaz de salida.
- ▶ Evidentemente, las interfaces de entrada y salida se encargan de adaptar las señales internas a niveles de la CPU. Por ejemplo, cuando la CPU ordena la activación de una salida, la interfaz adapta la señal y acciona un componente (transistor, relé, etc.)



Autómatas Programables

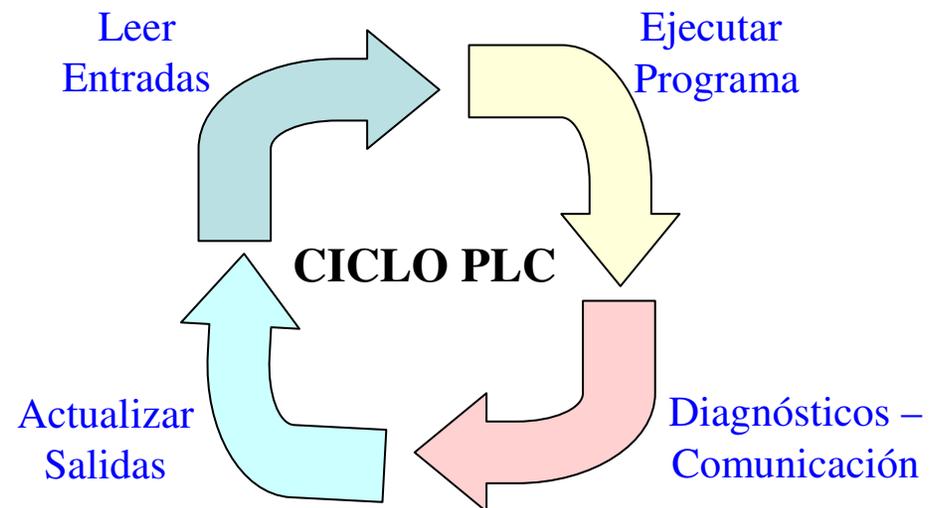


● Funcionamiento del PLC

- ▶ Al comenzar el ciclo, la CPU lee el estado de las entradas.
- ▶ A continuación ejecuta la aplicación empleando el último estado leído.

Una vez completado el programa, la CPU ejecuta tareas internas de diagnóstico y comunicación.

- ▶ Al final del ciclo se actualizan las salidas.
- ▶ El tiempo de ciclo depende del tamaño del programa, del número de E/S y de la cantidad de comunicación requerida.

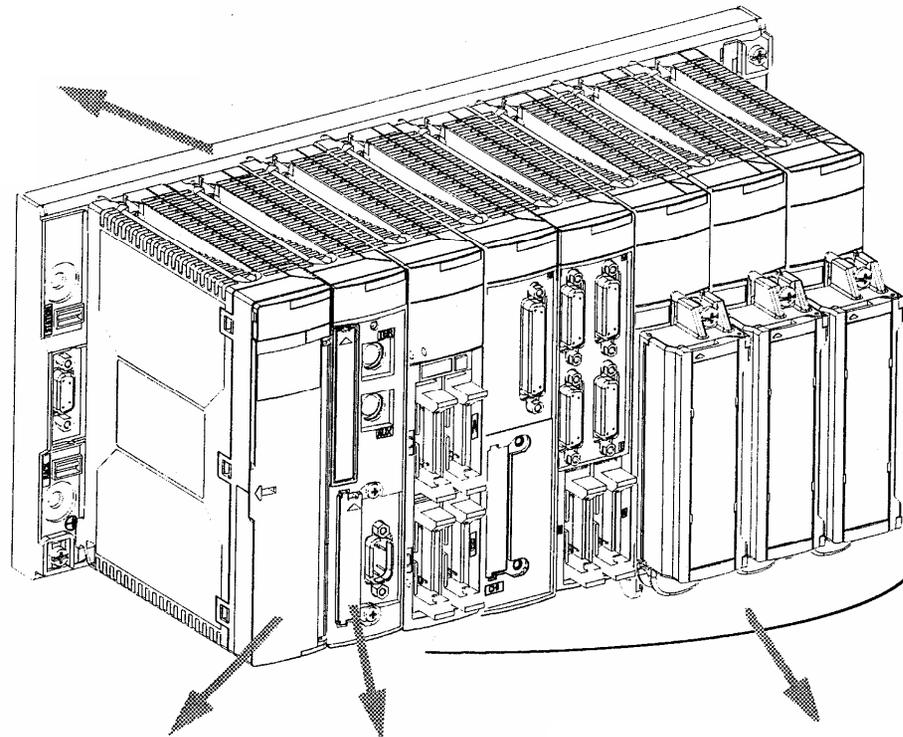




Autómatas Programables

● PLC - Télémécanique TSX

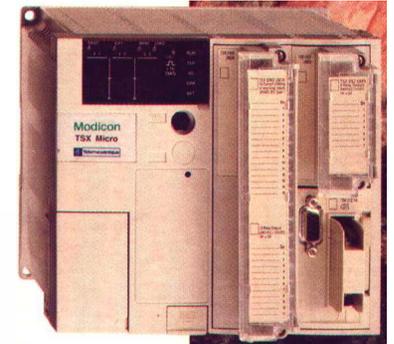
Rack modulares
6/8/12 posiciones
- Standard
- extensión



Módulo
Fuente Alim.

Módulo
CPU

Módulos específicos
Digitales, ANA, Comunic,
Contaje,...

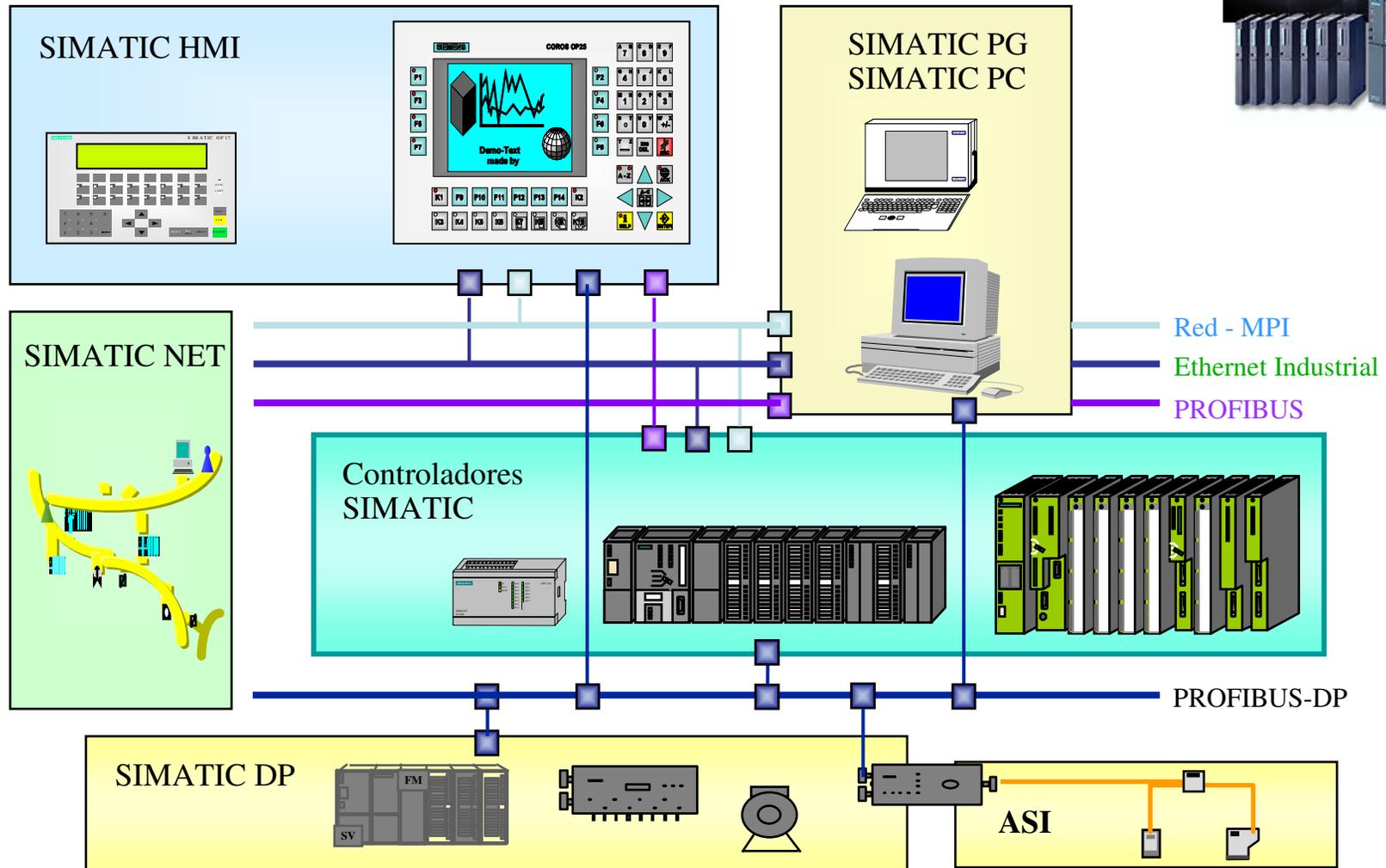




Autómatas Programables



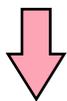
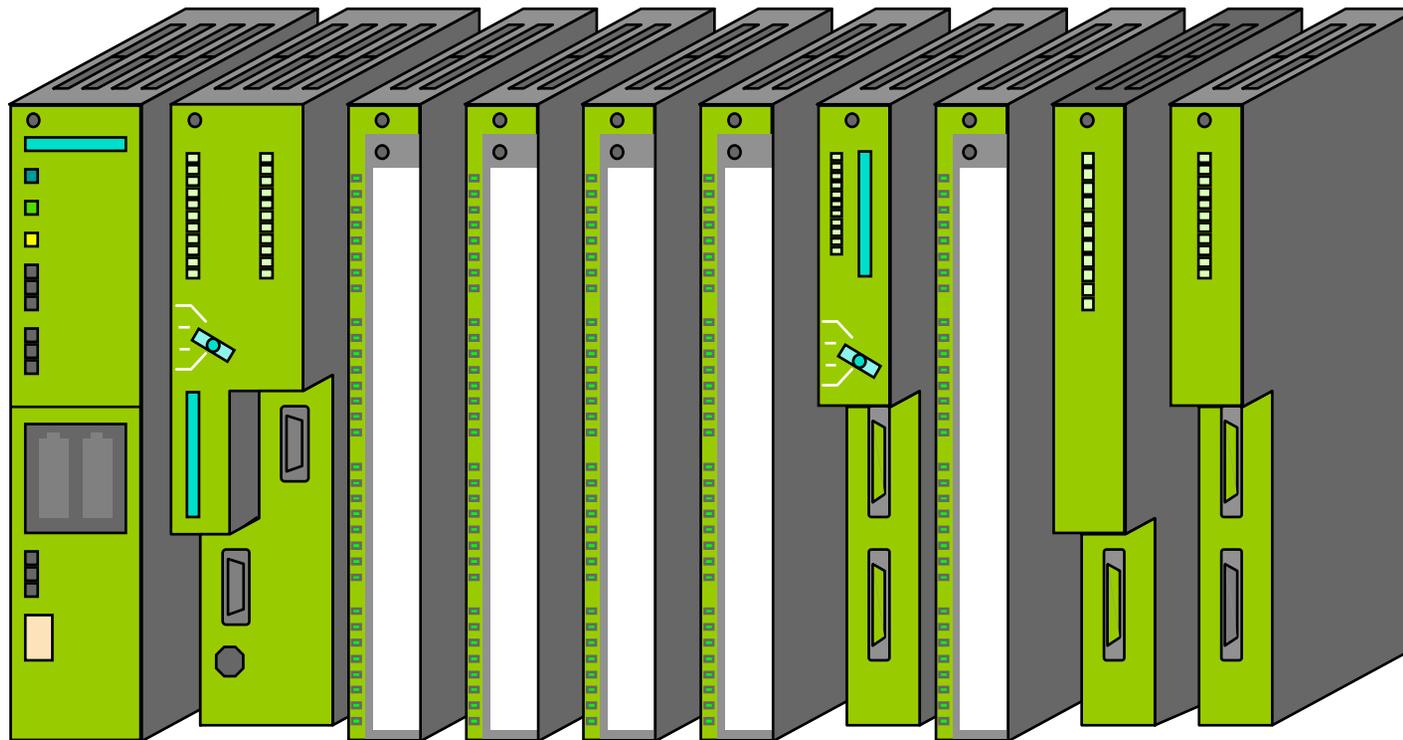
● PLC – Visión Global de Siemens SIMATIC



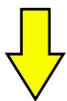


Autómatas Programables

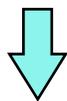
- PLC – Siemens Simatic S7-400



PS



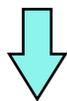
CPU



SM:
DI



SM:
DO



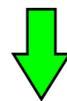
SM:
AI



SM:
AO



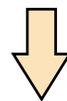
CP



FM



SM



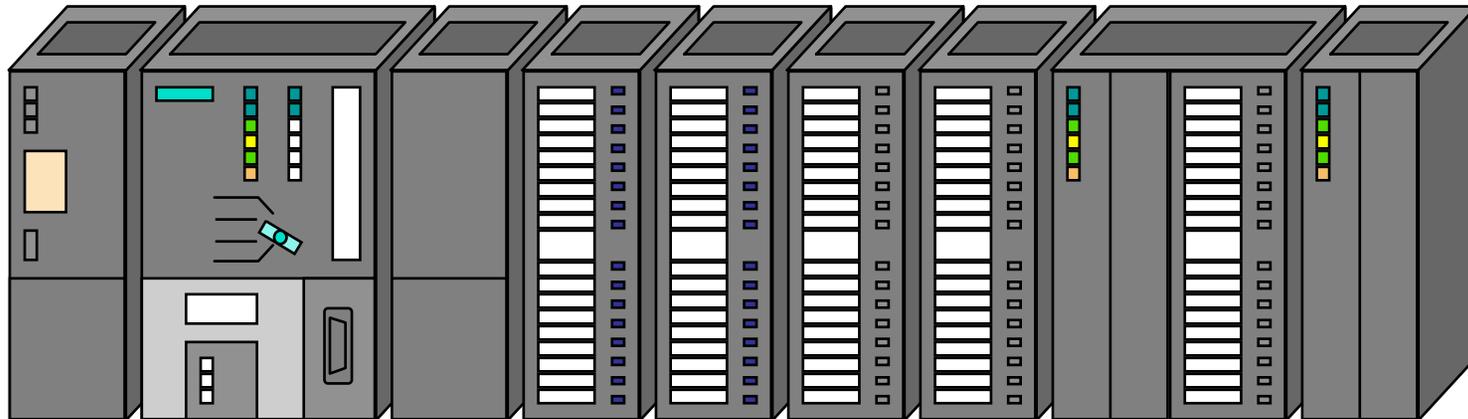
IM

PS: Power Supply
SM: Signal Modules
IM: Interface Modules
FM: Function Modules
CP: Communication
Modules



Autómatas Programables

● PLC – Siemens Simatic S7-300



↓
PS
(opcional)

↓
CPU

↓
IM
(opcional)

↓
**SM:
DI**

↓
**SM:
DO**

↓
**SM:
AI**

↓
**SM:
AO**

↓
FM:
- Contaje
- Posicionamiento
- Control en
Lazo Cerrado

↓
CP:
- Punto-a-Punto
- PROFIBUS
- Ethernet Industrial



Automatización Industrial -II

Autómatas Programables

- PLC – Siemens Simatic S7-200

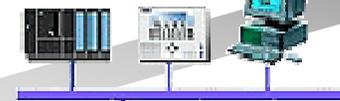


S7-200

LOGO!



S7-300



S7-400





Autómatas Programables



● Aplicaciones:

Tareas que:

- ▶ son repetitivas;
- ▶ requieren alta fiabilidad en la ejecución;
- ▶ se encuentran en ambientes industriales;
- ▶ no requieren flexibilidad;
- ▶ no requieren algoritmos complejos;
- ▶ son de eventos secuenciales o discreto;

Más información:

<http://www.plcs.net/contents.shtml>





Autómatas Programables

● Aplicaciones: Automatización de viviendas



Gestión de alarmas



Detección y aviso de intrusos (interior, exterior)



Detección, corte de suministro y aviso de fugas de agua



Detección, corte de suministro y aviso de fugas de gas



Detección y aviso de incendios



Simulación de presencia

Mediante iluminación, cargas y persianas. Gestión aleatoria.



Comunicaciones

Aviso de alarmas y conex./descon. de alarmas, calefacción, simulación de presencia, iluminación y cargas.



Calefacción

Grupos, horarios, termostatos, sonda de temperatura, visualización, ventanas abiertas



Control de cargas

Con./descon. de tomas de red, asignación de grupos, gestión horaria, presencia, temperatura



Iluminación

Asignación de grupos, gestión horaria, presencia, luminosidad



Toldos y persianas



Grupos, horarios y condiciones climáticas



Riego del jardín

Asignación de zonas, horarios, secuencialidad y condiciones climáticas

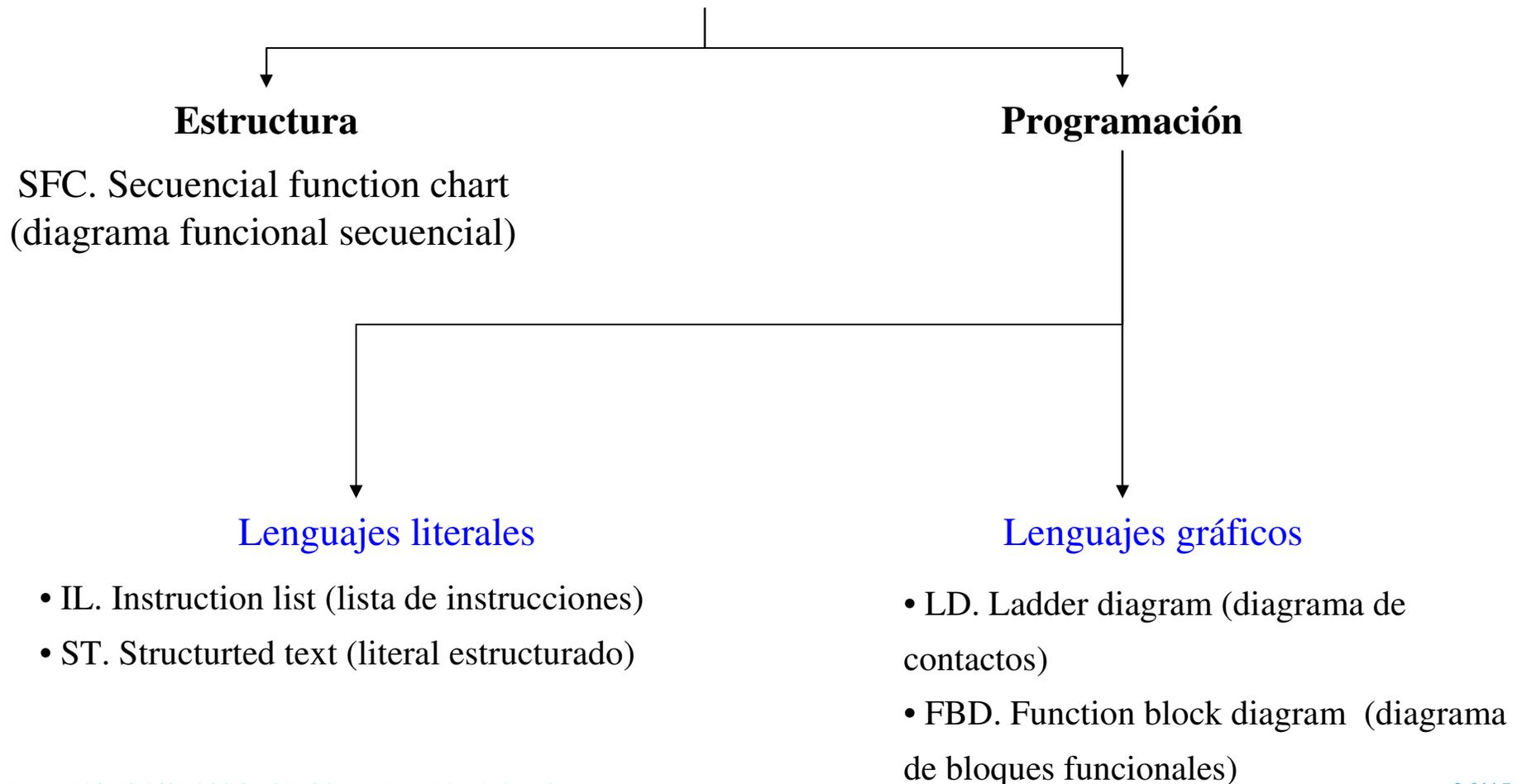


Autómatas Programables



● **Programación**

Norma IEC 1131-3





Autómatas Programables



- **Programación: Declaraciones según IEC 1131-3**

%LocalizaciónTamaño.i (...)

Entrada	I
Salida	Q
Memoria	M

Bit	
Bit	X
Byte (8 bits)	M
Palabra (16 bits)	W
Doble palabra (32 bits)	D
Cuadruple palabra	L

Ej.

%QX75 o %Q75

%MD48

%IW2.5.7.1



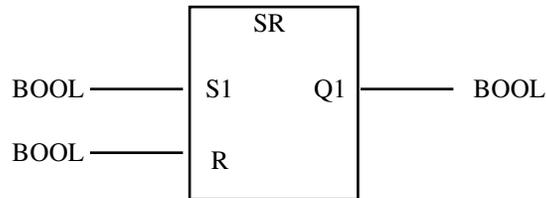
Autómatas Programables



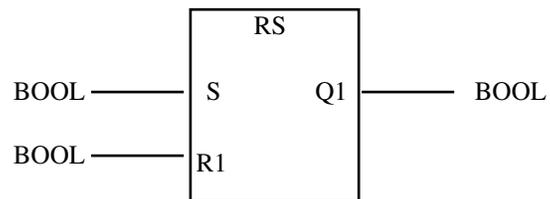
● Programación: Bloques de función estándar

▶ Biestables

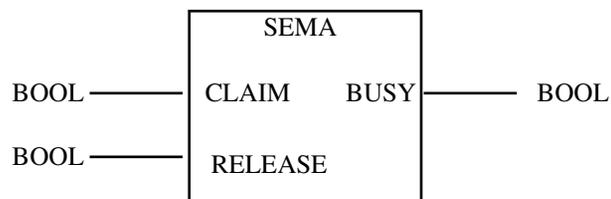
- RS (set dominante)



- RS (reset dominante)

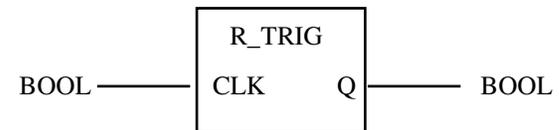


- Semáforo

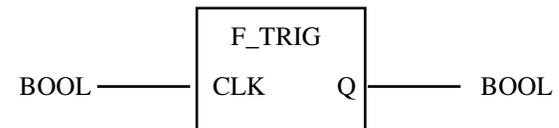


▶ Detectores de flanco

- Detector de flanco de subida



- Detector de flanco de bajada





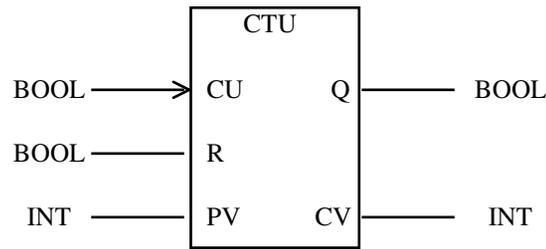
Autómatas Programables



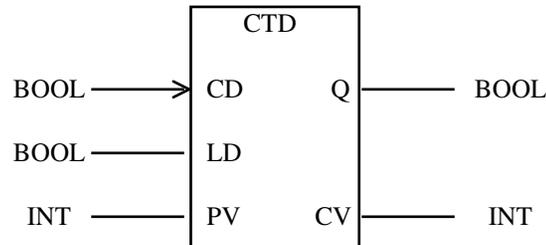
● Programación: Bloques de función estándar

▶ Contadores

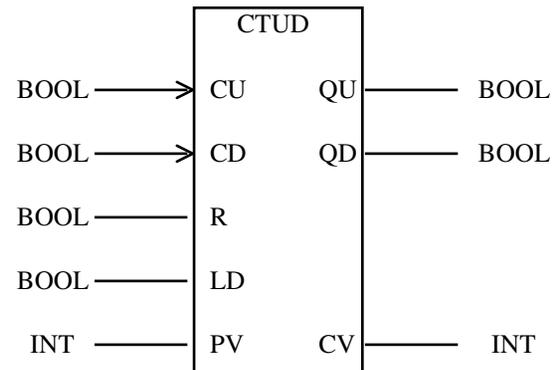
- Contador ascendente



- Contador descendente



- Contador ascendente/descendente





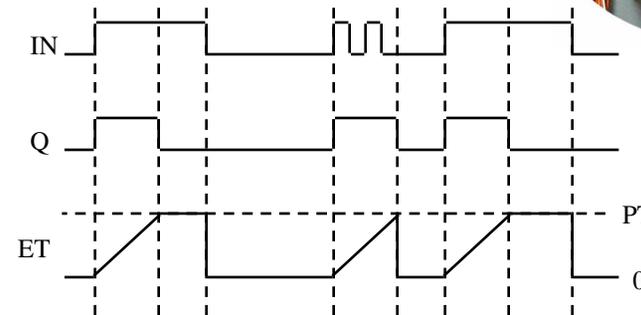
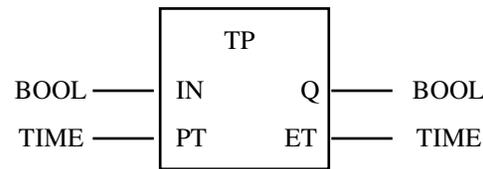
Autómatas Programables



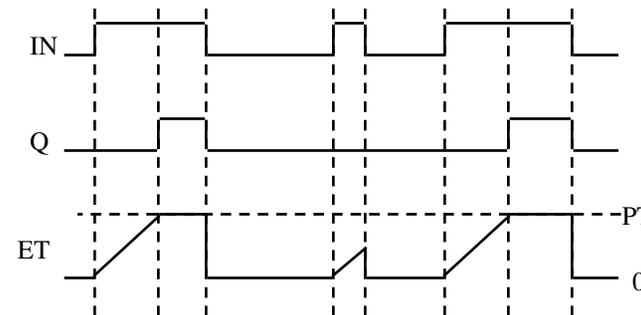
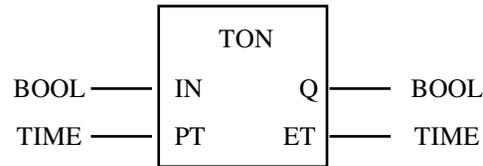
● Programación: Bloques de función estándar

▶ Temporizadores

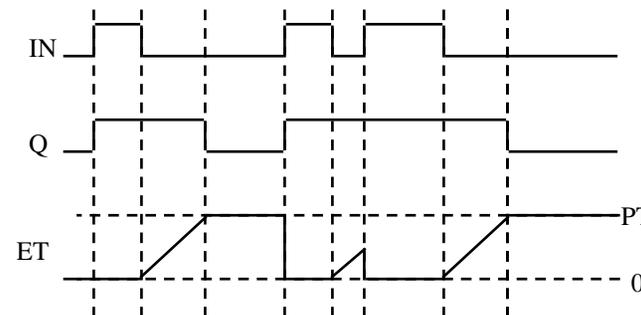
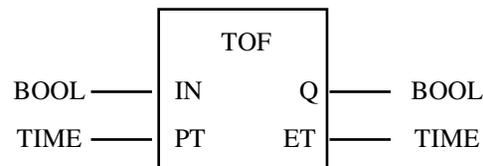
- Monoestable



- Retardo a la conexión



- Retardo a la desconexión





Autómatas Programables

Programación

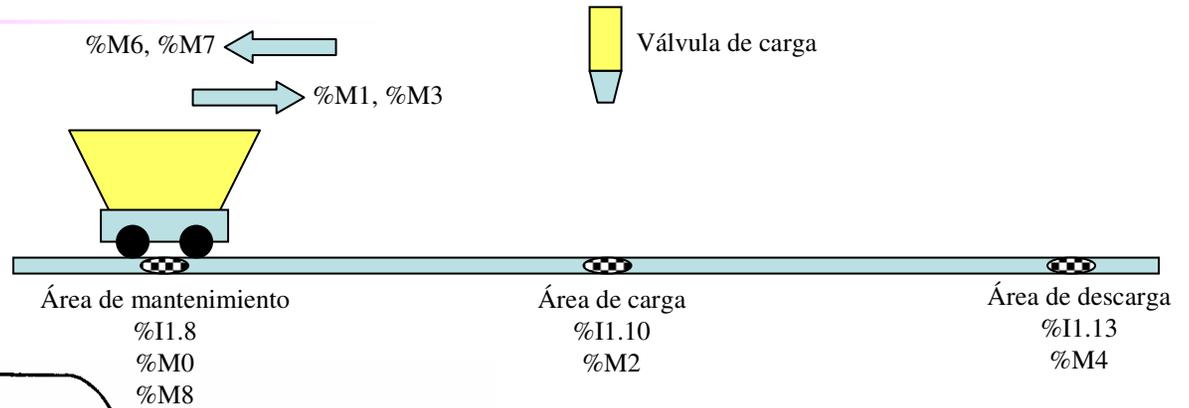
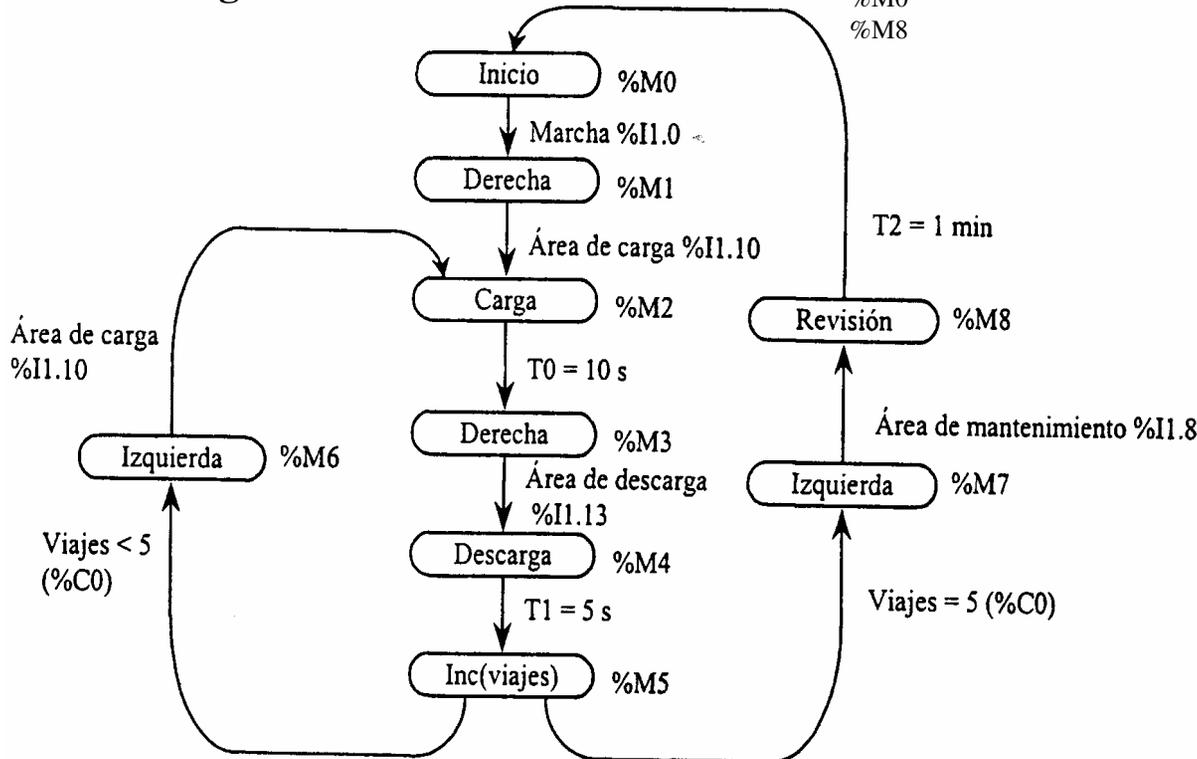


Diagrama de Estados



Variables

- %M0 Estado inicial
- %M1 Mover derecha hacia la zona de carga
- %M2 Cargando
- %M3 Mover derecha hacia zona de descarga
- %M4 Descargando
- %M5 Contar viajes
- %M6 Mover izquierda hacia zona de carga
- %M7 Mover izquierda hacia zona de revision
- %M8 Revisión
- %I1.0 Pulsador de arranque
- %I1.10 Sensor zona de carga
- %I1.13 Sensor zona de descarga

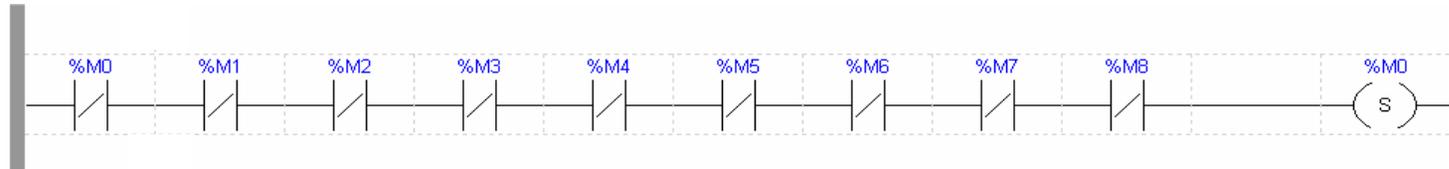


Autómatas Programables

● Programación

1. Inicialización

▶ Diagrama de contactos

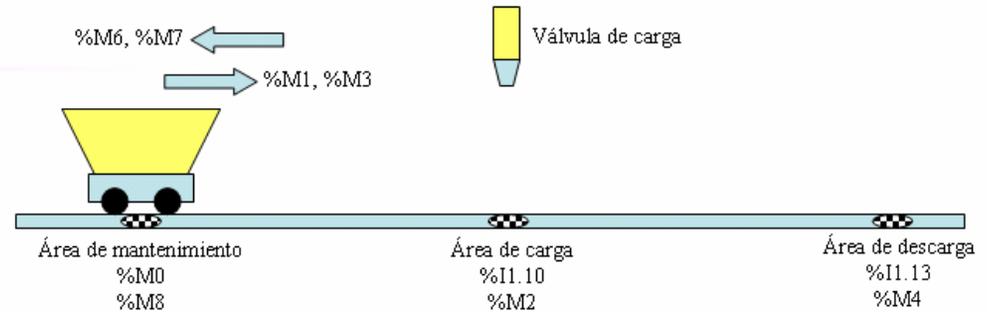


▶ Lenguaje Literal Estructurado

```
! (* Inicialización *)
IF NOT %M0 AND NOT %M1 AND NOT %M2
AND NOT %M3 AND NOT %M4 AND NOT %M5
AND NOT %M6 AND NOT %M7 AND NOT %M8
THEN
    SET %M0;
END_IF;
```

▶ Lista de Instrucciones

```
LDN %M0
ANDN %M1
ANDN %M2
ANDN %M3
ANDN %M4
ANDN %M5
ANDN %M6
ANDN %M7
ANDN %M8
S %M0;
```

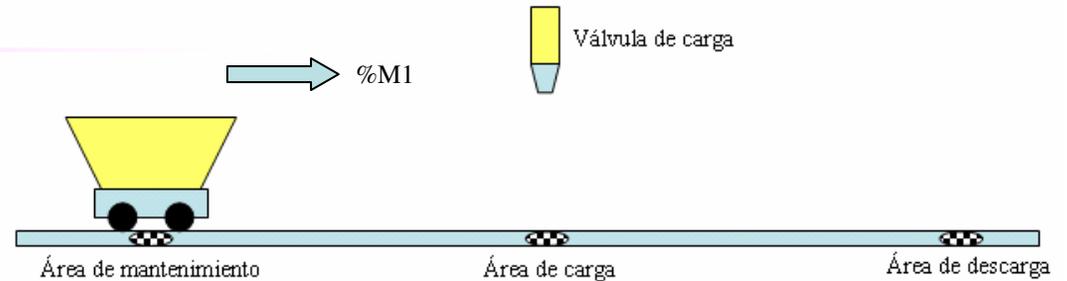




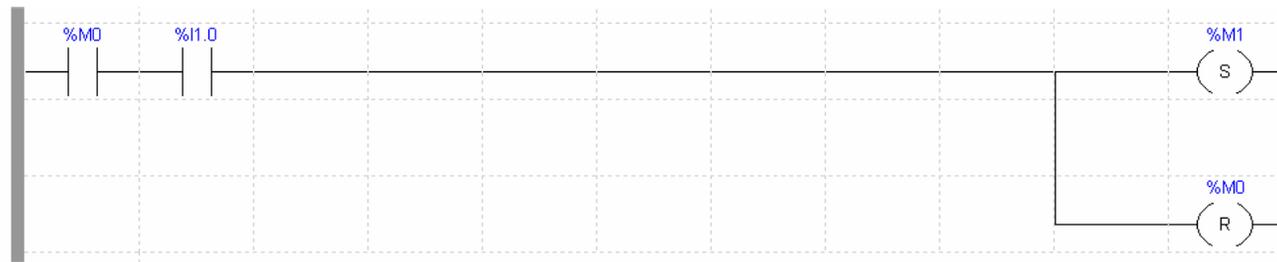
Autómatas Programables

● Programación

2. Puesta en marcha



▶ Diagrama de contactos



▶ Lenguaje Literal Estructurado

```
! (* Puesta en marcha *)
IF %M0 AND %I1.0 THEN
    SET %M1;
    RESET %M0;
END_IF;
```

▶ Lista de Instrucciones

```
LD %M0
AND %I1.0
S %M1
R %M0
```

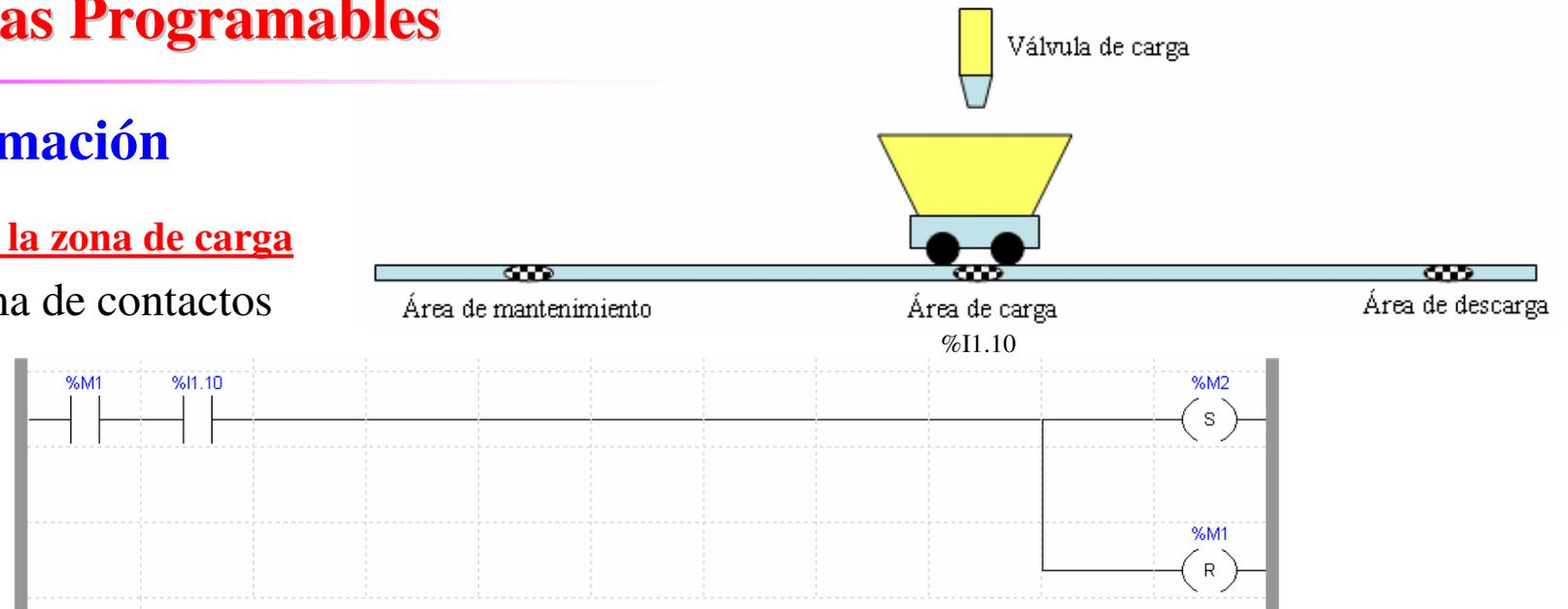


Autómatas Programables

Programación

3. Llegar a la zona de carga

► Diagrama de contactos



► Lenguaje Literal Estructurado

```
! (* Llegando a la zona de carga *)  
IF %M1 AND %I1.10 THEN  
    SET %M2;  
    RESET %M1;  
END_IF;
```

► Lista de Instrucciones

```
LD %M1  
AND %I1.10  
S %M2  
R %M1
```

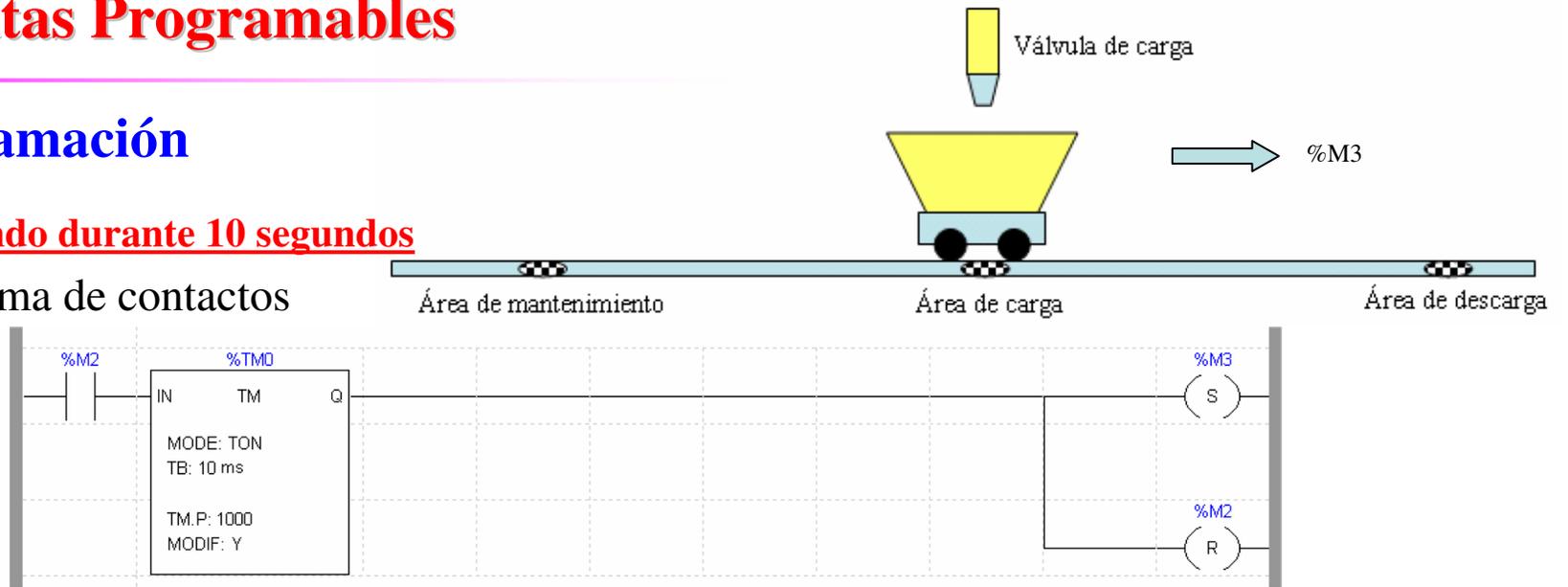


Autómatas Programables

● Programación

3. Cargando durante 10 segundos

▶ Diagrama de contactos



▶ Lenguaje Literal Estructurado

```
! (* Cargando durante 10 segundos. *)  
IF RE %M2 THEN  
    START %TM0;  
ELSEIF FE %M2 THEN  
    DOWN %TM0;  
END_IF;  
IF %M2 AND %TM0.Q THEN  
    SET %M3;  
    RESET %M2;  
END_IF;
```

▶ Lista de Instrucciones

```
LD %M2  
IN %TM0  
LD %TM0.Q  
S %M3  
R %M2
```



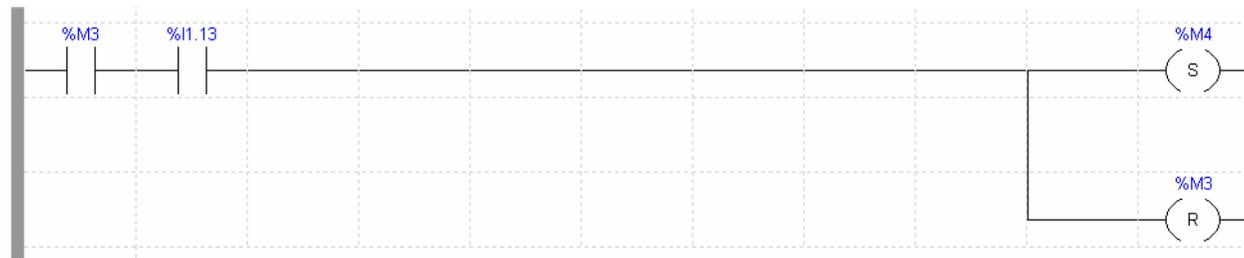
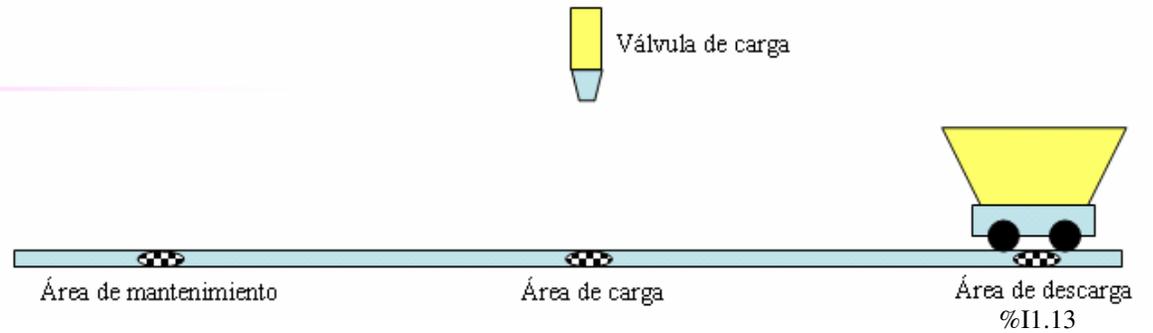
Autómatas Programables

Programación



4. Llegando a la zona de descarga

▶ Diagrama de contactos



▶ Lenguaje Literal Estructurado

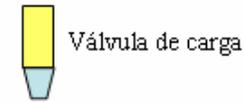
```
! (* Llegando a la zona de descarga *)
IF %M3 AND %I1.13 THEN
    SET %M4;
    RESET %M3;
END_IF;
```

▶ Lista de Instrucciones

```
LD %M3
AND %I.13
S %M4
R %M3
```



Autómatas Programables

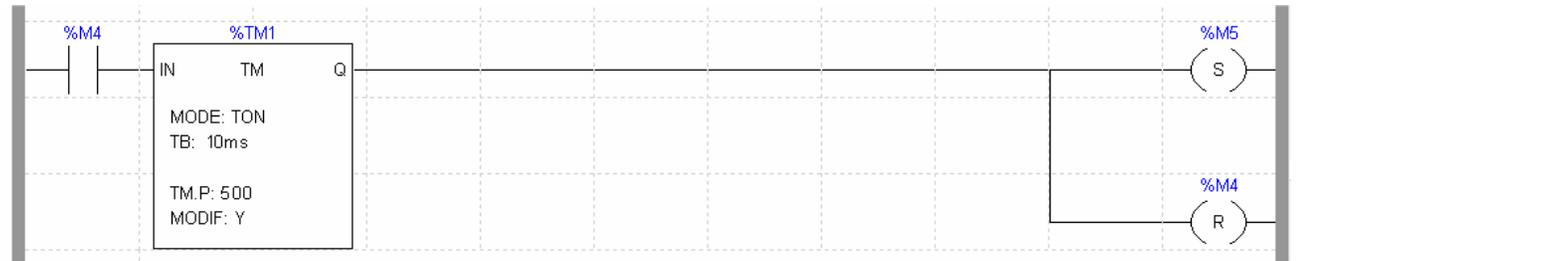


Programación



5. Descargando durante 5 segundos

► Diagrama de contactos



► Lenguaje Literal Estructurado

```
! (* Descargando durante 5 segundos *)
IF RE %M4 THEN
    START %TM1;
ELSEIF FE %M4 THEN
    DOWN %TM1;
END_IF;
IF %M4 AND %TM1.Q THEN
    SET %M5;
    RESET %M4;
END_IF;
```

► Lista de Instrucciones

```
LD %M4
IN %TM1
LD %TM1.Q
S %M5
R %M4
```

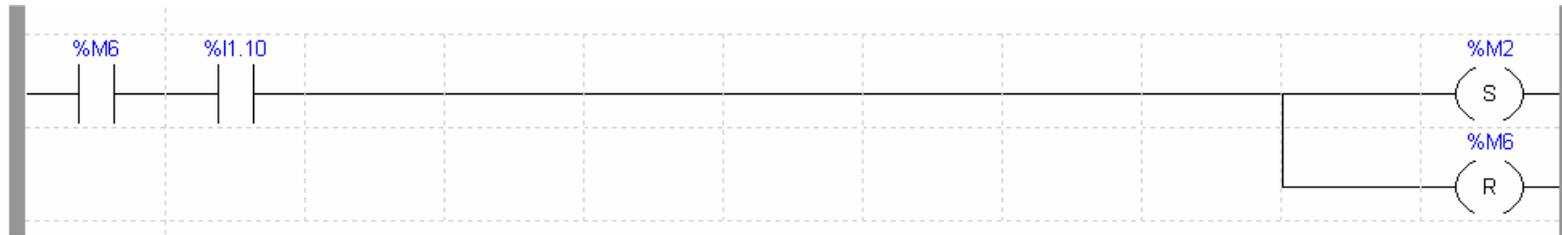
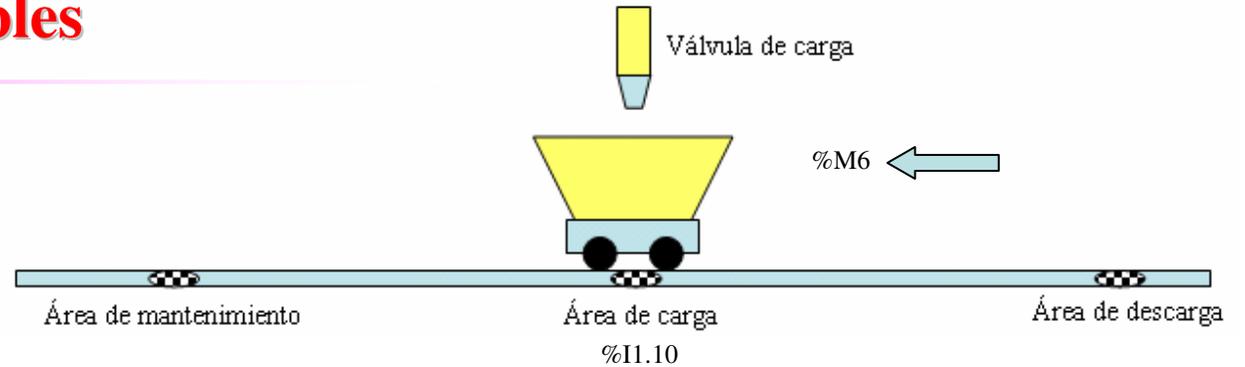


Autómatas Programables

● Programación

6. Llegar a la zona de carga

▶ Diagrama de contactos



▶ Lenguaje Literal Estructurado

```
! (* Llegada a la zona de carga *)  
IF %M6 AND %I1.10 THEN  
    SET %M2;  
    RESET %M6;  
END_IF;
```

▶ Lista de Instrucciones

```
LD %M6  
AND %I1.10  
S %M2  
R %M6
```

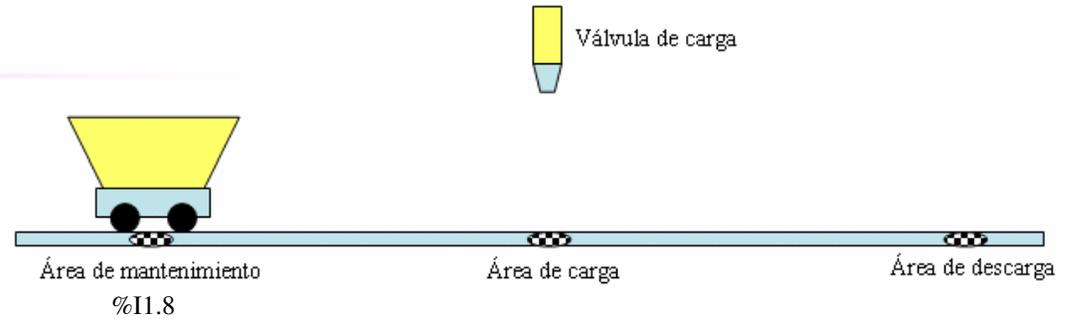


Autómatas Programables

Programación

7. Llegada a la zona de mantenimiento

▶ Diagrama de contactos



▶ Lenguaje Literal Estructurado

```
! (* Llegada a la zona de mantenimiento *)
IF %M7 AND %I1.8 THEN
    SET %M8;
    RESET %M7;
END_IF;
```

▶ Lista de Instrucciones

```
LD %M7
AND %I1.8
S %M8
R %M7
```

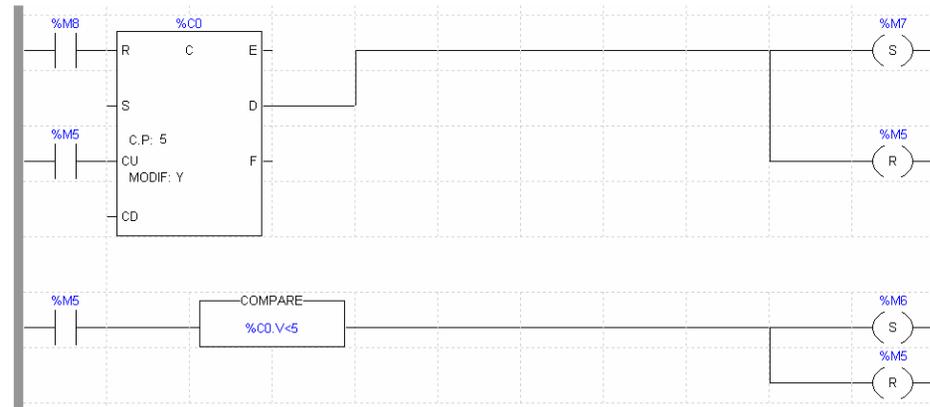


Autómatas Programables

Programación

8. Control del número de viajes

► Diagrama de contactos



► Lenguaje Literal Estructurado

```

IF %M8 THEN
    RESET %C0;
END_IF;
IF RE %M5 THEN
    UP %C0;
END_IF;
IF %M5 AND %C0.D THEN
    SET %M7;
    RESET %M5;
END_IF;
IF %M5 AND (%C0.V<5) THEN
    SET %M6;
    RESET %M5;
END_IF;

```

► Lista de Instrucciones

```

LD %M8
R %C0
LD %M5
CU %C0
LD %C0.D
S %M7
R %M5

LD %M5
AND [%C0.V<5]
S %M6
R %M5

```

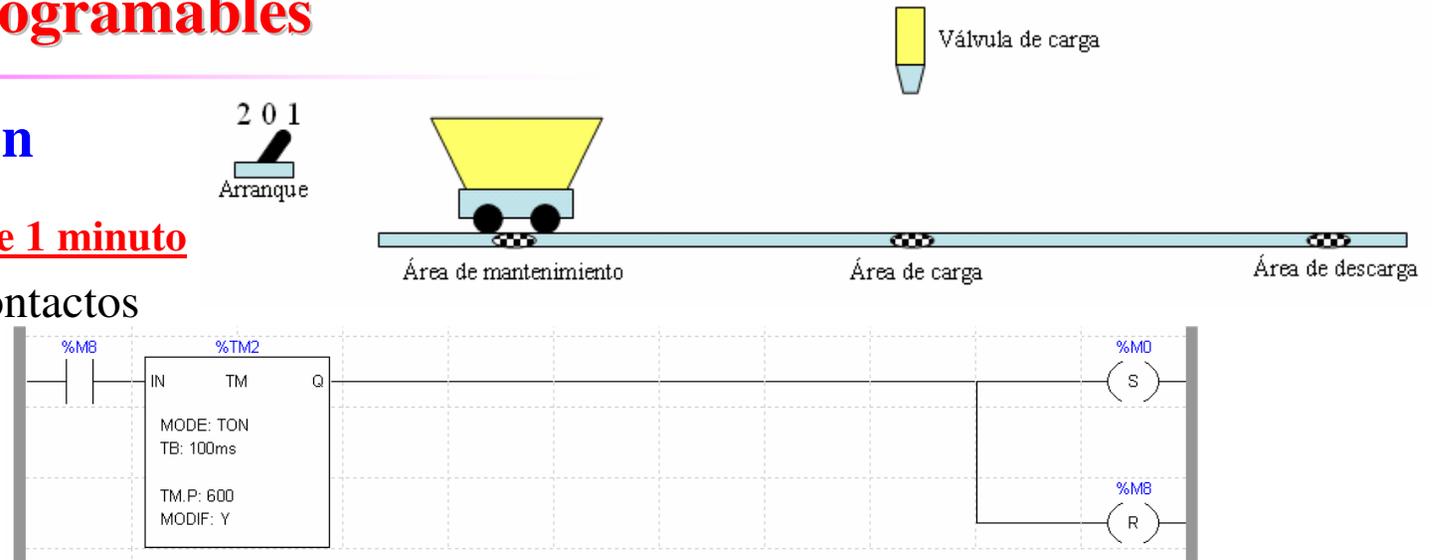


Autómatas Programables

Programación

9. Revisión durante 1 minuto

► Diagrama de contactos



► Lenguaje Literal Estructurado

```
! (* Revisión durante 1 minuto *)
IF RE %M8 THEN
    START %TM2;
ELSEIF FE %M8 THEN
    DOWN %TM2;
END_IF
IF %M8 AND %TM2.Q THEN
    SET %M0;
    RESET %M8;
END_IF;
```

► Lista de Instrucciones

```
LD %M8
IN %TM2
LD %TM2.Q
S %M0
R %M8
```

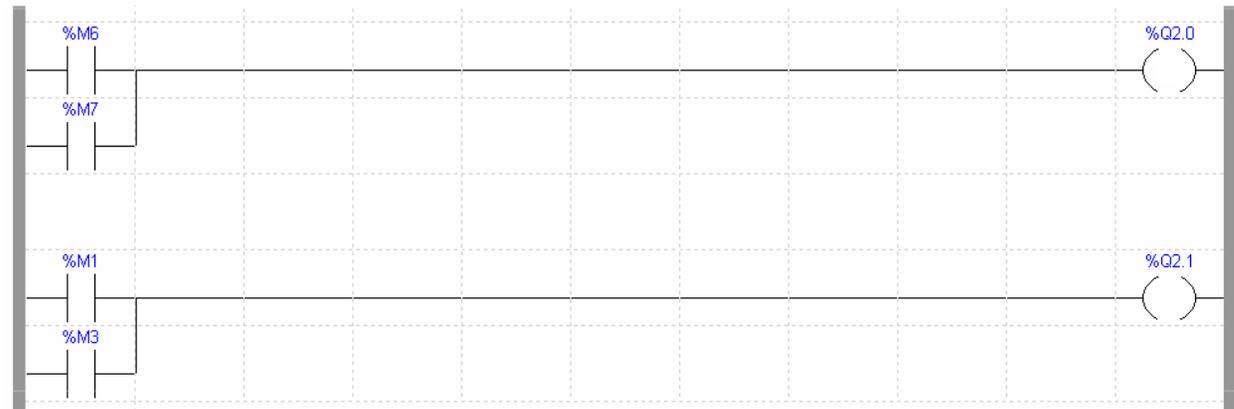


Autómatas Programables

● Programación

Activación de los motores

▶ Diagrama de contactos



▶ Lenguaje Literal Estructurado

```
! (* Activación de los motores *)  
%Q2.0 := %M6 OR %M7 ;  
%Q2.1 := %M1 OR %M3 ;
```

Variables

%Q2.0	Motor izquierda
%Q2.1	Motor derecha

▶ Lista de Instrucciones

```
LD %M6  
OR %M7  
ST %Q2.0  
  
LD %M1  
OR %M3  
ST %Q2.1
```

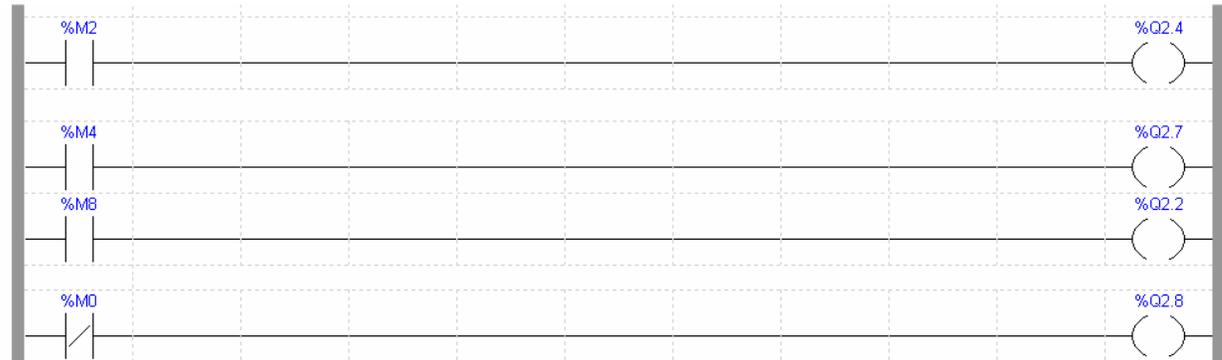


Autómatas Programables

● Programación

Activación de los LEDs

▶ Diagrama de contactos



▶ Lenguaje Literal Estructurado

! (* Activación de los LEDs *)

%Q2.4:=%M2;

%Q2.7:=%M4;

%Q2.2:=%M8;

%Q2.8:=NOT %M0;

Variables

%Q2.4	LED cargando
%Q2.7	LED descargando
%Q2.2	LED revisión
%Q2.8	Indicador de actividad

▶ Lista de Instrucciones

```
LD %M2  
ST %Q2.4
```

```
LD %M4  
ST %Q2.7
```

```
LD %M8  
ST %Q2.2
```

```
LDN %M0  
ST %Q2.8
```



Resumen:

- Un autómeta programable industrial (**API**) o Programmable Logic Controller (**PLC**), es un equipo electrónico, programable en lenguaje no informático, diseñado para controlar en tiempo real y en ambiente de tipo industrial, procesos secuenciales.
 - Un **PLC** trabaja en base a la información recibida por los captadores y el programa lógico interno, actuando sobre los accionadores de la instalación .
 - Los **lenguajes gráficos y textuales** definidos en el estándar son una fuerte base para entornos de programación potentes en PLC's. Con la idea de hacer el estándar adecuado para un gran abanico de aplicaciones, cinco lenguajes han sido definidos en total: Gráfico secuencial de funciones (grafcet), Lista de instrucciones (LDI o AWL), Texto estructurado, Diagrama de contactos y diagrama de bloques funcional(FBD).
 - **Diagrama de contactos:** es un lenguaje que utiliza un juego estandarizado de símbolos de programación
 - **Texto estructurado:** es un lenguaje de alto nivel estructurado por bloques que posee una sintaxis parecida al PASCAL
 - **Lista de instrucciones:** es un lenguaje de bajo nivel, similar al lenguaje ensamblador. Este lenguaje es adecuado para pequeñas aplicaciones y para optimizar partes de una aplicación
-



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 9 – Miércoles 17 de Mayo 2006

Autómatas Programables



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 10 – Jueves 18 de Mayo 2006

Supervisory Control And Data Acquisition

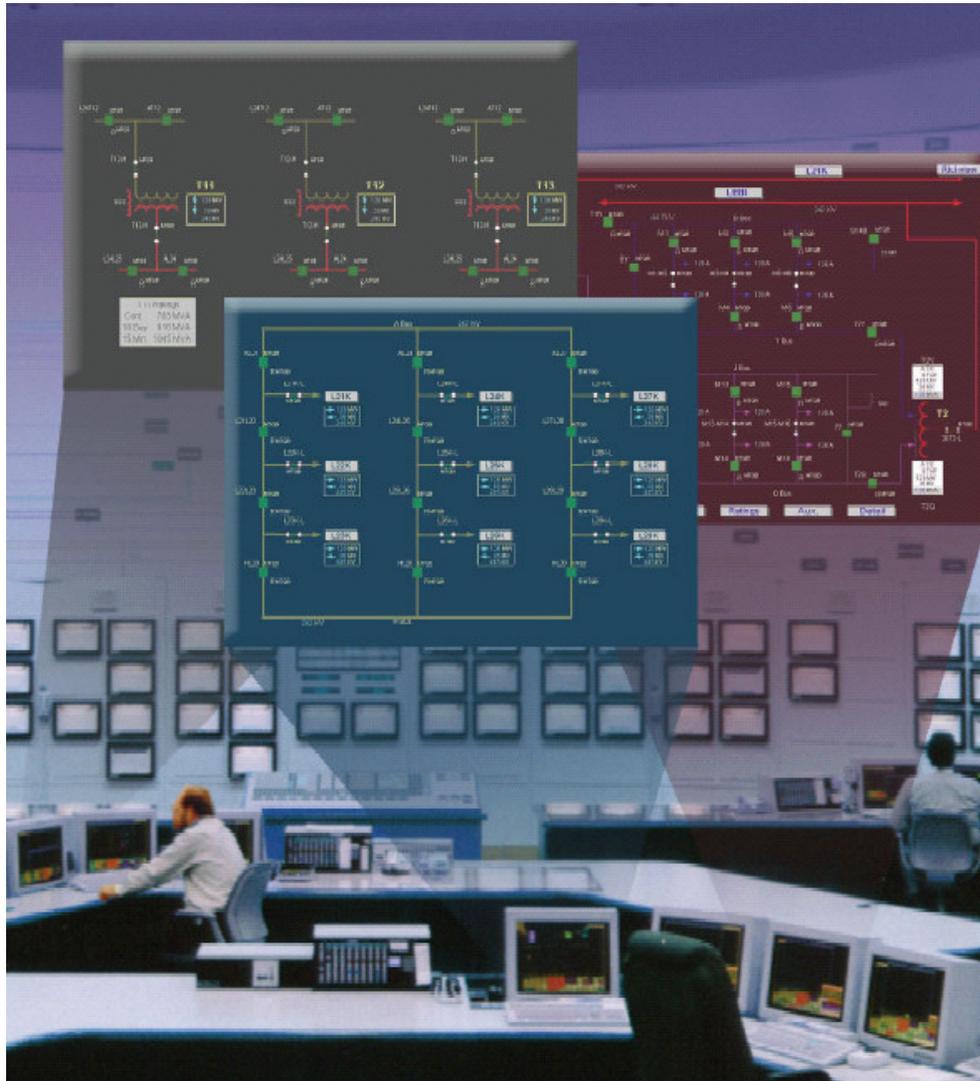
SCADA

(Adquisición de Datos y Control Supervisor)



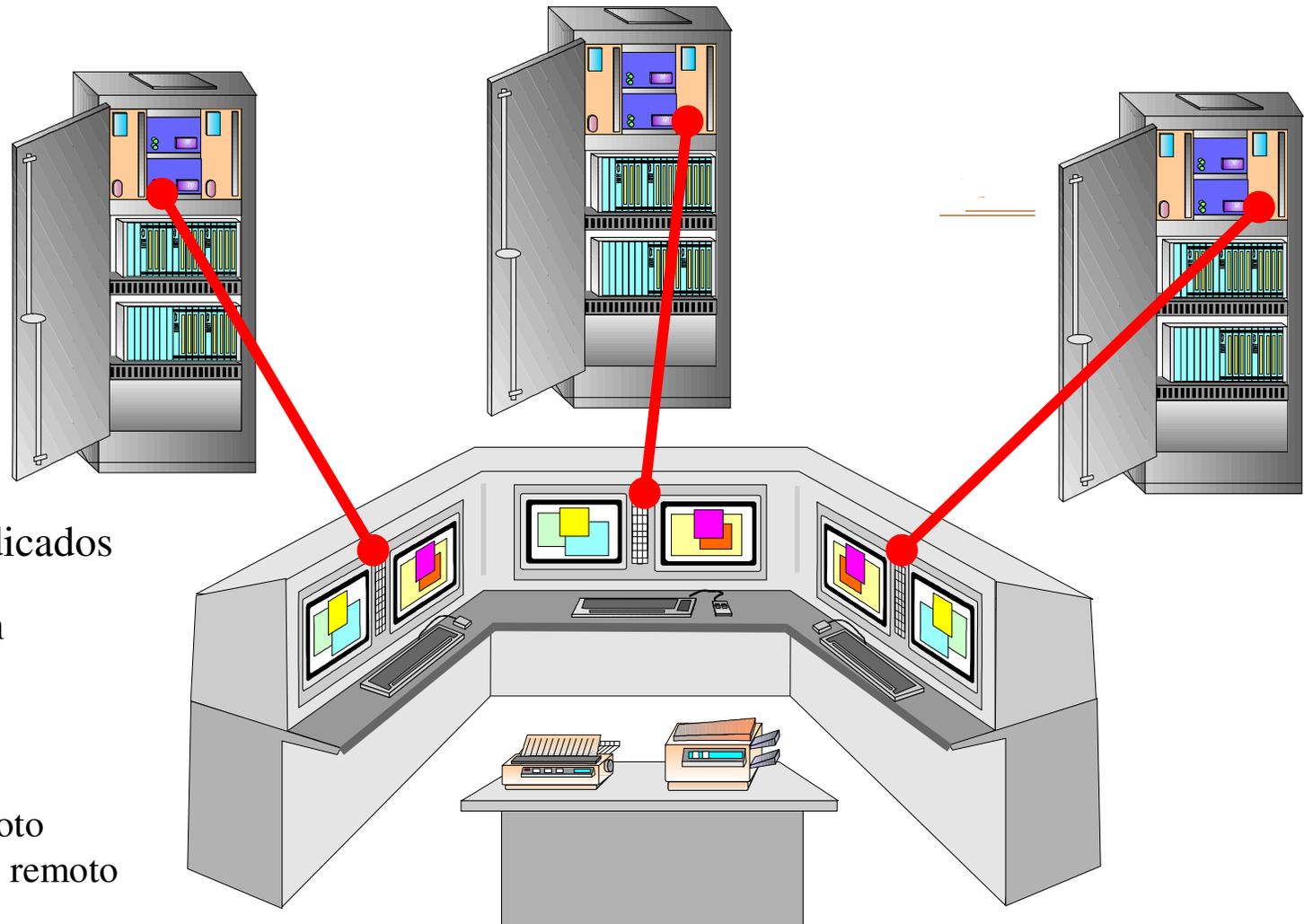
Automatización Industrial -II

Sistemas de control industrial





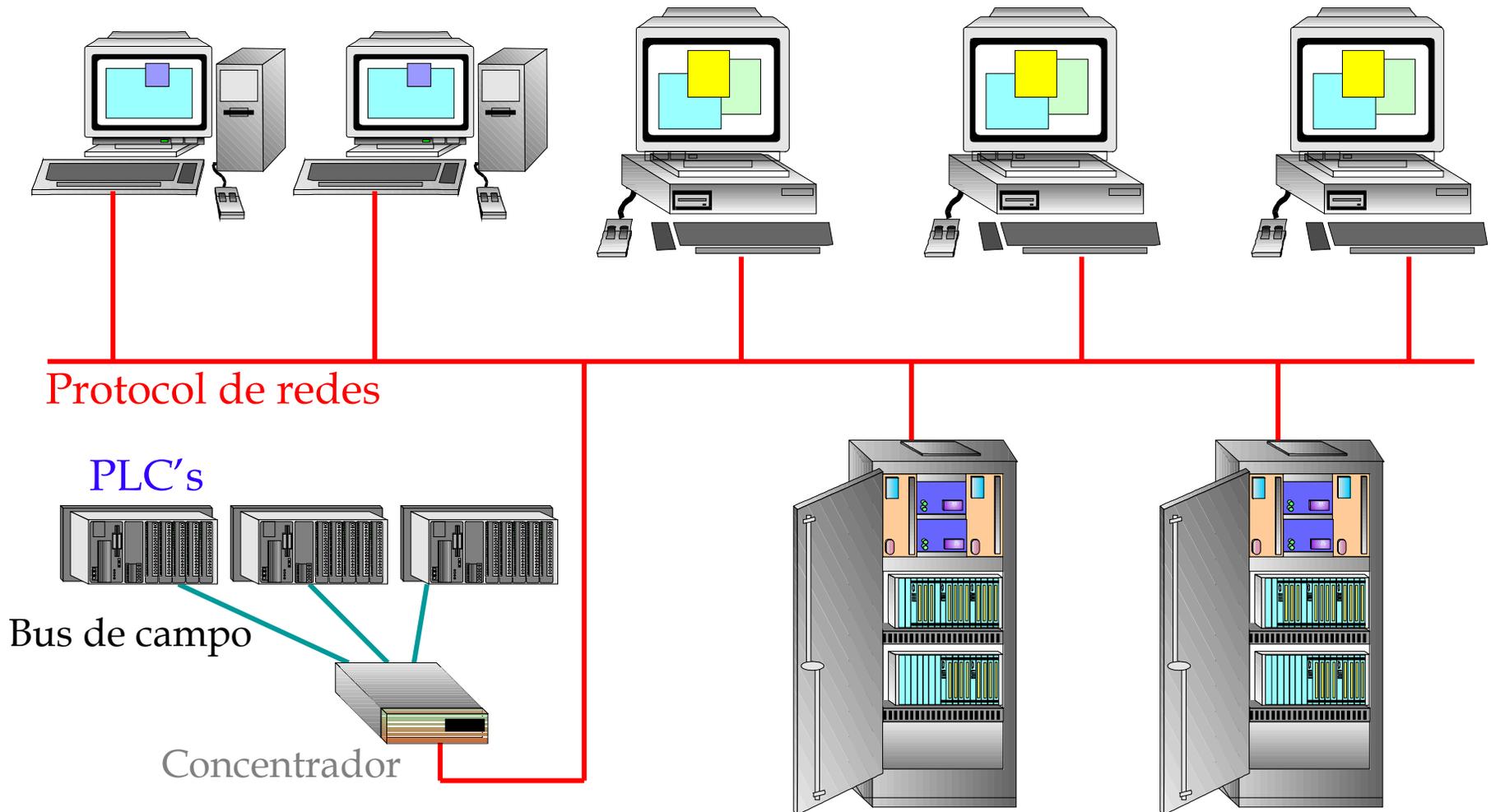
Sistemas de control tradicional



- ▶ Monitores dedicados
- ▶ Comunicación punto a punto.
- ▶ No red.
 - No acceso remoto
 - No diagnóstico remoto



Sistemas de control distribuido (DCS)





Sistemas de control distribuido (DCS)

● **Ventajas**

- ▶ Bases de datos/programas distribuidos
- ▶ Acceso distribuido
- ▶ Diagnostico distribuido
- ▶ Monitorización de cualquier variable desde cualquier lugar ‘**everything everywhere**’

● **Desventajas**

- ▶ Compatibilidad entre los sistemas de control distribuido.

● **Dificultad**

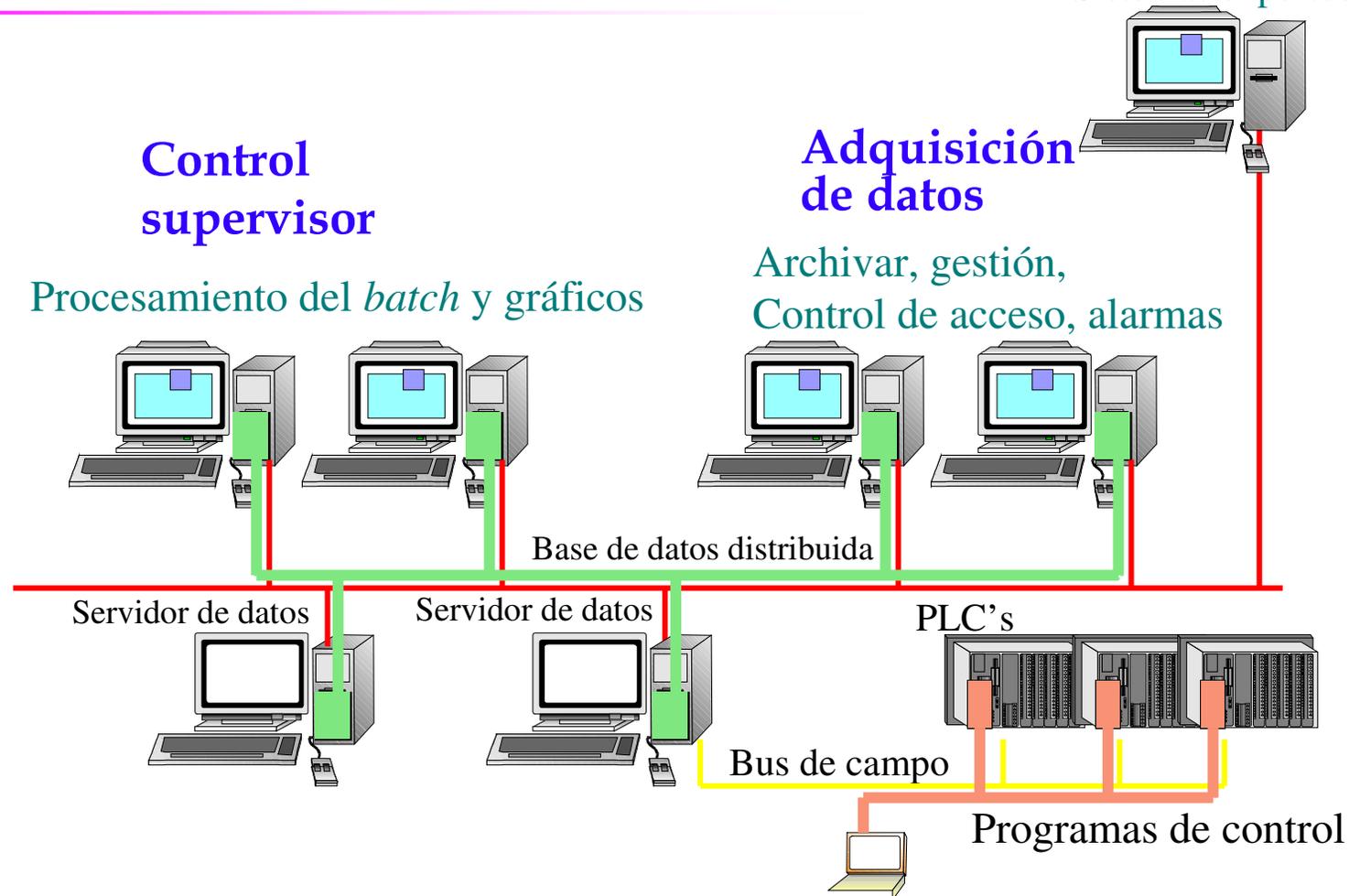
- ▶ La integración de varios componentes de buses de campo.



Sistemas SCADA

Sistemas de planificación de recursos ERP

Sistemas expertos

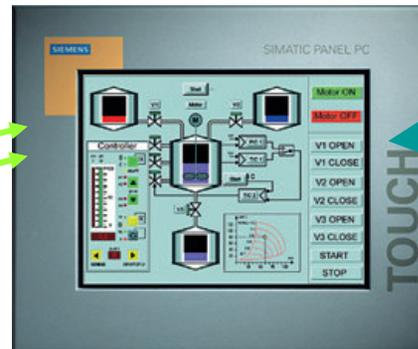


Los sistemas SCADA abarcan desde una pequeña estación de trabajo con un PC, hasta el complejo sistema que puede requerir una central nuclear con varios monitores, alarmas sonoras e incluso avisos vía SMS



Sistemas SCADA

Un **SCADA** es un sistema basado en computadores que permite supervisar y controlar a distancia una instalación de cualquier tipo.



- ▶ Control, supervisión y adquisición de datos.
- ▶ Paquete software enfocado a la monitorización y el mando de procesos industriales.
- ▶ Sirve de Interfaz entre el operador y el proceso.

El labor principal de los sistemas SCADA es la supervisión



Automatización Industrial -II

¿Para qué sirve el SCADA?

- ▶ El **SCADA** permite a un operador hacer cambios en el punto de operación (*Set Point*) sobre controladores de procesos distantes.
- ▶ Abrir o cerrar válvulas o switches.
- ▶ Monitorizar alarmas.
- ▶ Recoger desde una localidad central **información de mediciones** de procesos extensamente distribuidos, tales como, campos de gas o petróleo, sistemas de tuberías de transporte de líquidos o complejos de generación hidroeléctrica, etc.
- ▶ En procesos donde las dimensiones suelen ser de cientos o miles de kilómetros se apreciar el enorme beneficio que ofrecen los **SCADA**, en la **reducción de visitas** para monitorear los procesos.





Aplicaciones de los sistemas SCADA

▶ Grupos de pequeñas estaciones generadoras hidroeléctricas que son encendidas o apagadas en respuesta a las demandas de los clientes y que usualmente están localizadas en lugares remotos; estos pueden ser controlados mediante la apertura y cierre de válvulas de una turbina, pueden ser monitoreados continuamente y deben responder relativamente rápido ante la demanda de la red de energía eléctrica.

▶ Producción de gas o crudo suelen incluir pozos, mediciones de flujo, bombas; que suelen estar distribuidas sobre grandes áreas y requieren control simple como encendido/apagado de motores, recogida de datos regularmente y deben responder rápidamente a las condiciones del campo de producción.





Aplicaciones de los sistemas SCADA

► Redes de tuberías de gas, petróleo, químicos o agua; cuentan con elementos localizados a distancias variables de un punto de control central. Pueden ser controlados por la apertura/cierre de válvulas, encendido/apagado de bombas, monitoreados y deben responder rápidamente ante cambios en las condiciones del mercado y derrames de materiales peligrosos o daños al medio ambiente.



► Sistemas de transmisión eléctrica pueden cubrir miles de kilómetros cuadrados pueden ser controlados mediante la apertura/cierre de switches y deben responder casi inmediatamente ante cambios en la carga sobre las líneas de transmisión.

► Sistemas de riego que pueden cubrir cientos de kilómetros cuadrados pueden ser controlados por simple apertura/cierre de válvulas y requieren la recogida de mediciones de flujos de aguas, humedad etc.



Sectores de aplicación de los sistemas SCADA

- Producción automovilística y sus proveedores
- Industria química y farmacéutica
- Generación y distribución de energía
- Industria del plástico y caucho
- Metalurgia
- Siderurgia
- Industria alimenticia y de bebidas
- Tratamiento y depuración de aguas
- Industria de artes gráficas
- Comercio y sector servicios
- Construcción de maquinaria e instalaciones
- Fabricación y transformación de papel
- Transporte y tráfico
- ...





Sistemas SCADA y Sistemas de control distribuido

A diferencia de los Sistemas de Control Distribuido (**DCS**), el lazo de control es **generalmente** cerrado por el operador. Los **DCS** se caracterizan por realizar las acciones de control en forma automática. Hoy en día es fácil hallar un sistema **SCADA** realizando labores de control automático en cualquiera de sus niveles, aunque su labor principal sea de supervisión y control por parte del operador.

ASPECTO	SCADAs	DCS
TIPO DE ARQUITECTURA	CENTRALIZADA	DISTRIBUIDA
TIPO DE CONTROL PREDOMINANTE	SUPERVISORIO: Lazos de control cerrados por el operador. Adicionalmente: control secuencial y regulatorio.	REGULATORIO: Lazos de control cerrados automáticamente por el sistema. Adicionalmente: control secuencial, batch, algoritmos avanzados, etc.
TIPOS DE VARIABLES	DESACOPLADAS	ACOPLADAS
ÁREA DE ACCIÓN	Áreas geográficamente distribuidas.	Área de la planta.
UNIDADES DE ADQUISICIÓN DE DATOS Y CONTROL	Remotas, PLCs.	Controladores de lazo, PLCs.
MEDIOS DE COMUNICACIÓN	Radio, satélite, líneas telefónicas, conexión directa, LAN, WAN.	Redes de área local, conexión directa.
BASE DE DATOS	CENTRALIZADA	DISTRIBUIDA



Requisitos de los sistemas SCADA

Un **SCADA** debe cumplir varios objetivos para que su instalación sea perfectamente aprovechada:

- ▶ Deben ser sistemas de **arquitectura abierta**, capaces de crecer o adaptarse según las necesidades cambiantes de la empresa.
- ▶ Deben comunicarse con **total facilidad y de forma transparente** al usuario con el equipo de planta y con el resto de la empresa (redes locales y de gestión).
- ▶ Deben ser **programas sencillos de instalar**, sin excesivas exigencias de hardware, y fáciles de utilizar, con interfaces amigables con el usuario.

Fuente: <http://www.automatas.org/>



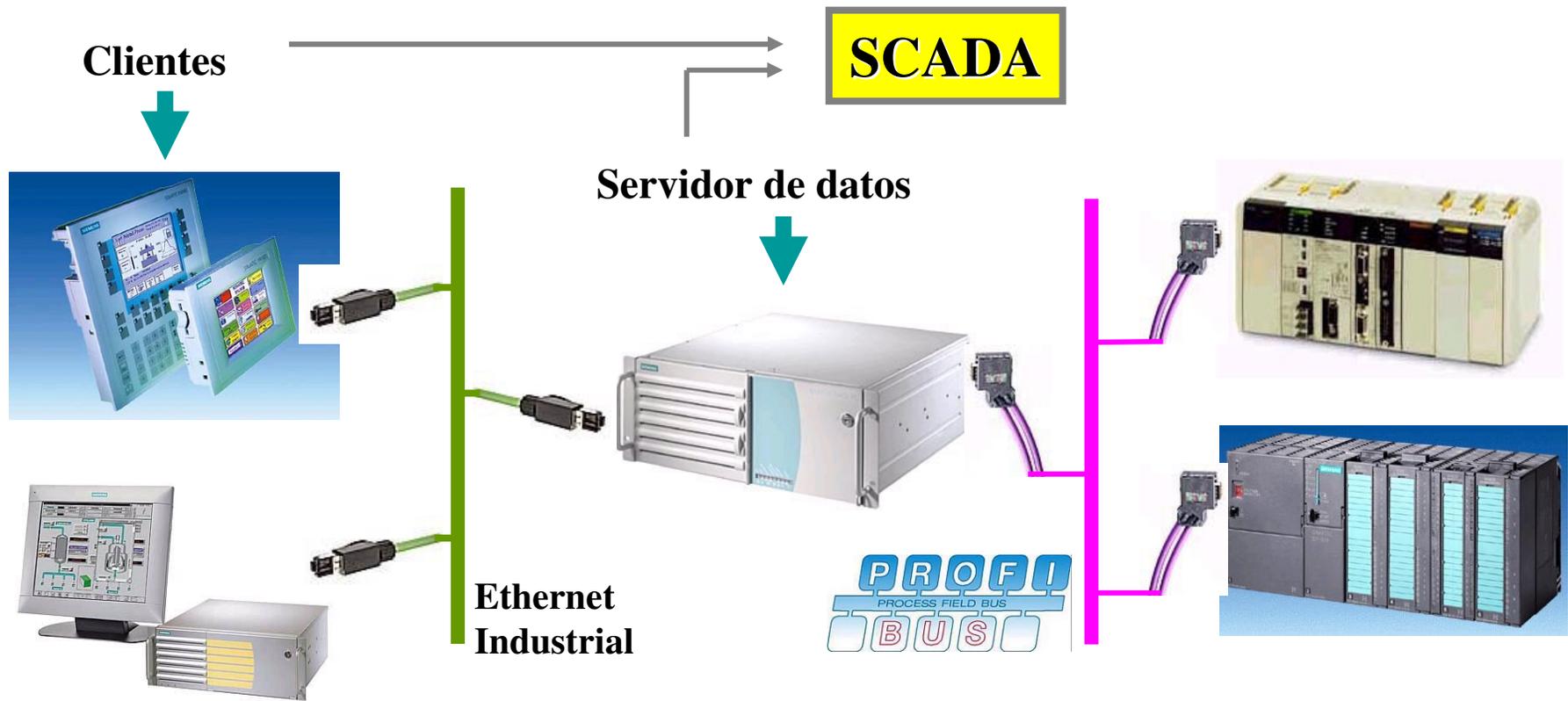
¿Qué nos ofrecen los Sistemas SCADA?

- ▶ Arquitectura (hardware y software) abierta y flexible.
- ▶ Funcionalidades básicas:
 - **Interfaz gráfico del operador:** proporciona al operador las funciones de control y supervisión de la planta. El proceso se representa mediante sinópticos gráficos almacenados en el ordenador de proceso y generados desde el editor incorporado en el SCADA o importados desde otra aplicación durante la configuración del paquete.
 - **Módulo de proceso:** ejecuta las acciones de mando preprogramadas a partir de los valores actuales de variables leídas .
 - **Gestión y archivo de datos:** se encarga del almacenamiento y procesado ordenado de los datos, de forma que otra aplicación o dispositivo pueda tener acceso a ellos.
 - **Comunicaciones:** se encarga de la transferencia de información entre la planta y la arquitectura hardware que soporta el SCADA, y entre ésta y el resto de elementos informáticos de gestión (como sistemas expertos y sistemas de planificación e recursos).
- ▶ Herramientas de desarrollo.



Sistemas SCADA

- Arquitectura Hardware





Sistemas SCADA

● **Arquitectura Hardware: Capa cliente**

- Responsable de la interacción hombre-máquina
- PC, HMI,...



● **Arquitectura Hardware: Capa servidor**

- Control de datos del proceso
- Los datos se obtienen de los dispositivos del proceso, generalmente PLC's a través de buses de campo, LAN o conexión directa

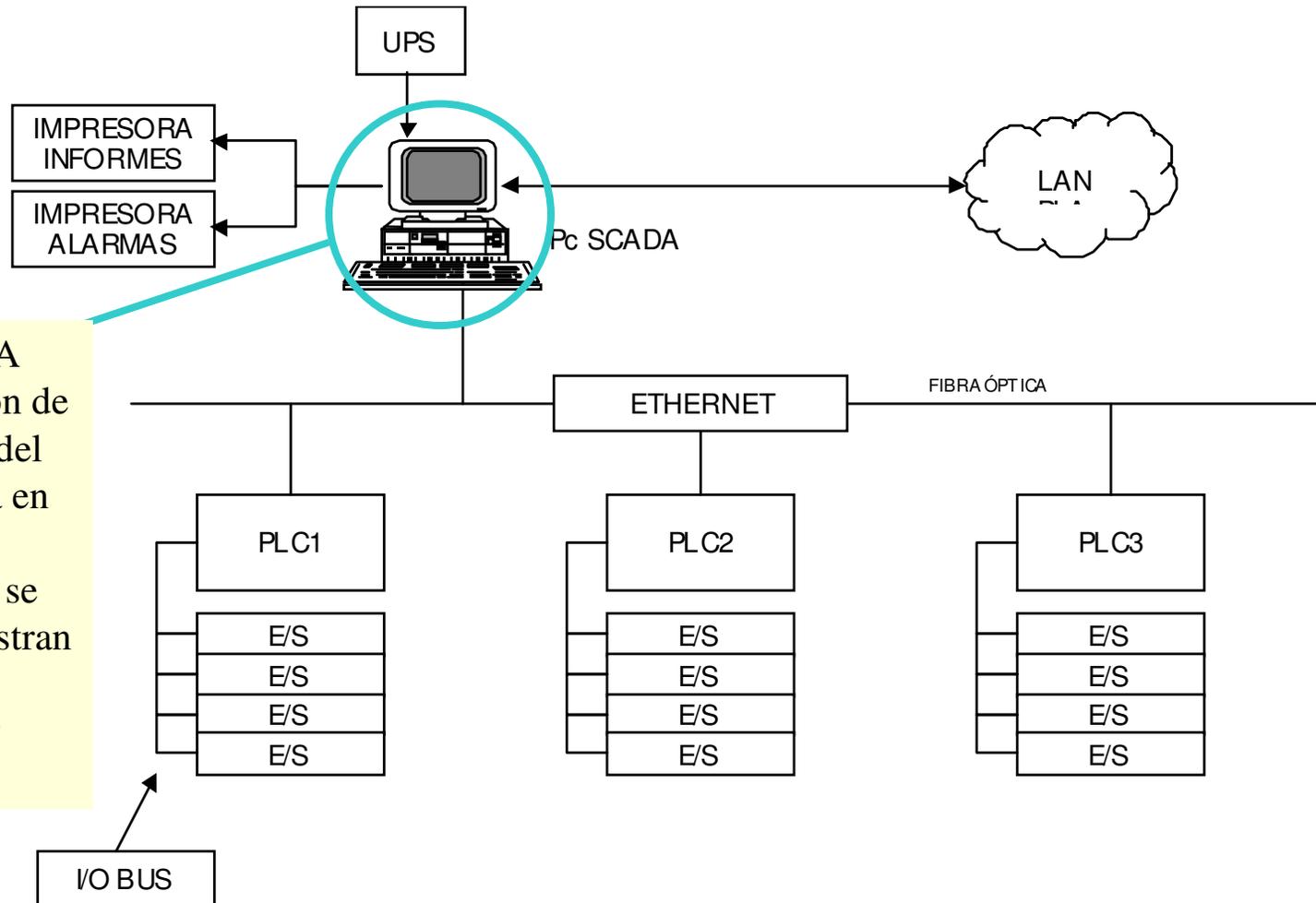


En sistemas sencillos, estas dos capas se encuentran unidas en un único hardware, por ejemplo un PC



Sistemas SCADA

- **Arquitectura Hardware: Distribución típica de un proceso**

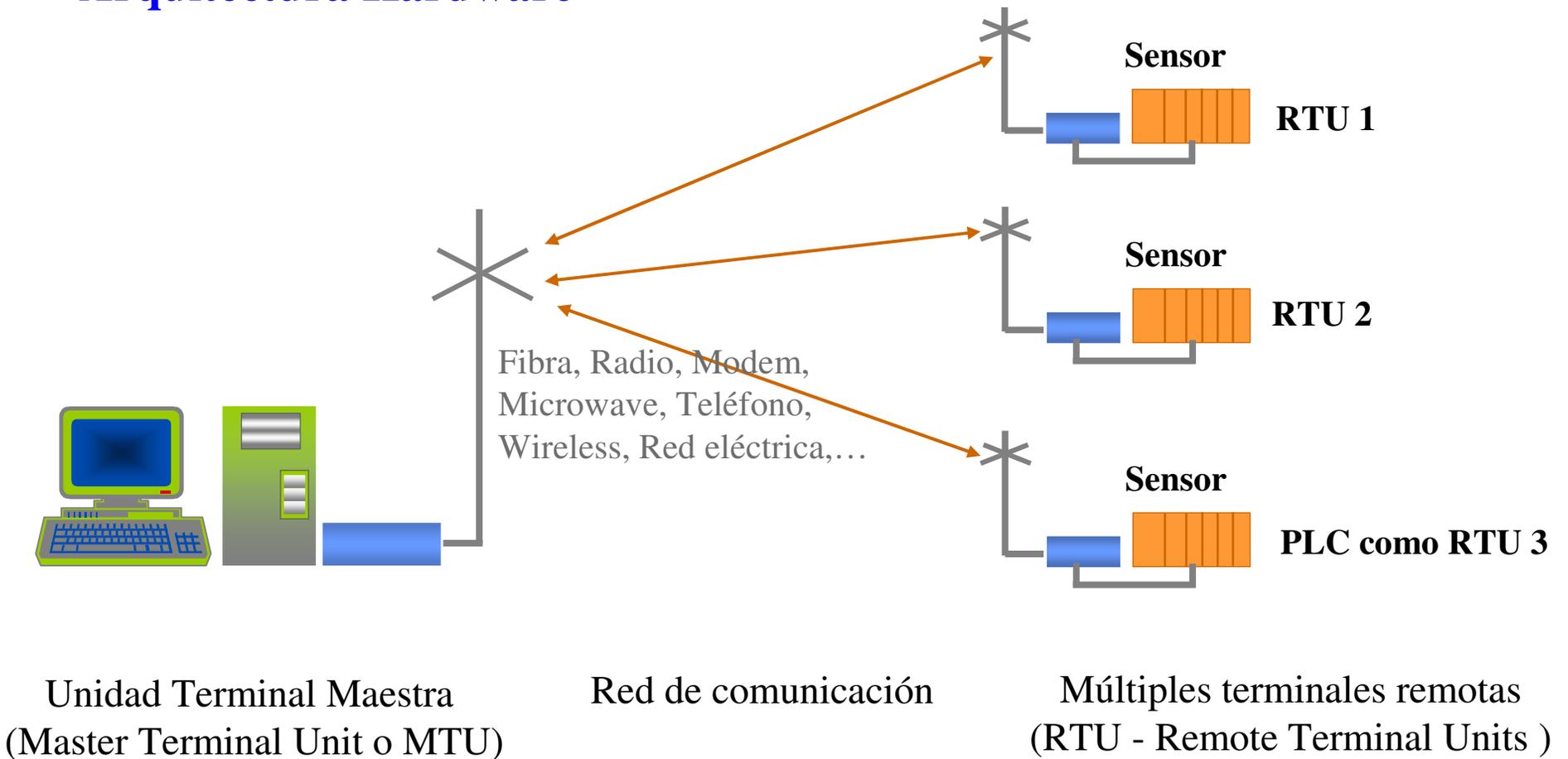


El sistema SCADA recoge información de diferentes puntos del proceso y los trata en una estación de trabajo. Los datos se analizan y se muestran al usuario de una forma coherente y ordenada.



Sistemas SCADA

● Arquitectura Hardware

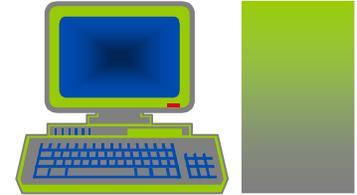




Sistemas SCADA

● **Arquitectura Hardware**

▶ **Unidad Terminal Maestra (Master Terminal Unit o MTU)**



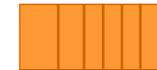
- Conocido como “*host computer*”.
- En un SCADA moderno la MTU es siempre basada en un computador.
- La MTU, puede **monitorizar y controlar el proceso** en ausencia de un operador.
- Esto se realiza por medio de una **esquema o agenda** que puede ser programada para repetir instrucciones periódicamente a intervalos conocidos.
- Por ejemplo, la agenda o esquema puede **solicitar una actualización** desde cada unidad remota (RTU) cada 5 minutos.
- Las MTUs, **deben comunicarse con las RTUs** del proceso y suelen utilizarse medios para este fin como Líneas de tierra (Fibra óptica, Líneas telefónicas, Líneas Coaxial, Red eléctrica), Radio, Telefonía móvil y Satélites.



Sistemas SCADA

● **Arquitectura Hardware**

▶ **Unidad Terminal Remota (Remote Terminal Unit o RTU)**



RTU

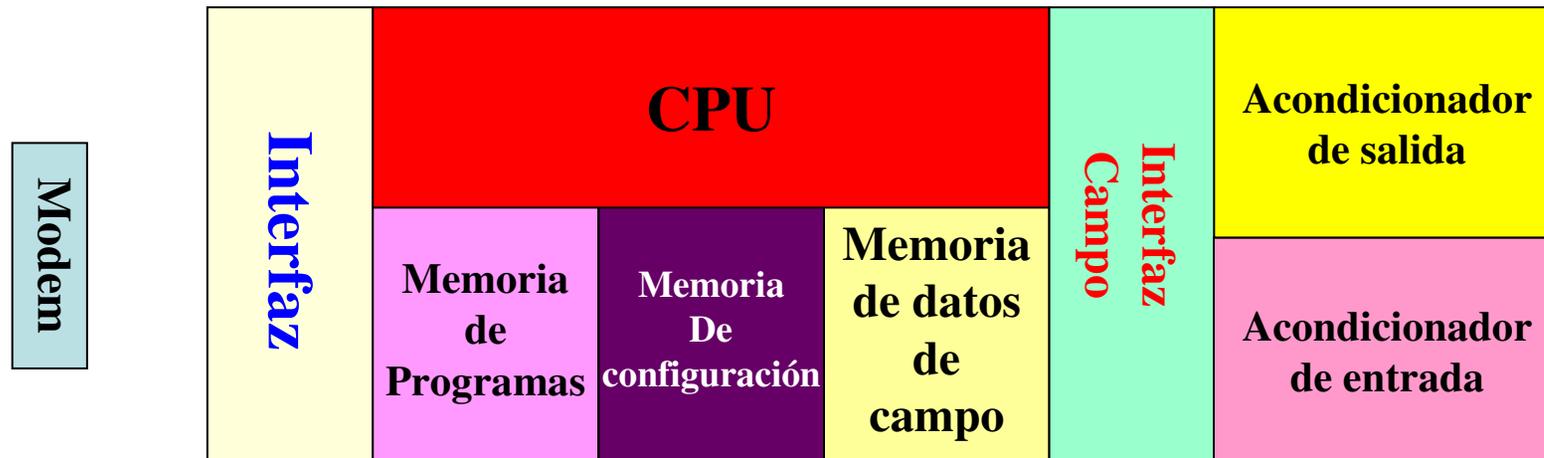
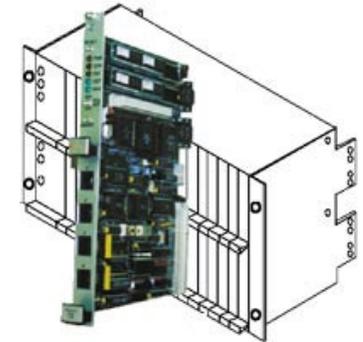
- La RTU se comunica con la MTU por medio de una señal modulada sobre un cable o vía radio.
- Cada RTU debe tener la **capacidad de comprender** cuando un mensaje es dirigido a ella, decodificarlo, actuar en función del mensaje, responder y esperar a un nuevo mensaje.
- Actuar puede ser un **procedimiento complejo**, que puede requerir algún campo de almacenamiento de memoria, comparaciones, envío de señales eléctricas a dispositivos de campo que ordene un cambio de estado, revisión de switches, etc.
- Por su complejidad en su mayoría los RTUs, son basados en ordenadores.



Sistemas SCADA

- Arquitectura Hardware**

- ▶ **Unidad Terminal Remota (Remote Terminal Unit o RTU)**

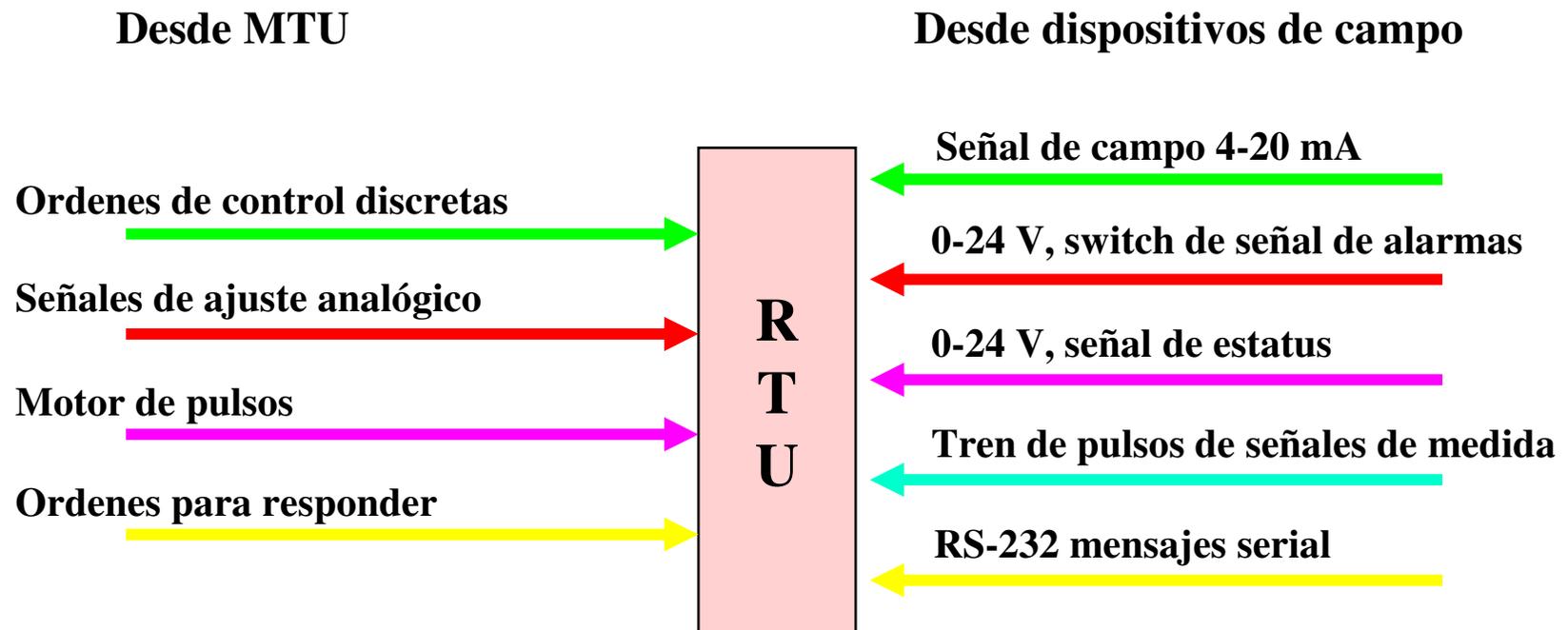




Sistemas SCADA

● Arquitectura Hardware

▶ Unidad Terminal Remota (RTU): Señales de Entrada

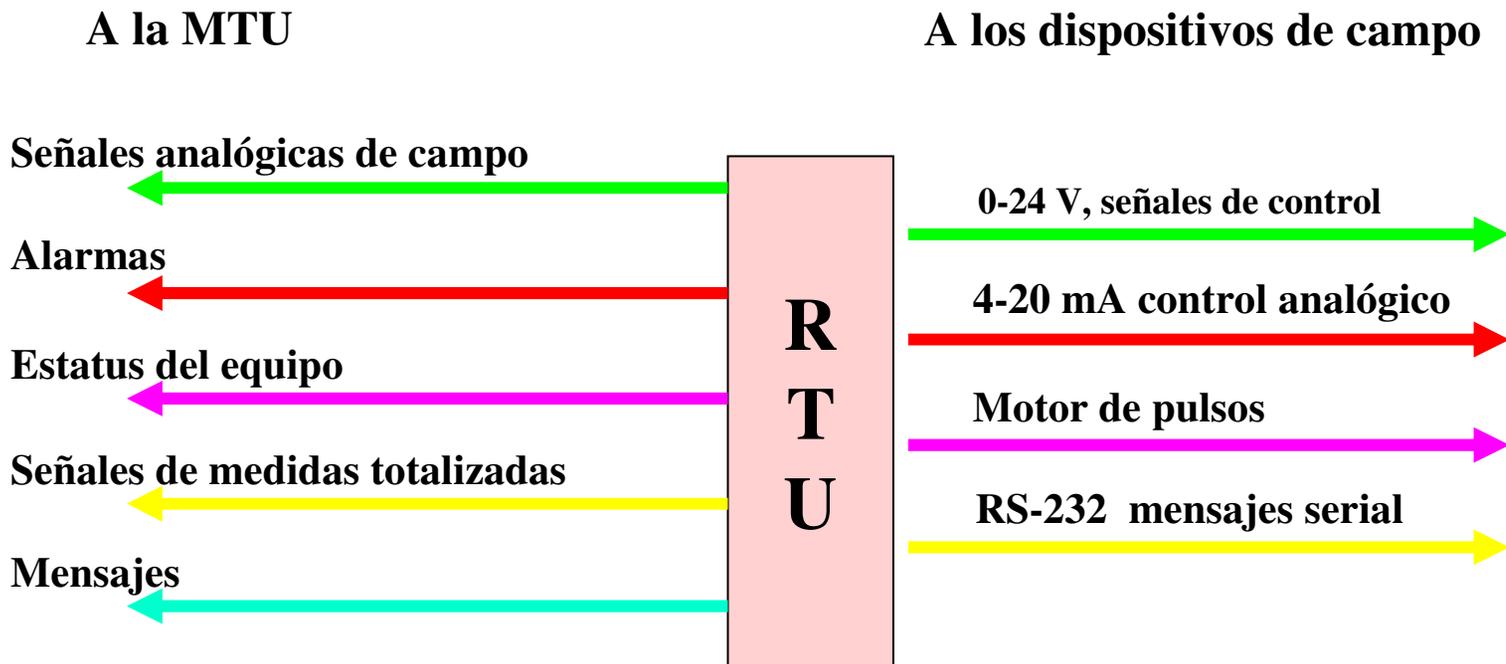




Sistemas SCADA

● Arquitectura Hardware

▶ Unidad Terminal Remota (RTU): Señales de Salida





Sistemas SCADA

- **Arquitectura Hardware**

- ▶ **Comunicaciones**

Dado que un sistema **SCADA** consiste de una o más MTUs enviando datos a, o recibiendo datos, de una o más RTUs, es claro que las comunicaciones juegan un papel fundamental en la operación del sistema.

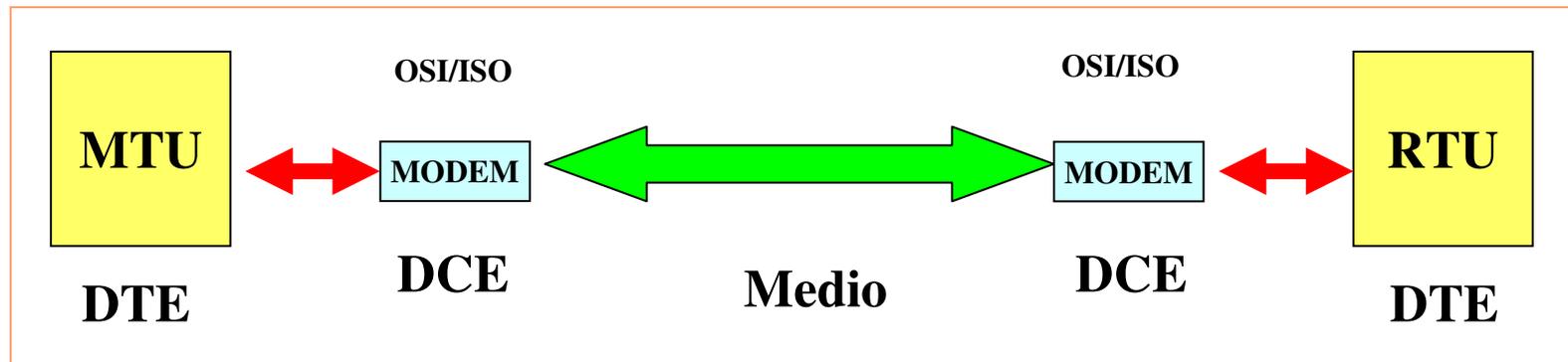
Si un enlace de comunicación **no puede ser establecido**, un sistema SCADA no puede ser desarrollado.



Sistemas SCADA

● Arquitectura Hardware

▶ Comunicaciones: Componentes del sistema de comunicación



- Hablando en términos de telecomunicaciones las MTUs y RTUs, son denominados **Equipos Terminales de Datos (DTEs)**. Los **DTEs** tienen la habilidad de conformar una señal con cierto significado para ser enviada y para decifrar el significado contenido en una señal recibida.
- Otro de los equipos utilizados son los **MODEMs**
- Modulador/Demodulador, conocidos como **Equipos de Comunicación de Datos (DCEs)**. Son capaces de recibir información desde los DTEs, hacer los cambios necesarios de forma y enviarlo sobre un medio a otro DCE, el cuál recibe y transforma para luego enviársela a un DTE

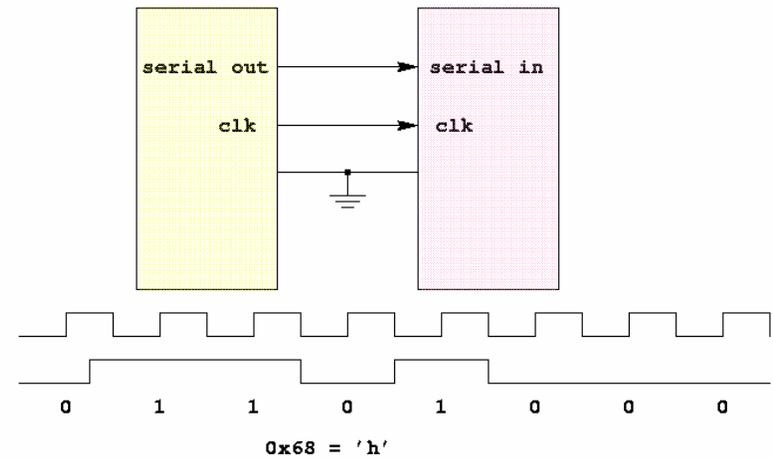


Sistemas SCADA

● Arquitectura Hardware

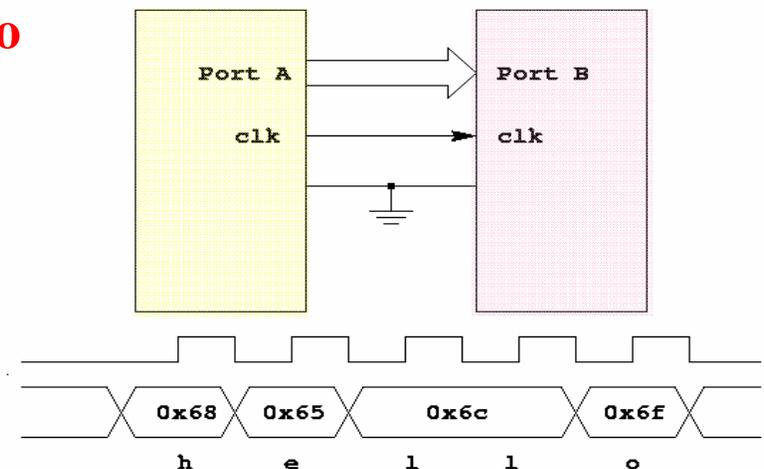
▶ Comunicaciones: **Comunicación Serial**

Todos los datos movidos entre las MTUs y las RTUs son comunicación **serial**, es decir una simple cadena de caracteres binarios son enviadas unas tras otras.



▶ Comunicaciones: **Comunicación en Paralelo**

Es similar a las utilizadas con una impresora, sin embargo, esto implica **más líneas de comunicación** y el costo resulta muy elevado, especialmente en grandes distancias



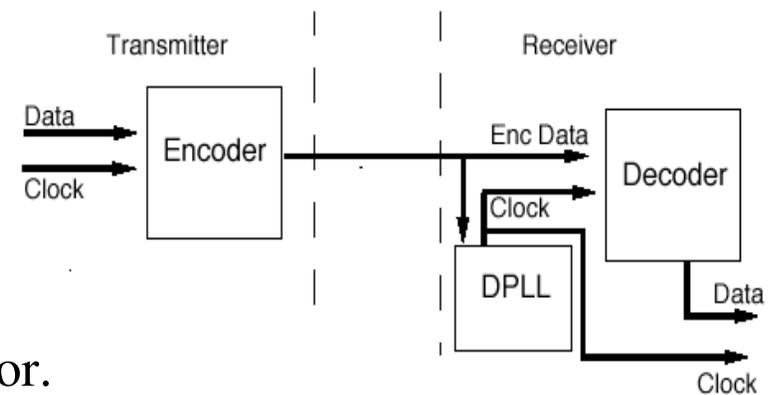


Sistemas SCADA

● Arquitectura Hardware

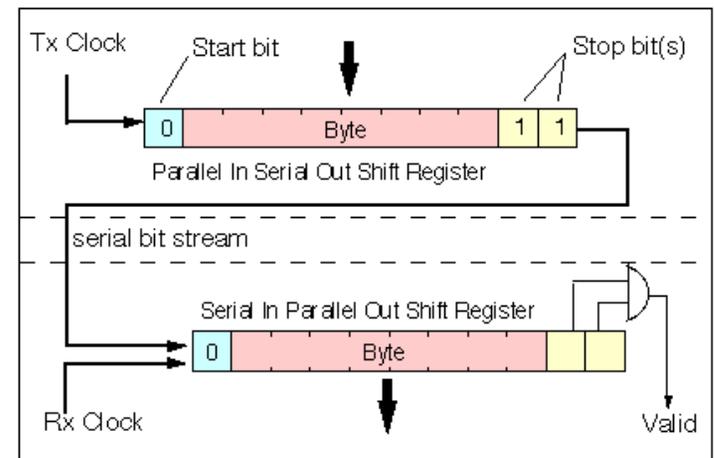
▶ Comunicaciones: **Comunicación Síncrona**

El **modem** transmisor debe enviar un pulso de reloj, para asegurar que el receptor está trabajando con la **misma velocidad de reloj**. Usualmente el reloj del transmisor es utilizado para sincronizar un reloj en el receptor.



▶ Comunicaciones: **Comunicación Asíncrona**

No es necesario la sincronización de relojes, unas señales de “**Inicio de mensaje**” y “**Fin de mensaje**”, son requeridas para informar al receptor del estatus del mensaje.





Sistemas SCADA

- **Arquitectura Hardware**

- ▶ **Comunicaciones: Cable telefónico o Radio**

- El medio de comunicación en los sistemas SCADA es determinado por dos factores: La tasa de datos a transmitir y el costo.

- En los inicios de los SCADAS, utilizar como medio el cable telefónico resultó muy popular, especialmente con las mejoras introducidas en los materiales que evitan las interferencias.

- Radios UHF han sido desarrollados especialmente para SCADAs, ellos ofrecen **flexibilidad, bajo costo y alta confiabilidad.**

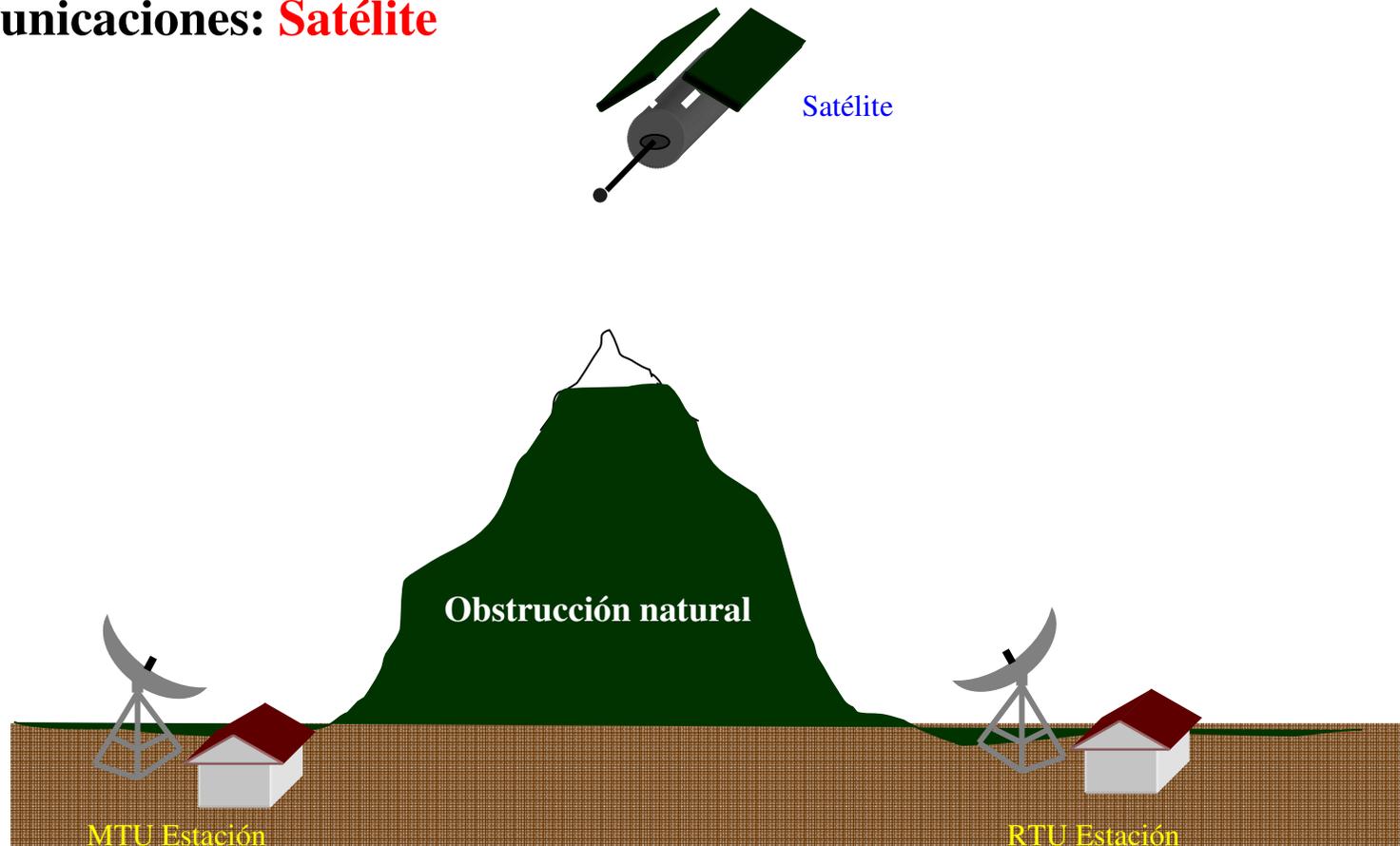
- Otros medios, especialmente la telefonía celular y comunicaciones satelitales comienzan a ofrecer alternativas atractivas a las comunicaciones en los sistemas SCADA.



Sistemas SCADA

- **Arquitectura Hardware**

- ▶ **Comunicaciones: Satélite**

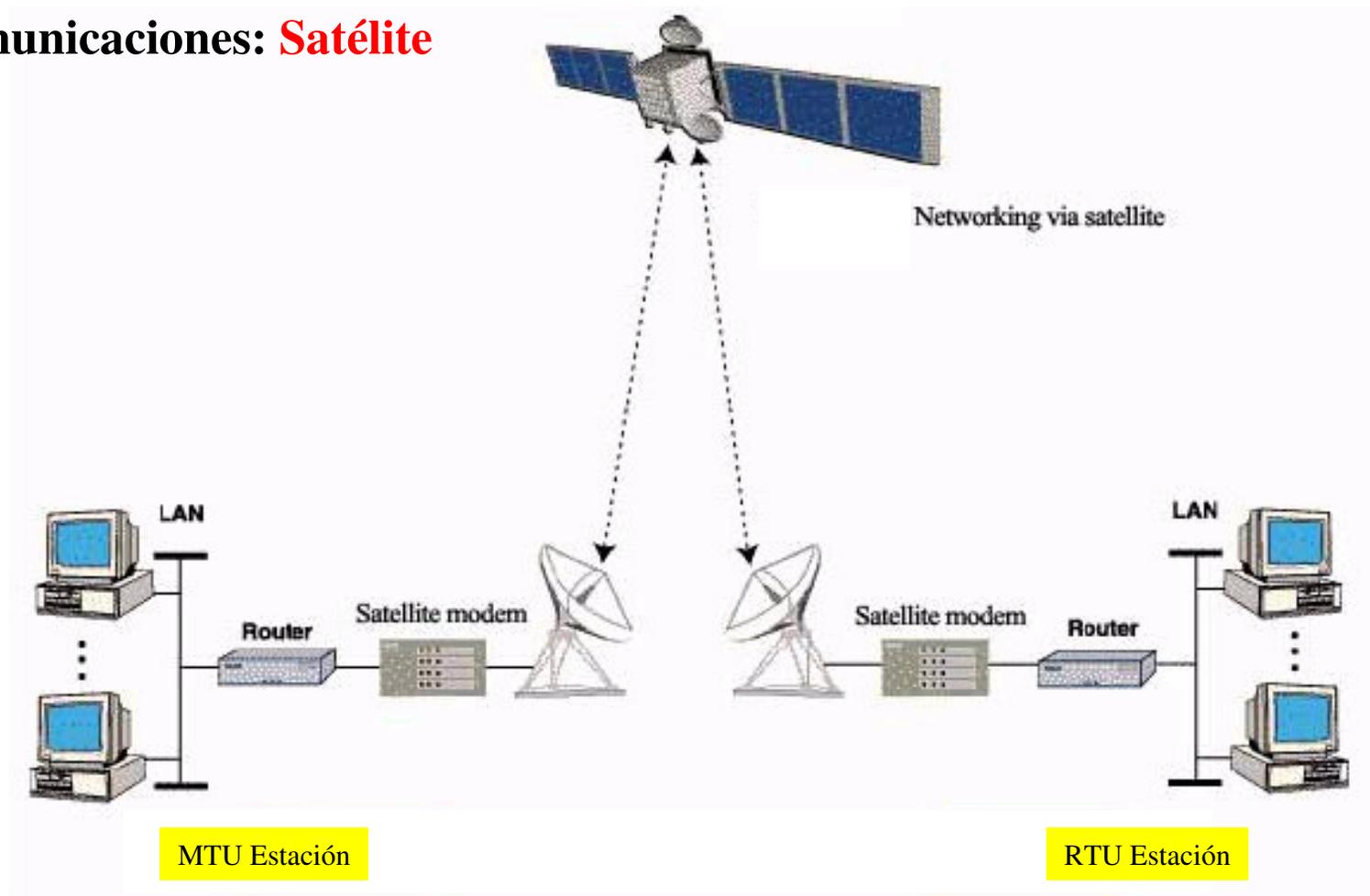




Sistemas SCADA

- **Arquitectura Hardware**

- ▶ **Comunicaciones: **Satélite****

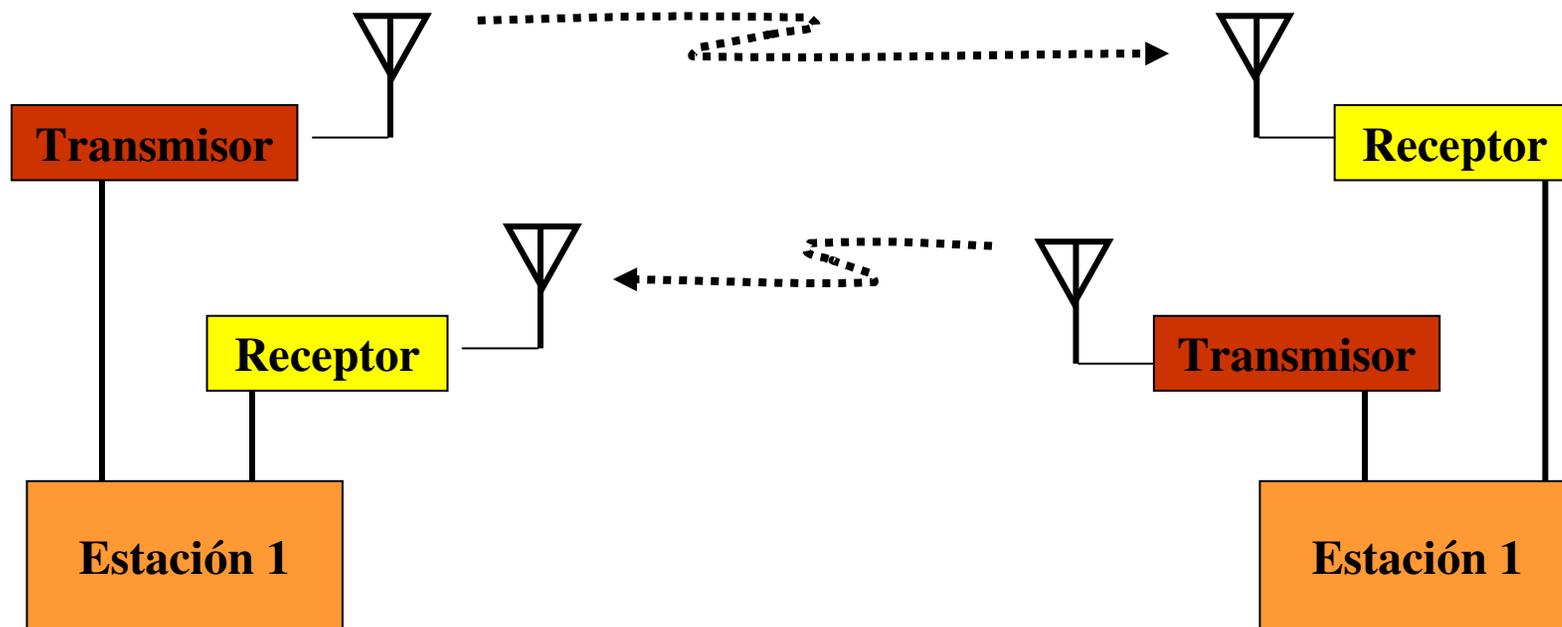




Sistemas SCADA

- **Arquitectura Hardware**

- ▶ **Comunicaciones: Simplex o Duplex (Half-duplex/Full-duplex) comunicación**





Sistemas SCADA

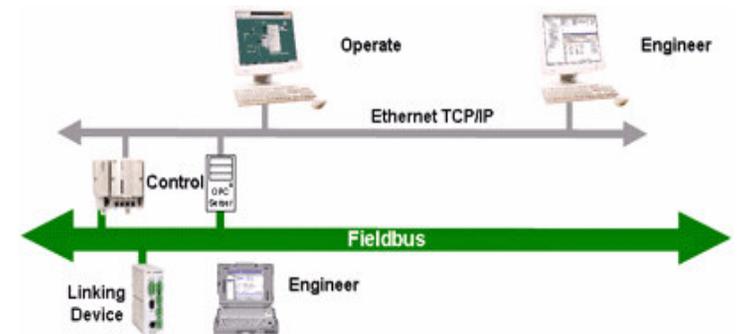
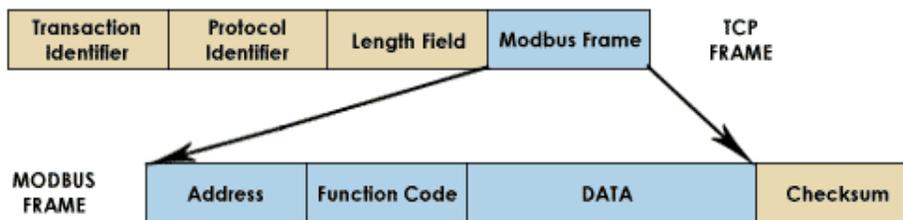
Arquitectura Hardware

Comunicaciones: Protocolo



- Es el conjunto de reglas que definen el significado de un grupo de palabras binarias.
- El protocolo establece, por ejemplo, que los mensajes enviados desde una MTU a una RTU, son una serie de dígitos binarios.
- El protocolo indica que representa el primer bit, el segundo,, el 247..
- El mismo protocolo utilizado por el que envía el mensaje, debe ser utilizado por el receptor para decodificarlo.

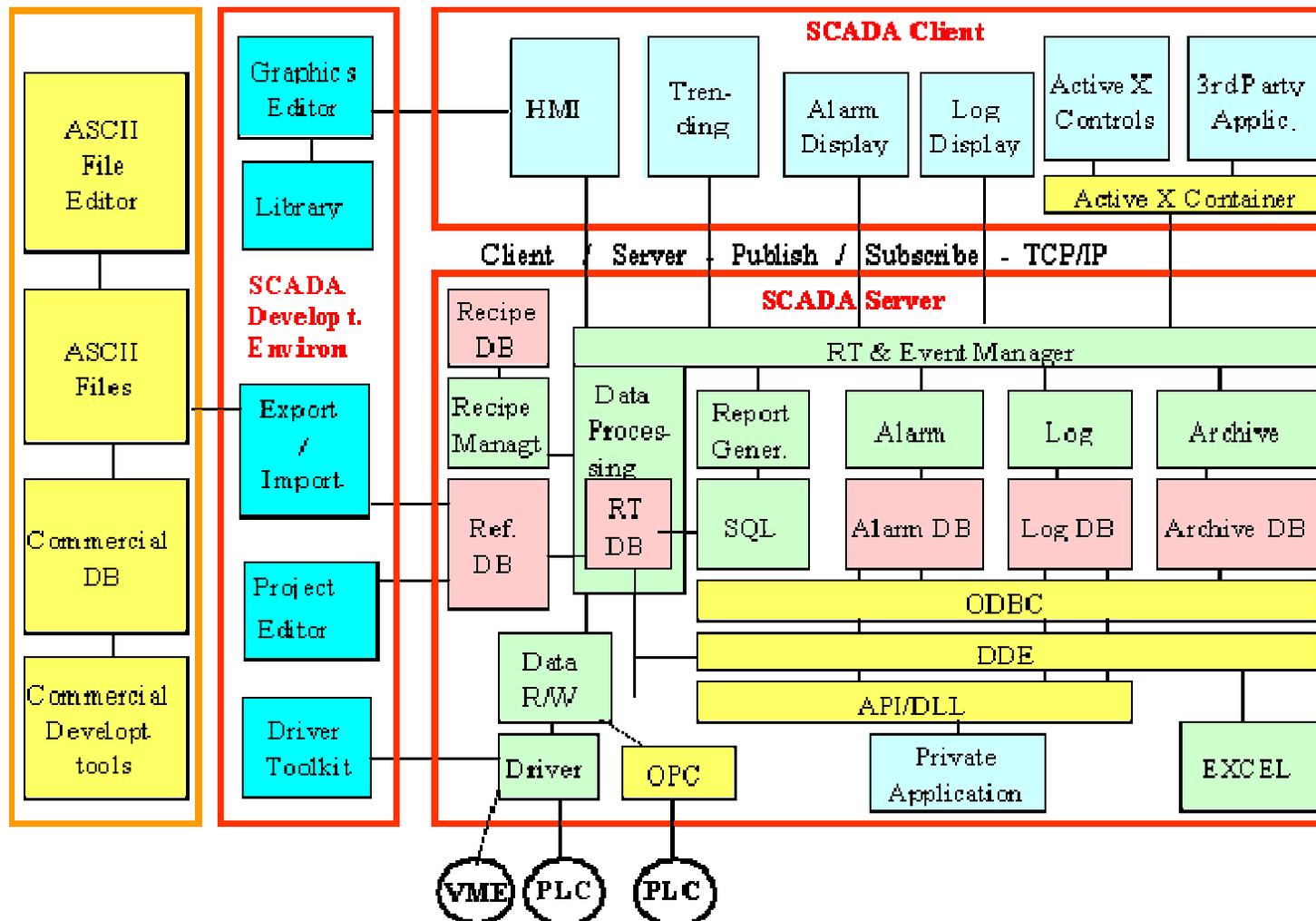
Ej.





Sistemas SCADA

Arquitectura Software





Sistemas SCADA

- **Arquitectura Software: Servidor SCADA**
 - Responsable de la adquisición de datos y su organización
 - Frecuencia de muestreo configurable
 - Base de datos con la información del proceso
 - Tiempo real
 - Gestión de alarmas priorizadas
 - Registro de evolución de los parámetros del proceso



Sistemas SCADA

● **Arquitectura Software: Cliente SCADA**

- Recoge y muestra los datos deseados del servidor SCADA
- Procesa los datos mostrando tendencias, históricos, log's, alarmas,...
- Permite la interconectividad entre programas ofimáticos comunes (importar/exportar datos)
- Control de acceso
- Generación de informes

● **Arquitectura Software: Comunicación**

Si son máquinas diferentes, la comunicación vía **TCP/IP** es la más común



Sistemas SCADA

● **Interfaces graficas: Representación del Proceso**

▶ En el momento de crear una representación gráfica, la elección de la iconografía adecuada, la distribución en la pantalla y la elección de los colores adecuados facilita la interacción entre el usuario y el proceso. Algunas consideraciones útiles:

- Las pantallas, o ventanas, tendrán una apariencia consistente, con zonas diferenciadas para mostrar el sinóptico, botoneras, menús y presentación de mensajes del sistema. Dicha distribución se conservará durante todo el proyecto, siempre que sea posible
- La información numérica presentada se hará sobre los elementos gráficos que la generan.
- La representación del proceso (sinópticos) se organizará de acuerdo con la distribución física de las células de producción, teniendo en cuenta que el sentido de lectura y observación normal es de izquierda a derecha y de arriba hacia abajo.



Sistemas SCADA

● **Interfaces graficas: Representación del Proceso**

- La utilización de colores, por ejemplo fondos y líneas (flechas, marcos, etc.) ayuda a la comprensión rápida de la información. Un número excesivo de ellos la oculta o dificulta.
- La presencia de intermitencias llama la atención del observador, pero dificulta su lectura, por tanto se recomienda su utilización en partes gráficas y no en texto.
- Debe establecerse un uso lógico de la coloración y se preverá la diferenciación de información por otros procedimientos (texto, parpadeo, etc.) en caso de utilizar monitores monocromos, en lugares con exceso de iluminación exterior o de enfermedades como daltonismo en el operario.
- Aunque no existe una normativa a aplicar a la hora de representar el proceso en el sinóptico, la nomenclatura propuesta por la ISA (Instrument Society of America) para la documentación de procesos, es ampliamente utilizada y puede ser una forma de unificar las representaciones en el sistema de monitorización



Sistemas SCADA

- Interfaces graficas: Representación del Proceso**

Significado habitual de los colores en ambientes tecnológicos

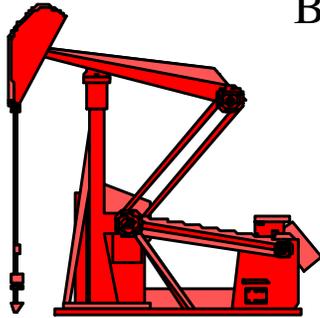
Rojo	Amarillo	Verde	Azul	Púrpura	Blanco	Gris	Negro
Peligro. Gas o líquido inflamable. Positivo en fuentes de alimentación. Señalización de paro.	Circuitos de calefacción. Agentes oxidantes. Elementos radiactivos. Aviso de peligro. Precaución.	Rejas de Protecc. Elec. Nitrógeno. Permiso. Canalización de aceite.	Colector de transistores. Precaución reparación. Material de protección de tuberías. Agua. Mar.	Negativo en fuentes de alimentación. Radiaciones. Materiales valiosos..	Regulación de tráfico.	Líneas de alimentación alterna. Canalización de vapor.	Tierra en sistemas eléctricos. Materiales corrosivos. Contornos geográficos.



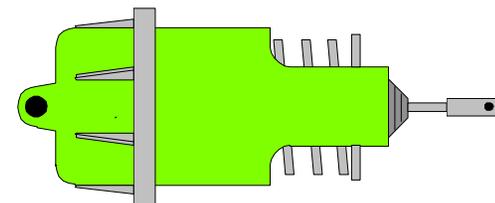
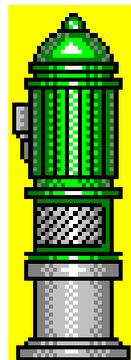
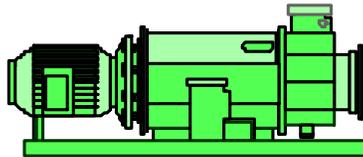
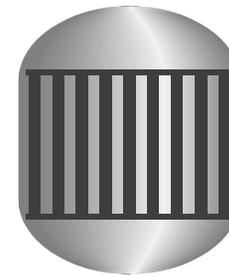
Sistemas SCADA

- Interfaces graficas: **Sinópticos**

Bombas



Motores

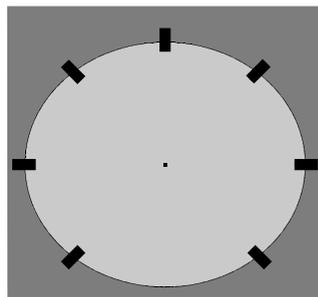
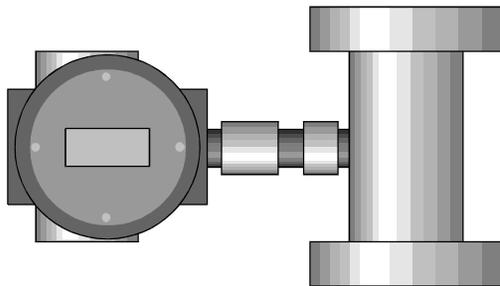




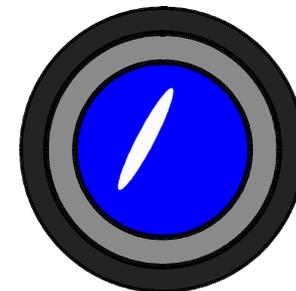
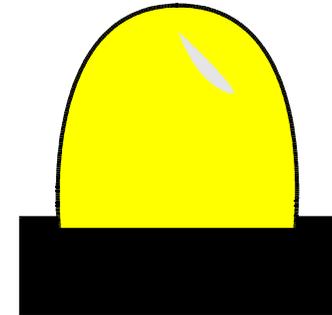
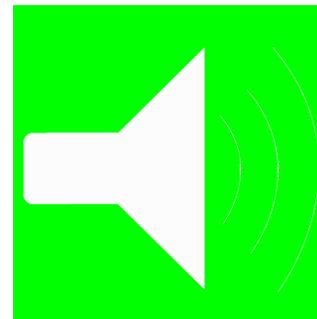
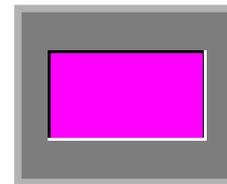
Sistemas SCADA

- Interfaces graficas: **Sinópticos**

Medidas



Indicadores

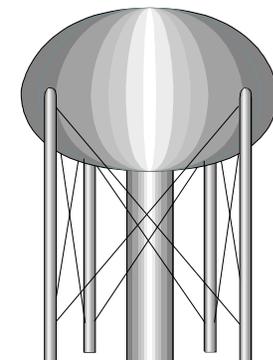
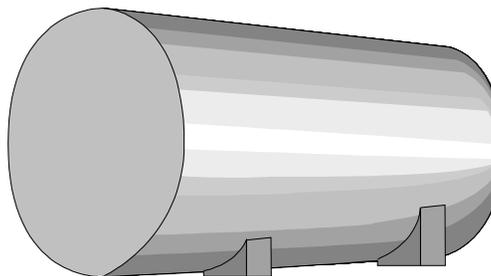
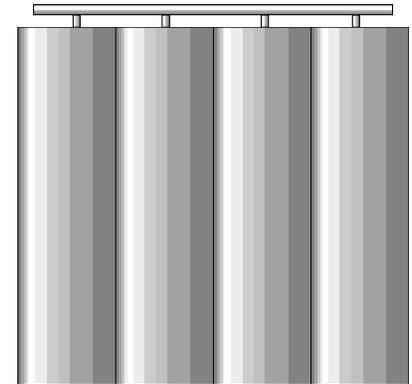
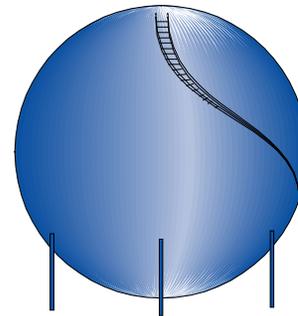
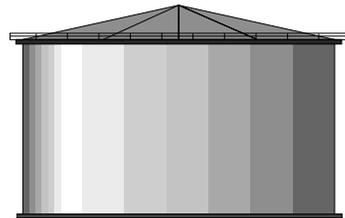
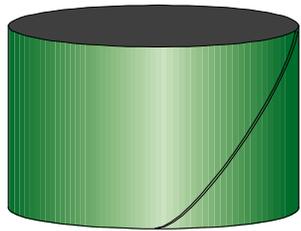




Sistemas SCADA

- Interfaces graficas: **Sinópticos**

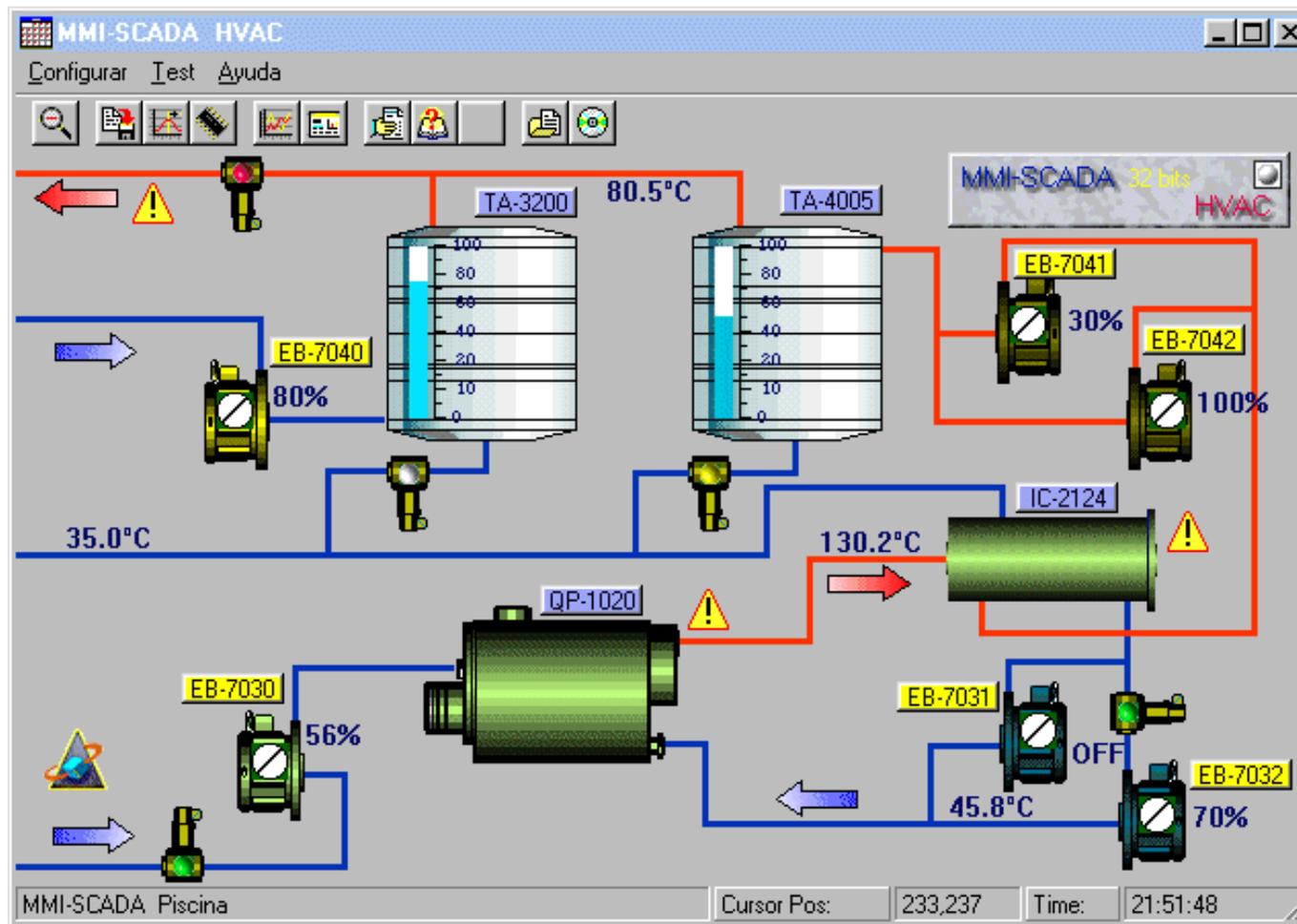
Tanques





Sistemas SCADA

- Interfaces graficas: Representación de proceso





Sistemas SCADA

● Alarmas

▶ Las desviaciones en la magnitud de una variable superiores a unos límites especificados son concebidas en los sistemas de monitorización como alarmas.

▶ La automatización de la tarea de vigilancia del proceso se logra en los entornos de monitorización mediante estas alarmas usadas para detectar situaciones de comportamiento anómalo.



Tipos de alarmas

▶ Una clasificación simple de los diferentes tipos de alarmas pueden realizarse en función del tipo de variable sobre la que se definen.

▶ Así, podemos diferenciar entre **alarmas discretas** (variables discretas) y **alarmas sobre variables continuas** o de umbral.



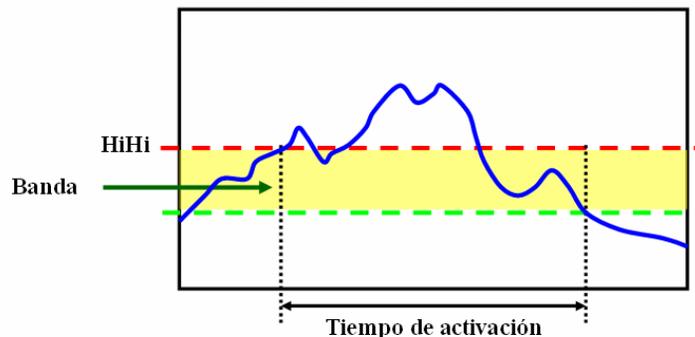
Sistemas SCADA

- **Alarmas: Alarmas discretas**

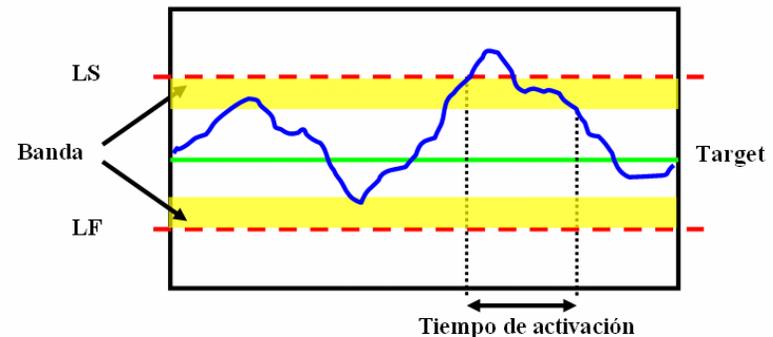
- ▶ Son indicadores de un **cambio binario** en el estado de la variable que representan. Una variable discreta puede ser utilizada para definir el estado de un sistema (paro/marcha, manual/automático), la presencia o ausencia de material, un fin de carrera, detección de paso, etc.

- **Alarmas: Alarmas continuas**

- ▶ Se utilizan umbrales numéricos para **designar los límites de operación normal** de dicha variable. Su elección se hará de acuerdo con el significado físico de la variable y los límites permitidos para el funcionamiento correcto del proceso.



Umbrales de alarmas absolutos



Umbrales de alarma relativos



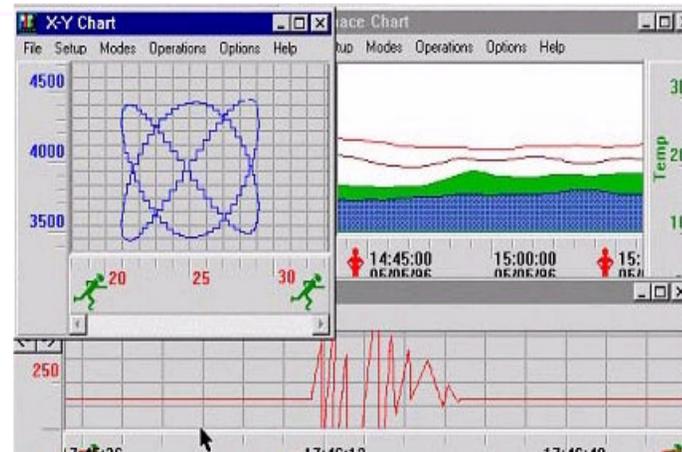
Sistemas SCADA

● Gráficas y tendencias

▶ Aún conociendo el concepto de alarma asociado a las variables, la **visualización gráfica de su historia** permite al operario **anticipar algunas situaciones**.

▶ La observación de la **evolución** que dichas gráficas experimentan y su comparación con experiencias previas o con otras variables relacionadas que permiten al operario **establecer conjeturas/hechos** que se verifican o no, a medida que el proceso evoluciona y la gráfica se actualiza.

▶ El operario de esta forma observa la **tendencia** del proceso a través de las gráficas.



MINISTERIO DE MEDIO AMBIENTE	
OBRA DE LLEGADA	
Bombas de agua bruta	
Consigna caudal de agua bruta	
Banda muerta nivel agua bruta	
Tmpo antes arranque de bomba	
Tiempo antes paro de bomba	
Tmpo. máx. funcionam. BB-201B	
Tmpo. máx. funcionam. BB-201C	
Compuertas de entrada/sada.	
Consigna para abrir un 2º canal	
Consigna para cerrar un 2º canal	
Tamices	
Tiempo marcha tamices	
Tiempo paro tamices.	
Retardo desconexión tornillo	
Desnatador	
Tiempo marcha clasificador	
Tiempo paro clasificador	
Tiempo marcha desnatador	
Tiempo paro desnatador	
Tiempo apertura valv. desengrase	
DECANTACION SECUNDARIA	
Arqueta de flotantes	
Tmpo. antes alarma arqueta	



Sistemas SCADA

● **Gestión y registro de alarmas**

- ▶ Los sistemas SCADA disponen de mecanismos de **filtrado de alarmas**, que consisten básicamente en asociar prioridades a las alarmas en el momento de su definición.
- ▶ En consecuencia el mecanismo de filtrado consiste en la activación de alarmas de acuerdo con **su prioridad**.
- ▶ La gestión de alarmas supone el seguimiento y registro temporal, (momento en que se inicia una alarma, la duración, el instante en que el operario la reconoce o el momento en que desaparece).
- ▶ Estos sucesos son conocidos como **eventos**.



Sistemas SCADA

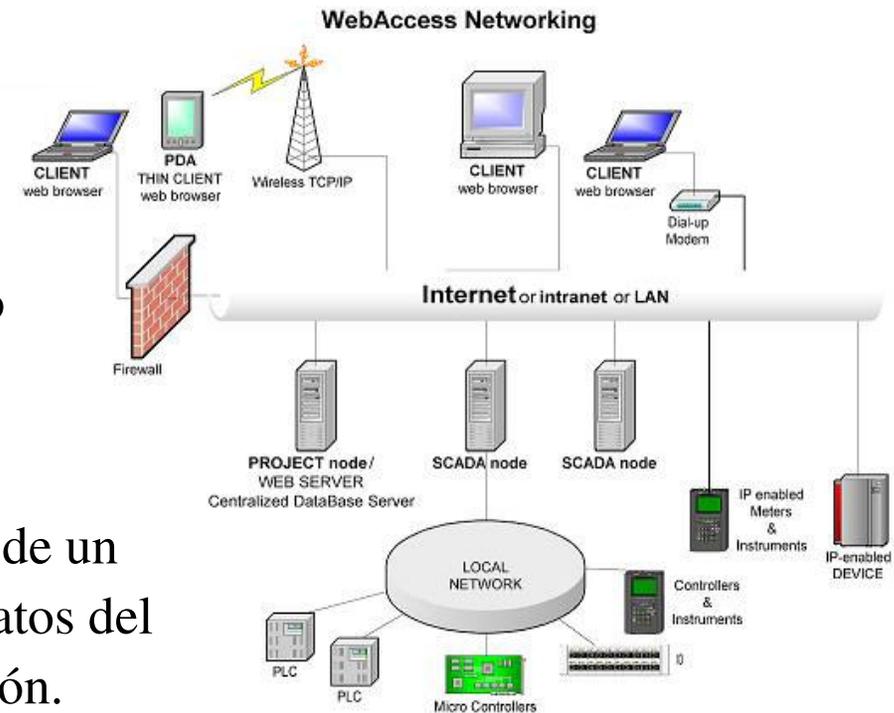
● Históricos y bases de datos

▶ El registro continuo de datos permite la posterior **recuperación** de éstos y por tanto también su graficación, **comparación**, **creación de estadísticas**, **análisis**, etc.

▶ Por tanto, será una funcionalidad básica de un sistema de monitorización el registro de datos del proceso (históricos) y su posible explotación.

▶ La utilización de bases de datos externas permite **el acceso** tanto desde el entorno de monitorización como desde otras aplicaciones a través de lenguajes estándar, resultando la forma más cómoda de integración de los sistemas informáticos industriales.

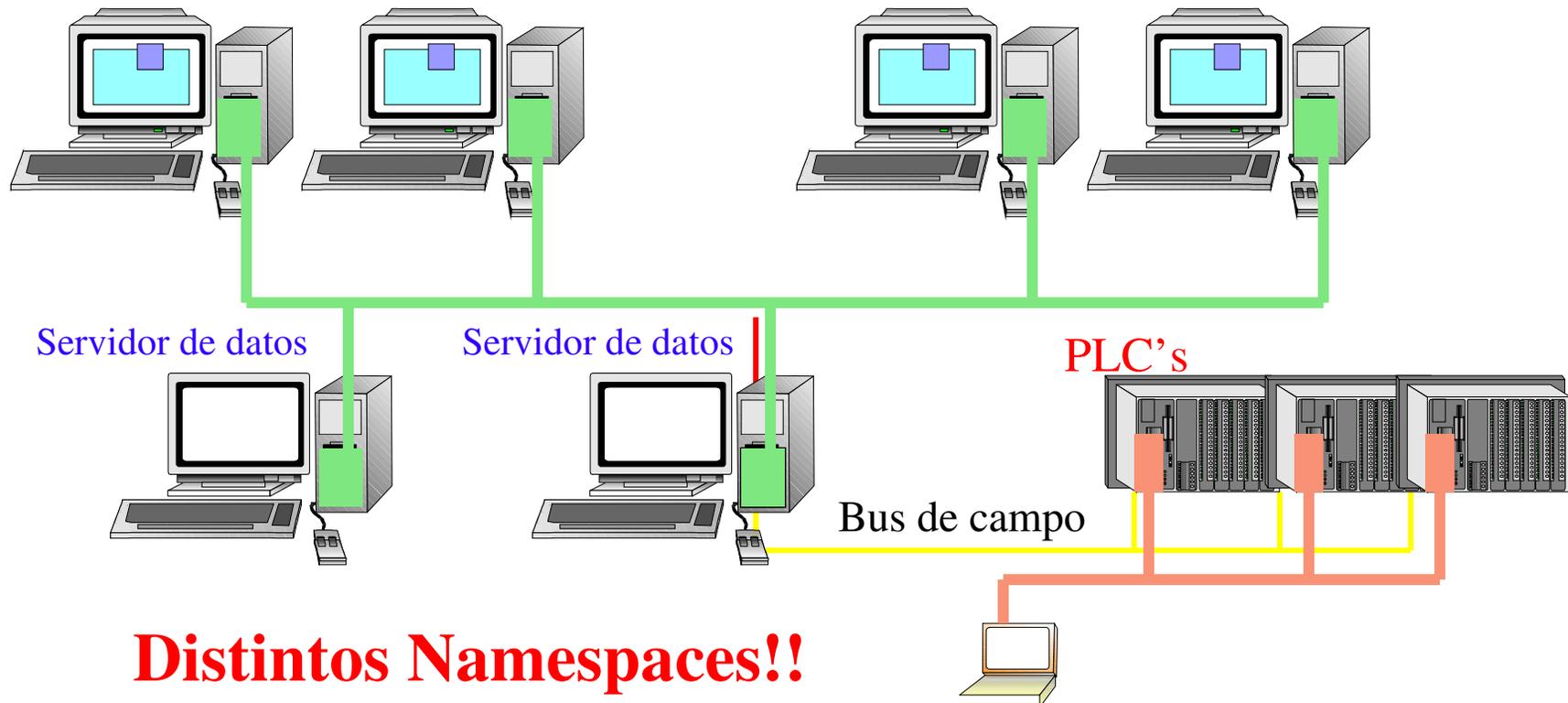
▶ **SQL** (Structured Query Language) es uno de los lenguajes de interrogación más extendidos y adoptado.





Sistemas SCADA

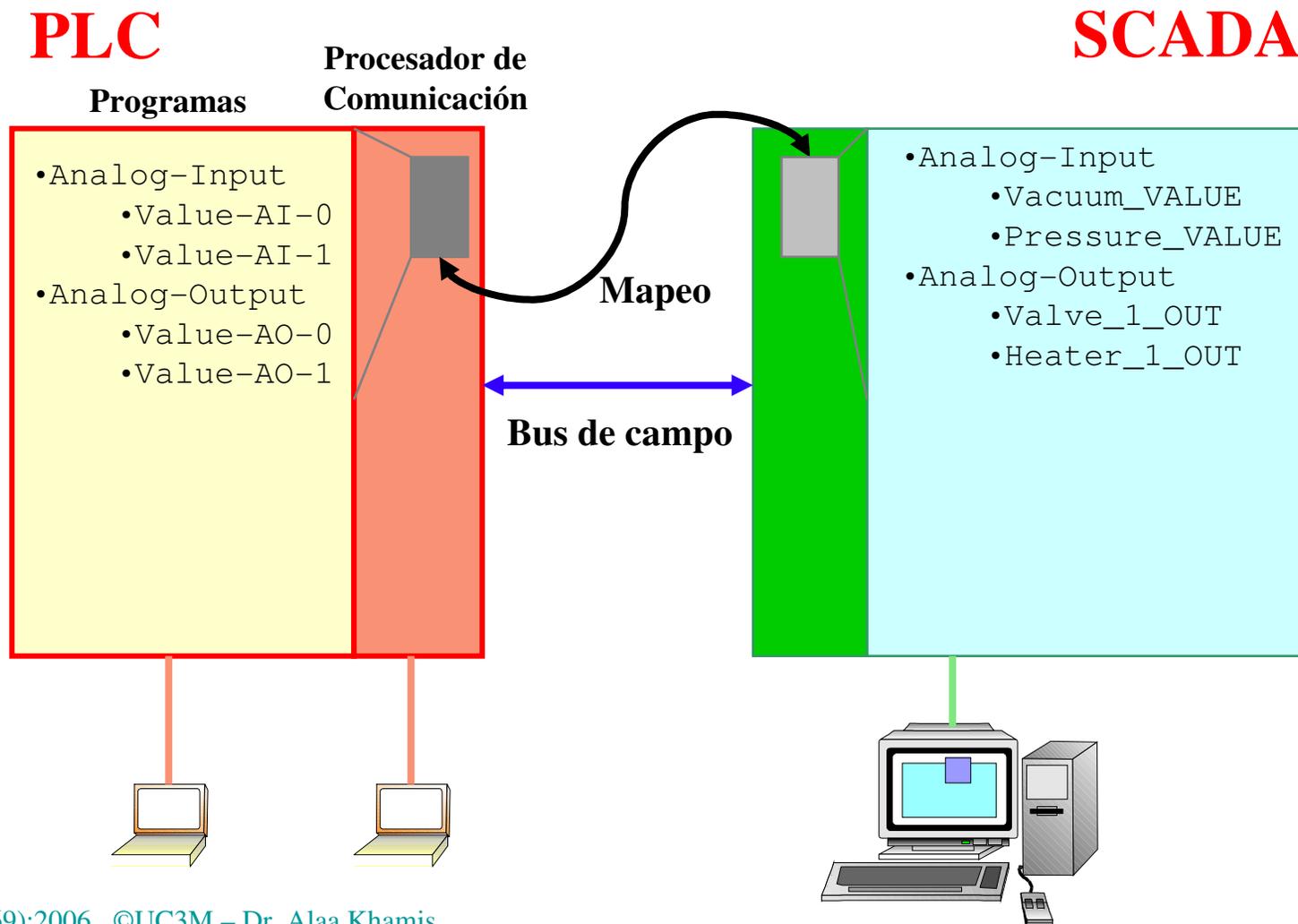
- Configuración





Sistemas SCADA

● Configuración





Sistemas SCADA

● Configuración

SCADA

Servidor de datos

- Analog-Input
 - Vacuum_VALUE
 - Pressure_VALUE
- Analog-Output
 - Valve_1_OUT
 - Heater_1_OUT

Generación de alarmas

```
(generic) Alarm scripts:  
  
IF (Pressure_VALUE >  
Pressure_HIHI_ALARMVAL)  
    {Pressure_ALARM_HIHI=TRUE  
    Pressure_ALARM_STATE=CRITICAL  
    Pressure_ALARM_COLOR=RED }  
  
IF (Pressure_VALUE <  
Pressure_LOLO_ALARMVAL)  
    {Pressure_ALARM_LOLO=TRUE  
    Pressure_ALARM_STATE=ATTENTION  
    Pressure_ALARM_COLOR=BLUE }
```

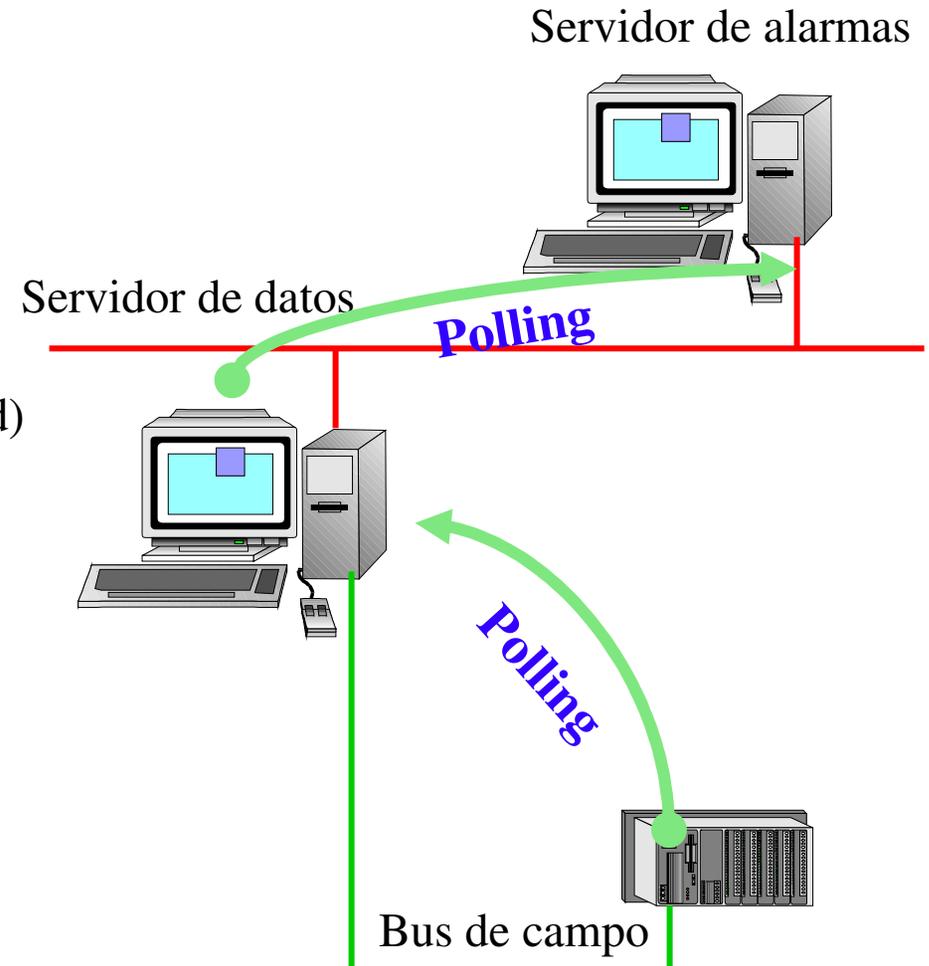


Sistemas SCADA

● Acceso a los datos

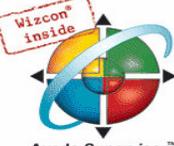
El servidor de alarmas típicamente saca/sondea (*poll*) datos del servidor de datos.

- ▶ (-> impacto en el ancho de banda de la red)
- ▶ Técnicas más avanzadas como publisher/subscriber (editor/suscriptor)





Principales sistemas SCADA

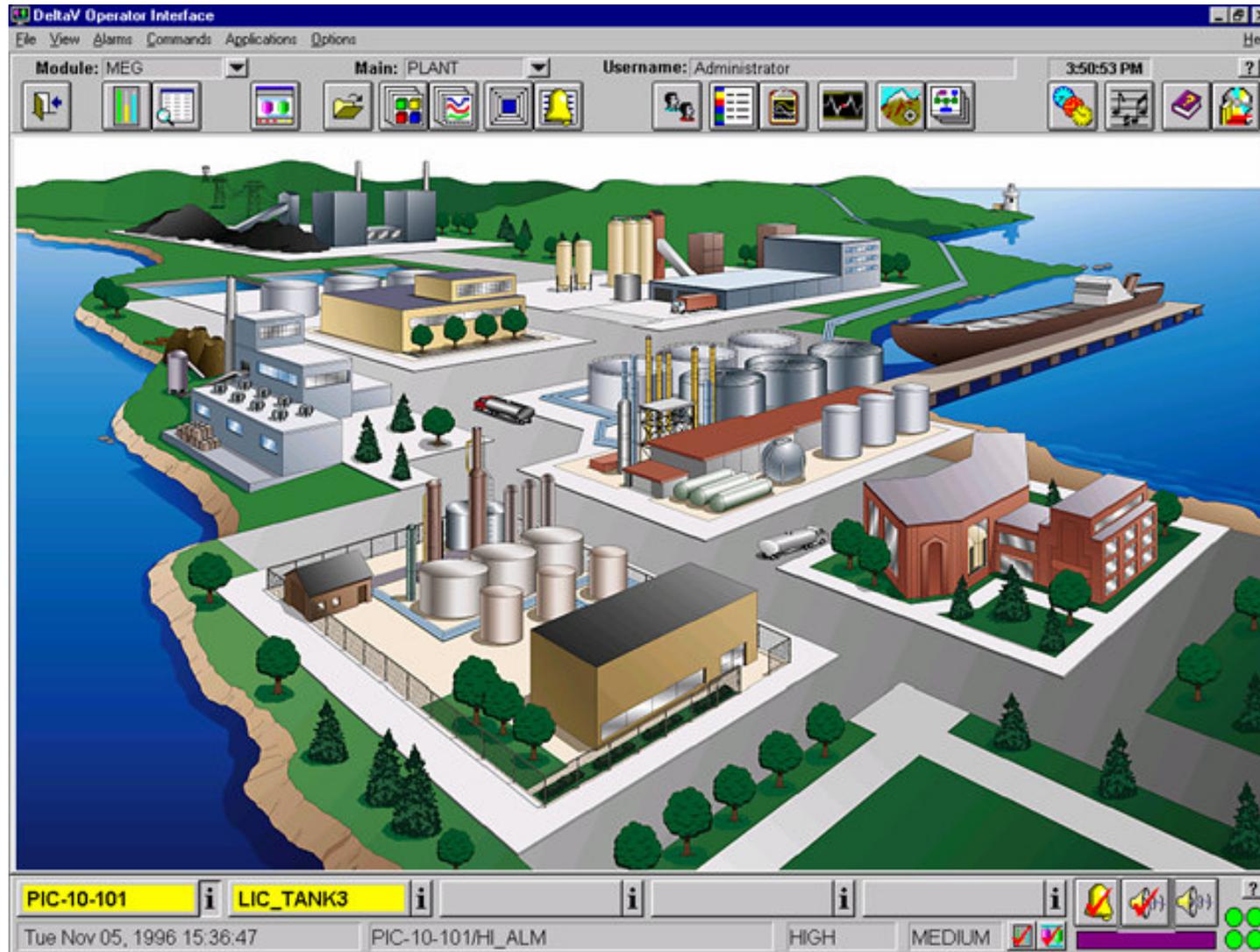
Proficy [®] HMI/SCADA - iFIX	
Proficy [®] HMI/SCADA - CIMPLICITY	
	
FactoryLink	
invensys [®] 	
	
	
	
	
	
	
	
	

Proveedor	URL	Producto
USDATA	http://www.usdata.com/	Factory Link 7
Advantech	http://www.advantech.com	Paradym-31
eMation	http://www.emation.com/	WizFactory
GE Fanuc	http://www.gefanuc.com/	Cimplicity
Iconics	http://www.iconics.com/	Genesis32
Intellution	http://www.intellution.com/	Intellution Dynamics
National Instruments	http://www.ni.com/	LabView Lookout 4.5
Nematron	http://www.nematron.com/	HMI/SCADA Paragon
Opto 22	http://www.opto22.com/	FactoryFloor Software
Rockwell Automation	http://www.software.rockwell.com/	RSView32
Siemens	http://www.siemens.com	- HYBREX (Hybrid Expert System) - WinCC HMI Ver. 5.0 - Web Control Center (webCC) - SIMATIC WinAC ODK (Open Developer Kit) - SIMATIC WinAC (Windows Automation Center) - IMATIC PLCSim - SIMATIC Protocol
TA-Engineering Products	http://www.ta-eng.com/home.htm	Aimax
Wonderware	http://www.wonderware.com/	FactorySuite 2000 SuiteVoyager 1.0



Principales sistemas SCADA – DeltaV

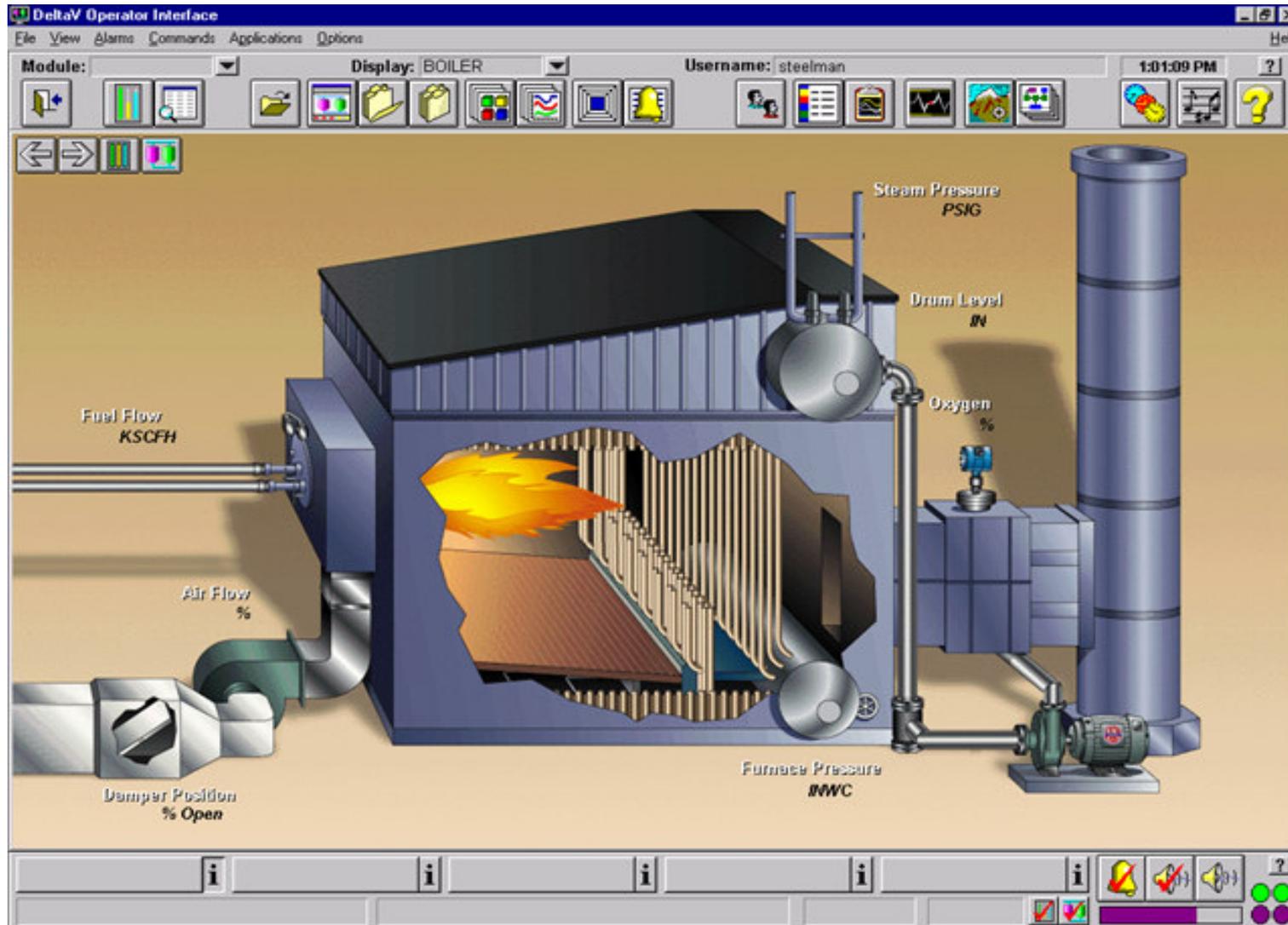
<http://easydeltav.com/>





Principales sistemas SCADA – DeltaV

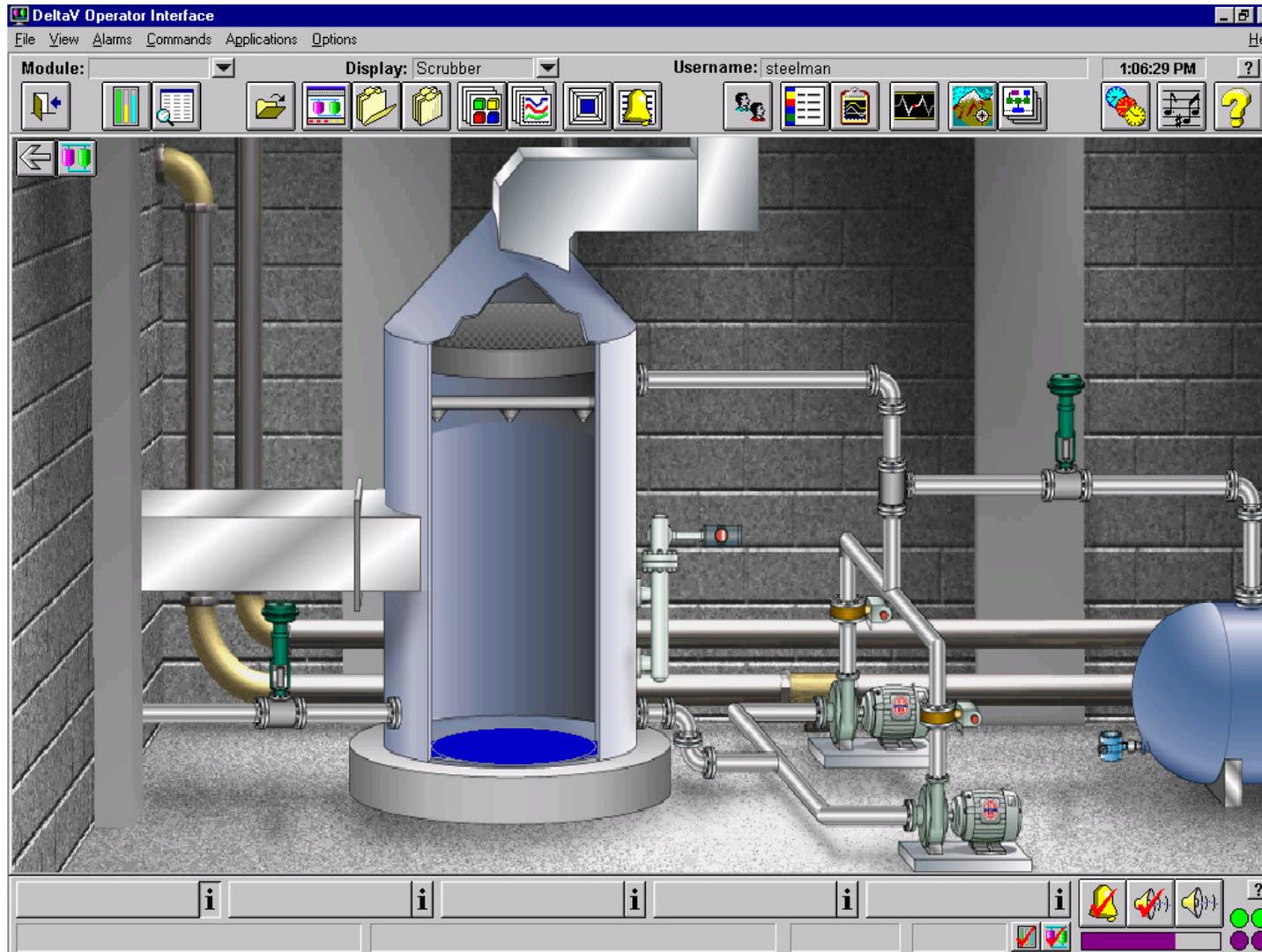
<http://easydeltav.com/>





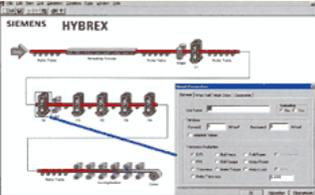
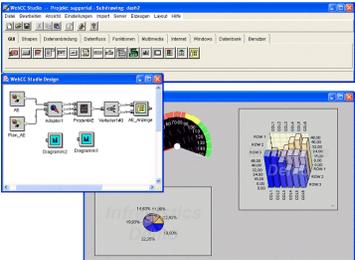
Principales sistemas SCADA – DeltaV

<http://easydeltav.com/>





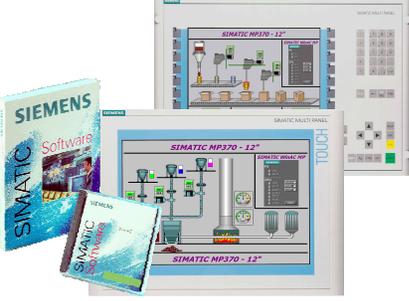
Principales sistemas SCADA – Productos Siemens

Producto	Descripción
HYBREX (Hybrid Expert System) 	Herramienta de simulación que permite realizar cambios virtuales en la planta y observar sus resultados sin ningún riesgo. Está específicamente orientada a procesos de laminado en plantas siderúrgicas y se puede utilizar en cualquiera de las etapas del ciclo de vida de la planta, desde construcciones nuevas hasta plantas en procesos de optimización y modernización.
Web Control Center (webCC) 	Permite que una gran variedad de información desde varias áreas de la planta se despliegue simultáneamente en una sola interfase de usuario. Así, se facilita el control corporativo y la operación y monitoreo de las instalaciones de proceso y plantas de producción. Mediante un navegador de internet basado en Java se tiene acceso simultáneo a diversos sistemas de control central, tales como SiiX-IS, WinCC, Sicalis PMC y Simatic PCS 7. Bases de datos y videos se integran desde la red y pueden enviarse correos electrónicos en cualquier momento. Los datos maestros de sistemas SAP y otros pueden consultarse desde cualquier parte del mundo haciendo realidad el concepto de compañía virtual.



Principales sistemas SCADA – Productos Siemens



Producto	Descripción
<p>SIMATIC WinAC (Windows Automation Center)</p> 	<p>Mediante esta solución integrada para control, HMI, redes y procesamiento de datos –todos en la misma plataforma– es posible emular el funcionamiento de un PLC en una PC (la parte de control permite que se utilice una PC para emular a un PLC). Esta solución se configura, programa y mantiene con Simatic Step 7, el ambiente estándar de desarrollo para los PLCs de Siemens. Las funciones de computación y visualización proveen todas las interfases que se necesitan para ver el proceso y modificarlo mediante aplicaciones estándar, tales como Microsoft Excel, Visual Basic o cualquier otro paquete estándar HMI.</p>
<p>SIMATIC WinAC ODK (Open Developer Kit)</p> 	<p>Herramienta de software para extender las funciones de WinAC que ofrece una serie de interfaces abiertas de alta velocidad para integrar datos de procesos en tiempo real con otras aplicaciones de software. Estas interfases, tales como ActiveX y OPC, están diseñadas para proveer a las aplicaciones del usuario una manera asíncrona de acceder a los datos del proceso. Permite que el código escrito por el usuario (C, C++, Visual Basic, Java) o aplicaciones de terceros se ejecuten como parte del ciclo de barrido del software WinAC.</p>



Principales sistemas SCADA – Productos Siemens

Producto	Descripción
<p>SIMATIC PLCSim</p>  <p>S7-PLCSIM © STEP 7 S7/M7</p>	<p>Herramienta de software que permite ejecutar programas de Simatic Step 7 en una PC. Así, se pueden depurar con mayor facilidad los programas antes de ingresarlos a los PLCs, facilitando el proceso de liberación de los sistemas de automatización. También funciona con sistemas Simatic WinCC, lo que permite verificar previamente los programas HMI.</p>
<p>SIMATIC Protool</p> 	<p>Herramienta de software que permite a los operadores visualizar e interactuar con los equipos a través de MMI. Comparte un ambiente de configuración común que permite el transporte de aplicaciones entre un amplio rango de plataformas de hardware, incluyendo Windows CE, Windows 95/98 y NT, así como con la línea de consolas de operador de SIMATIC.</p>



Principales sistemas SCADA – Productos Siemens



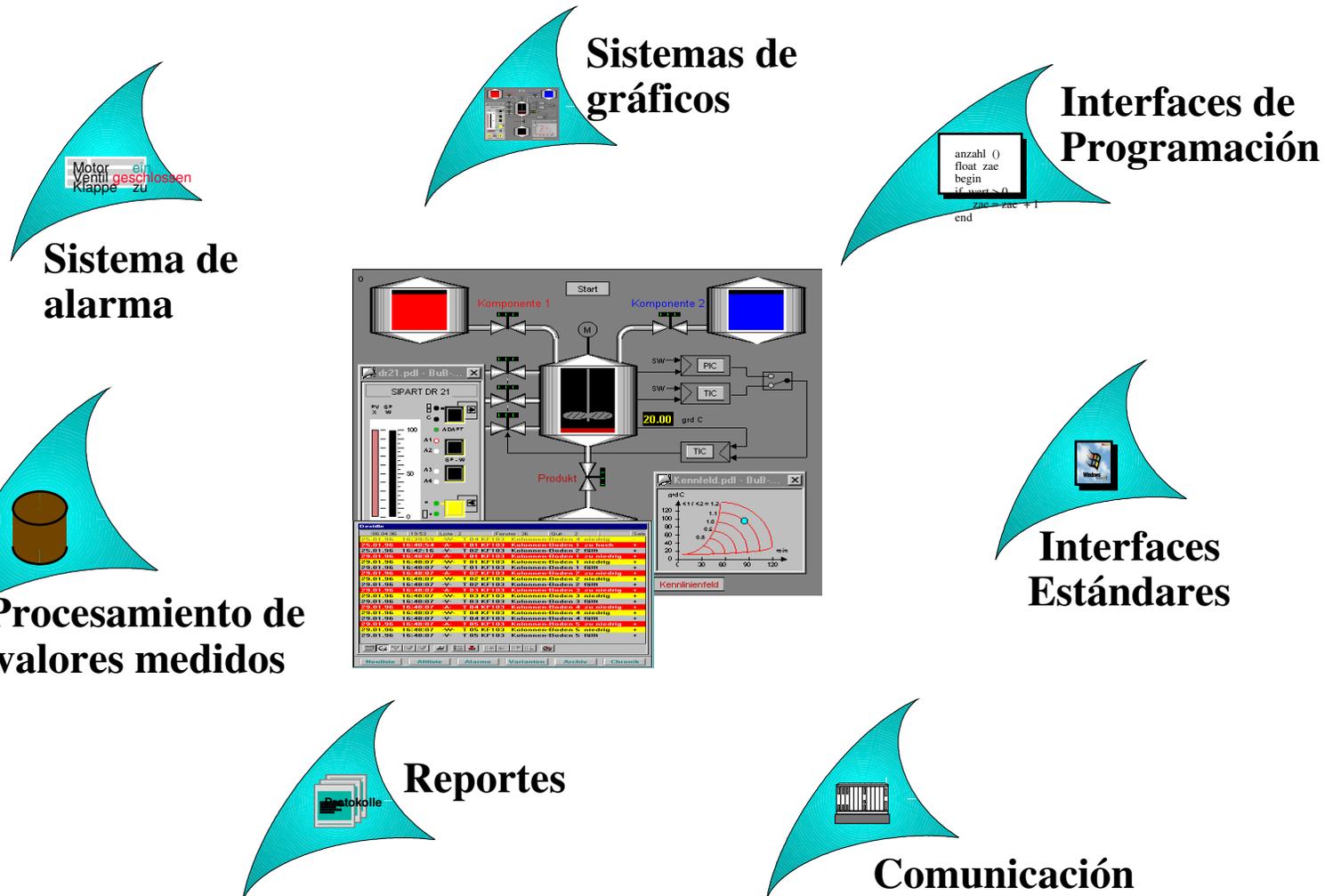
Producto	Descripción
<p data-bbox="212 532 514 581">WinCC HMI</p> 	<p data-bbox="590 532 1934 1409">Para integrar software en la manufactura se requiere usar normas abiertas que puedan enlazar fuertemente la información del piso de la planta y el sistema de negocios a través de ella. WinCC HMI, software de 32 bits integrado completamente con Microsoft Windows NT, combina las características estándar (gráficos, alarmas, administración de recetas, etcétera) con otras avanzadas (reportes, referencias entre proyectos, diagnóstico de proceso, soporte multilingüe y redundancia completa). Además, mejora su funcionalidad mediante la integración de bases de datos con MES/ERP, internet y tecnologías de cliente delgado.</p>



Principales sistemas SCADA



● Siemens WinCC Flexible





Principales sistemas SCADA



● Siemens WinCC Flexible

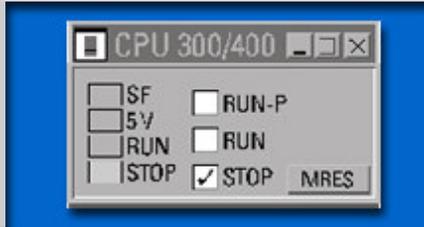
Operación

- Diagnostico
- Mantenimiento Remoto



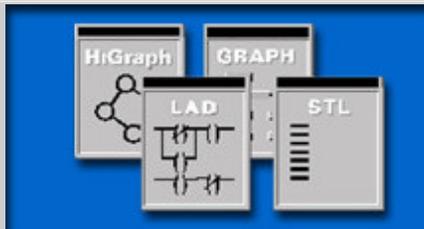
Pruebas

- Simulación
- Diagnostico
- Funciones de pruebas



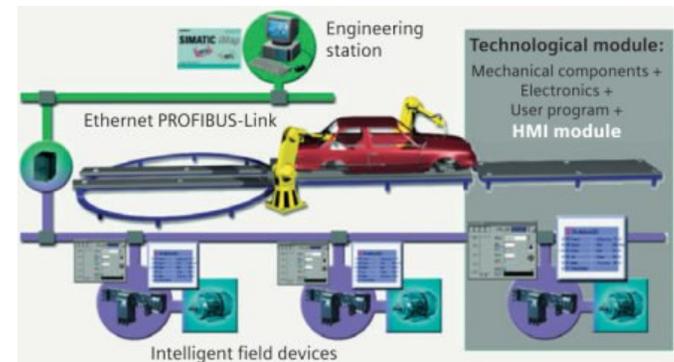
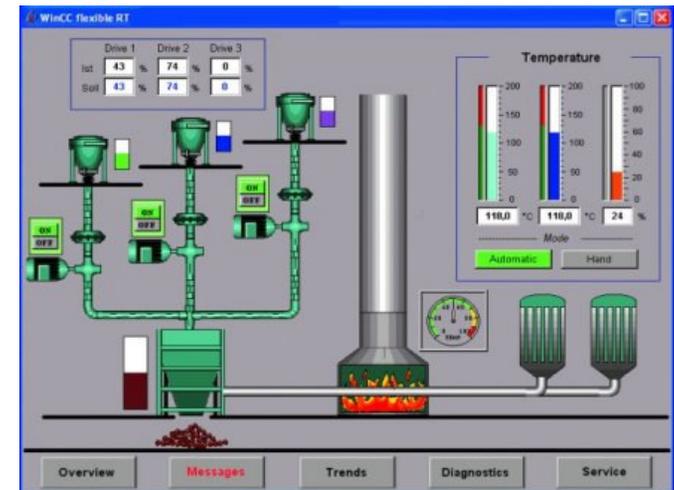
Programación

- Lenguajes estándares
- Orientada a tareas



Configuración Hardware

- Controladores
- Entradas/Salidas
- Redes





Resumen:

- **SCADA** (**S**upervisión, **C**ontrol y **A**dquisición de **D**atos) es un sistema basado en computadores que permite supervisar y controlar a distancia una instalación de cualquier tipo.
- Los componentes básicos de un SCADA son:
 - ▶ Unidad terminal maestra (Master Terminal Unit o MTU)
 - ▶ Múltiples terminales remotas (RTU - Remote Terminal Units)
 - ▶ Red de comunicación.
- Sectores de aplicación de los sistemas SCADA incluyen Producción automovilística y sus proveedores, Industria química y farmacéutica, Generación y distribución de energía, Industria del plástico y caucho, Metalurgia, Siderurgia, Industria alimenticia y de bebidas, Tratamiento y depuración de aguas, Industria de artes gráficas, Comercio y sector servicios, Construcción de maquinaria e instalaciones, Fabricación y transformación de papel y Transporte y tráfico.



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 10 – Jueves 18 de Mayo 2006

Supervisory Control And Data Acquisition

SCADA

(Adquisición de Datos y Control Supervisor)



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 11 – Miércoles 24 de Mayo 2006

Buses Industriales

Ref:

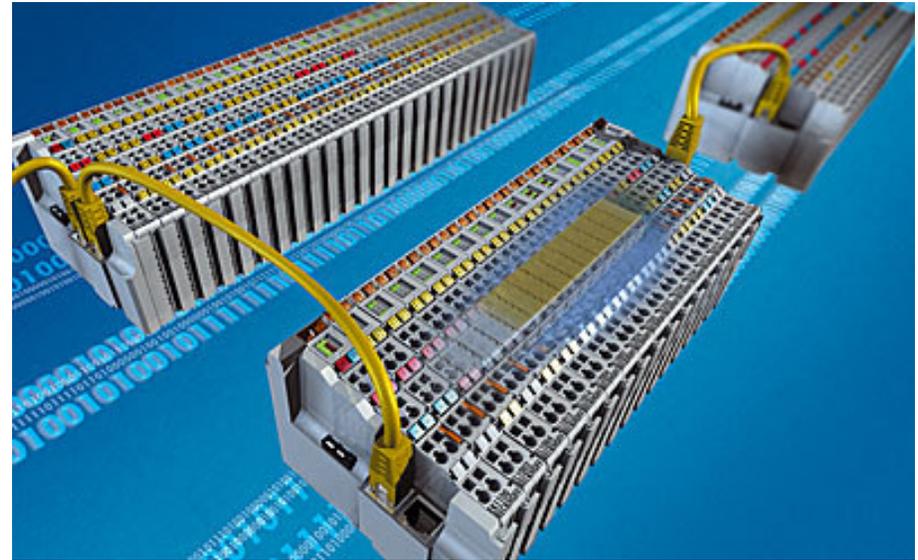
- Los apuntes del Profesor: José Manuel Pastor García, “Comunicaciones en Fabricación”, Universidad Carlos III de Madrid.
- Carlo Cloet , “Ethernet for Industrial Automation: Have Fieldbus wars moved to a new battlefield?”, EE290-O Presentation.
- R. Moreno. Ingeniería de la Automatización Industrial. 2º Edición, Ra-Ma, 2004.



Integración de componentes de un sistema industrial

Para integrar dos componentes de un sistema, los dos componentes tienen que saber los siguientes:

- ▶ ¿Quién es el maestro?
- ▶ ¿Dónde se guardan los datos y cómo se puede acceder a los mismos?
- ▶ ¿Quién inicia la comunicación?
- ▶ ¿Quién se encarga de la retransmisión si los datos no llegan al receptor?
- ▶ ¿Cuándo necesita el receptor los datos?
- ▶ ¿El receptor puede solicitar datos al emisor o tiene que esperar la próxima actualización de parte del emisor?





Sistema de cableado clásico

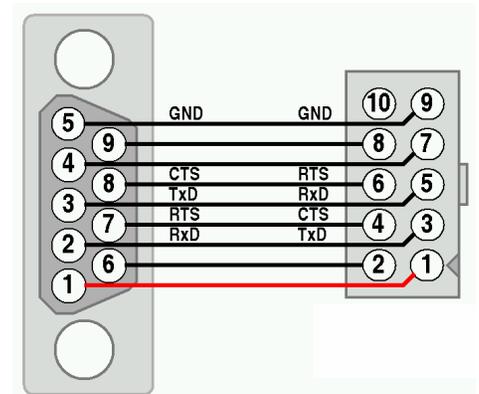
- ▶ Los cables deberán **ser numerados** en sus dos extremos y también utilizar mangueras de cables **multicolores** para evitar la confusión.
- ▶ Existen posible **excesiva longitud** de cable desde los captadores hasta el armario del autómeta.
- ▶ El **número de cables crece** con el número de captadores y accionadores complicando el proceso de puesta en marcha de una instalación.
- ▶ En los cables de señal se pueden introducir **ruidos** que den lugar a fallos en el funcionamiento, sobre todo en la medida y el control de señales analógicas.
- ▶ Los cables de potencia salen del armario de preaccionadores y deben llegar hasta el accionador, con las posible **caídas de tensión**.
- ▶ Los captadores y accionadores **sólo** pueden ser leídos y escritos desde el autómeta al que están cableados.





Terminales de comunicación serie RS232

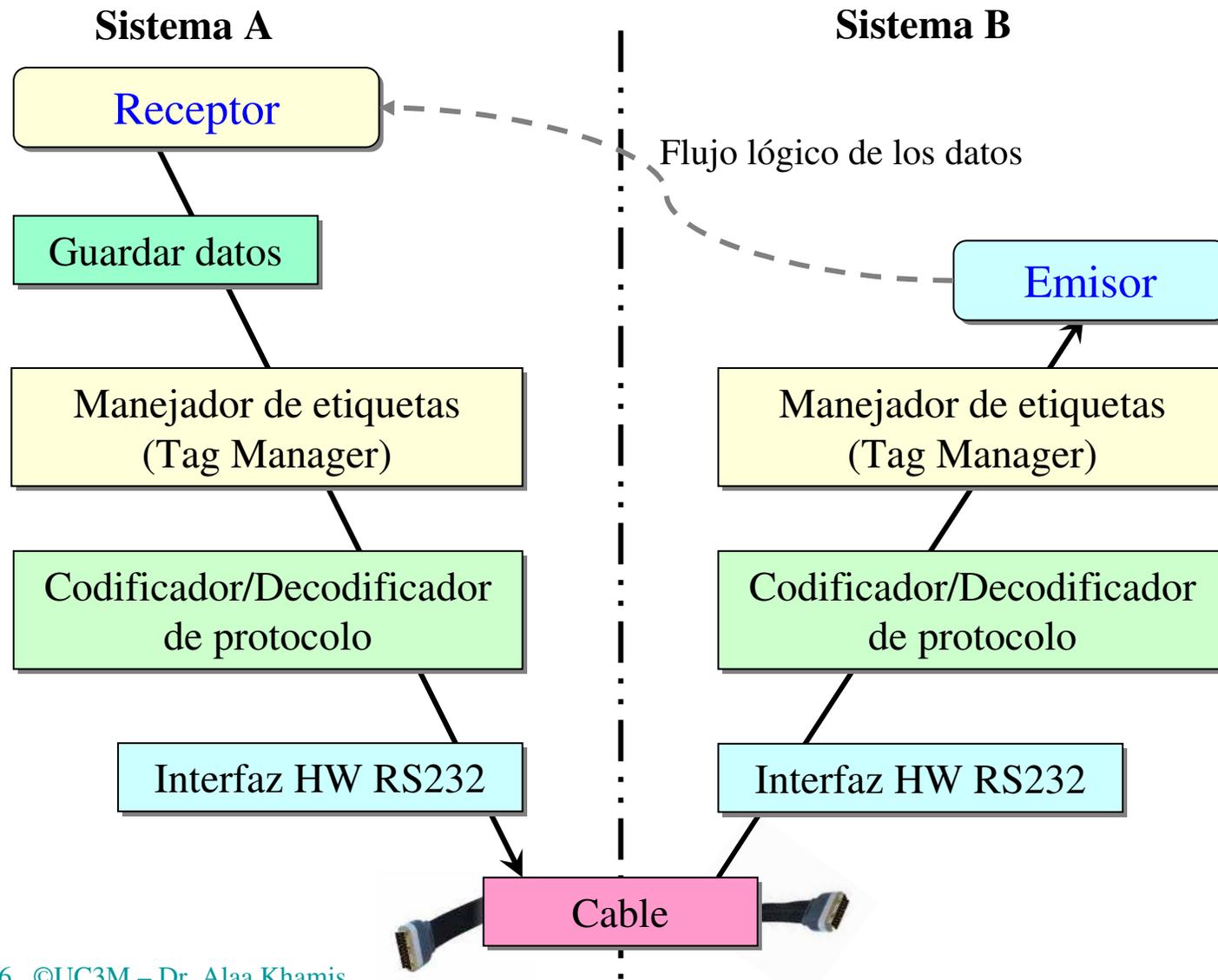
- ▶ Casi todos los sistemas soportan la comunicación serie utilizando los terminales **RS232C**.
- ▶ Estos terminales describen qué voltajes se usan para representar "0" "y 1".
- ▶ El emisor y el receptor tienen que utilizar un **protocolo común**.
- ▶ En un protocolo de comunicación, los datos se organizan de **manera predefinida** (marcos o Frames).



- ▶ Para garantizar la compatibilidad entre el emisor y el receptor, el contenido del bloque de datos tiene que seguir un protocolo común utilizando **etiquetas estándares** (Standard tags)



Terminales de comunicación serie RS232





Buses de campo (Fieldbus)

Los buses de campo (Fieldbus) se usan como sistema de comunicación entre los elementos de control industrial y los dispositivos de campo.

- **Ventajas**

- ▶ **Reducen el coste** de cableado, configuración y mantenimiento.
- ▶ Por el cable de comunicación del bus **se transmite toda la información** relevante en un proceso automatizado, los datos de entrada y salida de los captadores y accionadores, y también los datos de parametrización, datos de diagnóstico y programas de aplicación.
- ▶ Permiten la **comunicación con dispositivos inteligentes** mediante un cable de comunicación serie. Las **modificaciones y ampliaciones** de las instalaciones se pueden realizar **fácilmente** con sólo extender el cable del bus y conectar los nuevos dispositivos.





Buses de campo (Fieldbus)

- **Características de los buses de campo**
 - ▶ **Medio físico**
 - ▶ **Topología**
 - ▶ **Tipo de transmisión**
 - ▶ **Método de acceso al medio**
 - ▶ **Velocidad de transferencia**
 - ▶ **Distancias medias**
 - ▶ **Número de equipos de proceso conectables**





Buses de campo (Fieldbus)

● Características de los buses de campo

▶ Restricciones del entorno

- Alimentación de los equipos conectados
- Número de sensores y actuadores
- Distancias
- Perturbaciones electromagnéticas
- Elementos móviles
- Estanqueidad
- Entorno hostil (sal, agua, ácido, ...)
- Entorno explosivo.

▶ Restricciones de tiempo

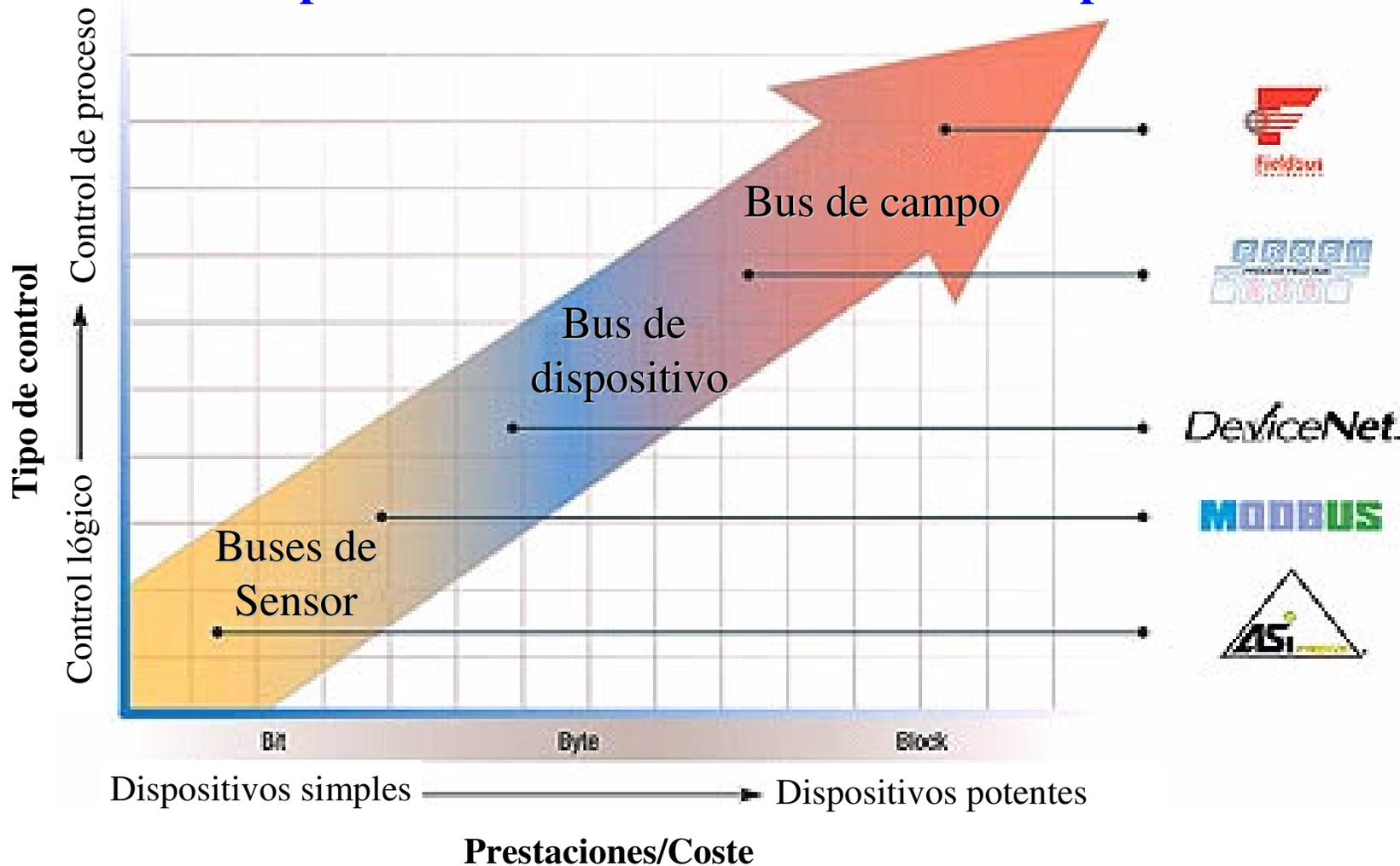
- Tiempo máximo de respuesta, en función del número de E/S
- Datos necesarios para la aplicación (velocidad de transferencia)
- Determinístico: tiempo máximo de respuesta conocido.





Buses de campo (Fieldbus)

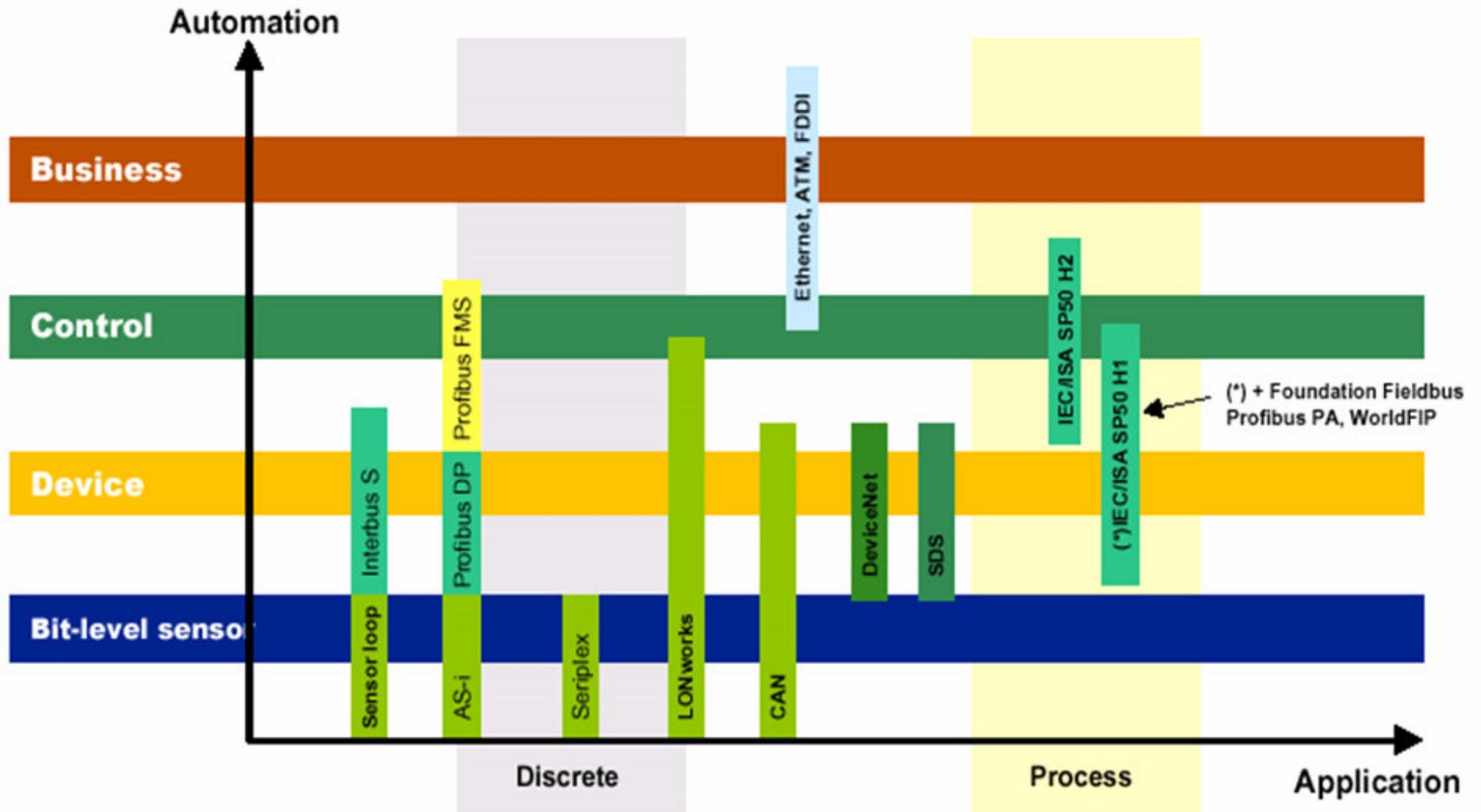
Diferentes posibilidades de conexionado en campo



Buses de Sensor (AS-I, LonWorks, Seriplex)
 Bus de dispositivo (CAN, ControlNet, DeviceNet, LonWorks, Profibus DP, Interbus)
 Bus de campo (Fieldbus, WorldFIP, Profibus PA)



Buses de campo (Fieldbus)





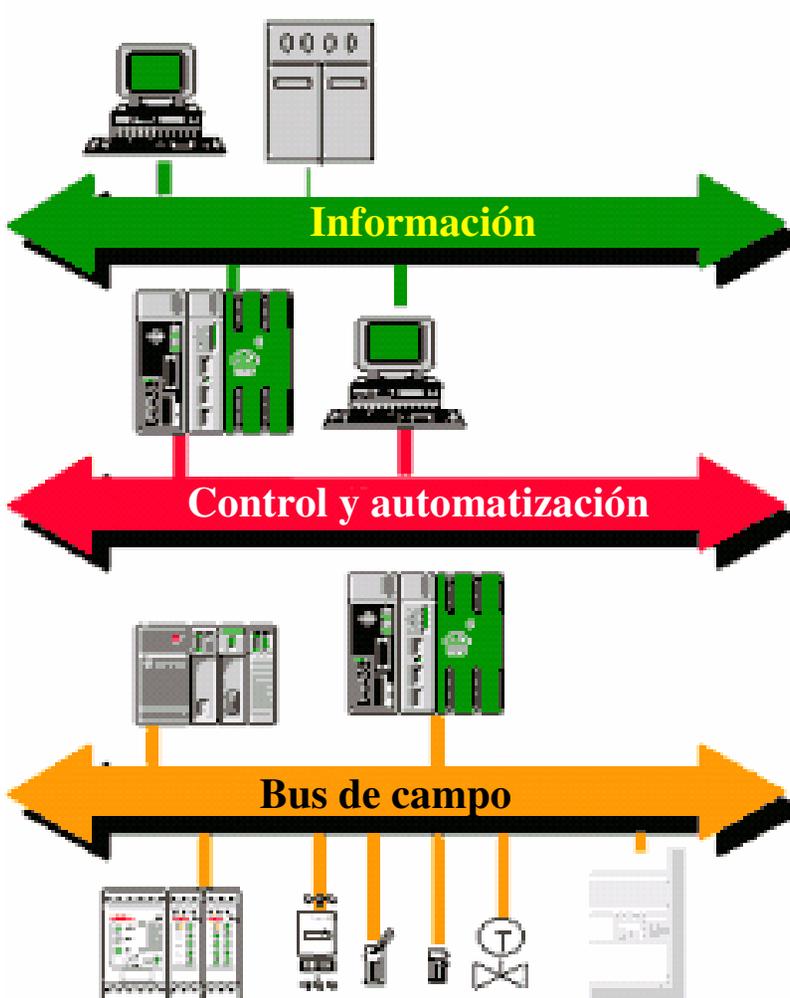
Buses de campo (Fieldbus)

- Conexionado en campo





Buses de campo (Fieldbus)

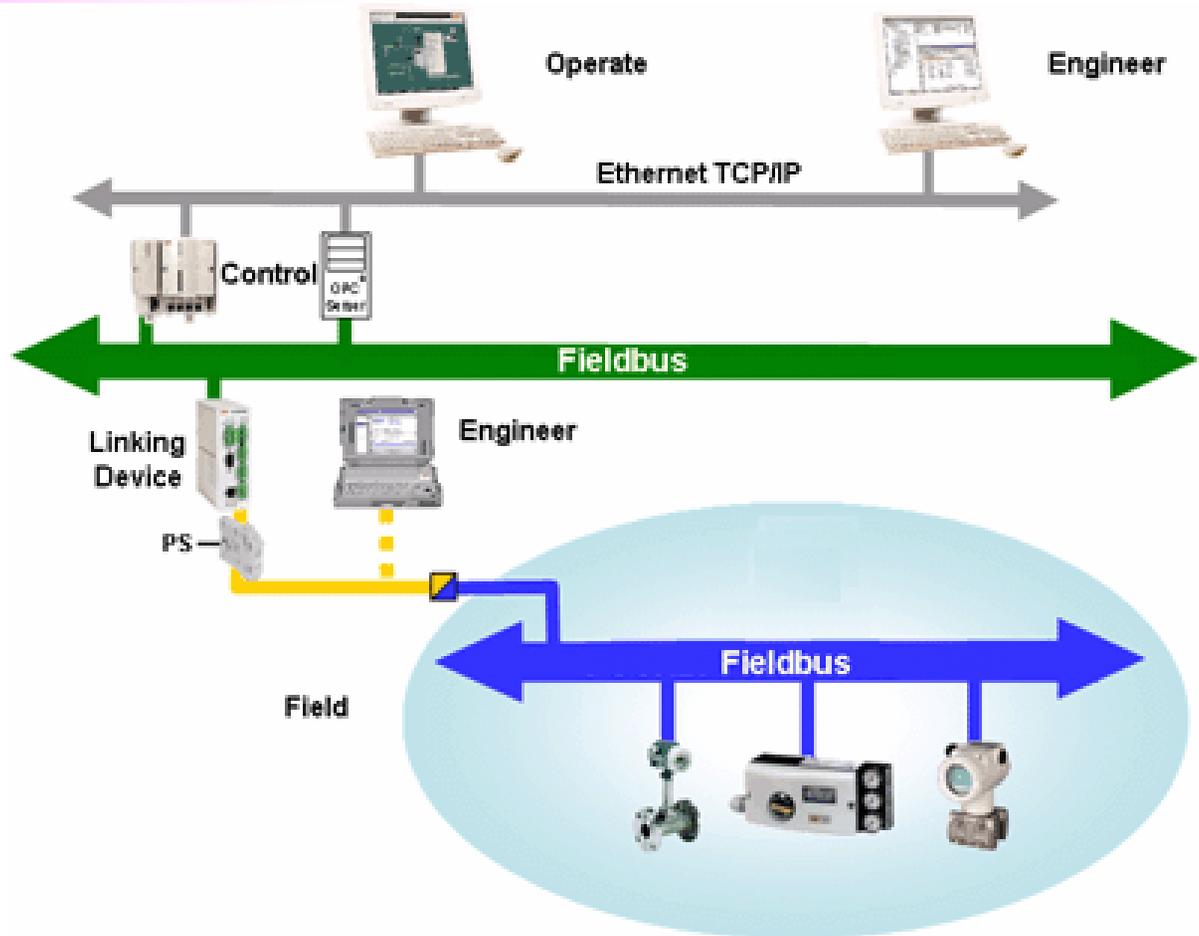


Aplicación	Tamaño de la red	Cantidad de datos	Tiempo de reacción	Coste
LAN, WAN	Grande	Muy Grande	1 s	elevado
Control + automatiz.	Medio	Grande	100 ms	Bajo
Entradas/ Salidas Inteligentes distribuidas	Pequeño	Media	10 ms	Bajo



Buses de campo (Fieldbus)

- ▶ AS-Interface
- ▶ CAN
- ▶ DeviceNet
- ▶ FOUNDATION fieldbus
- ▶ HART Protocol
- ▶ Ethernet Industrial
- ▶ LonWorks
- ▶ modbus
- ▶ Netbus
- ▶ Profibus





Buses de campo (Fieldbus)

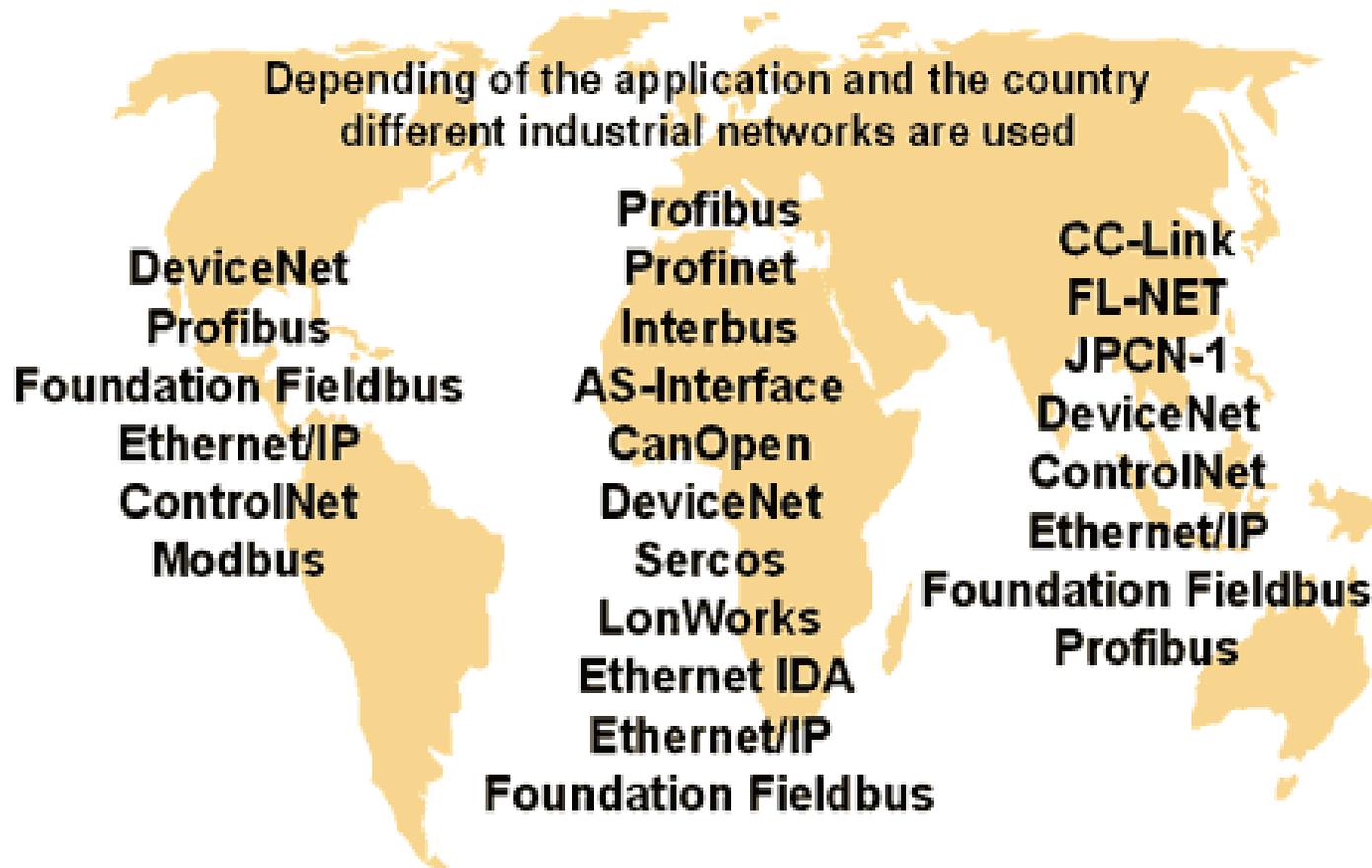
- **Situación actual de los buses de campo**

Sector	Buses de campo principales en el sector
Fabricante de automóviles y maquinaria	CAN, DeviceNet, Interbus-S, AS-I
Plantas de proceso	Foundation Fieldbus, Modbus, WorldFIP, Profibus
Domótica	LonWorks



Buses de campo (Fieldbus)

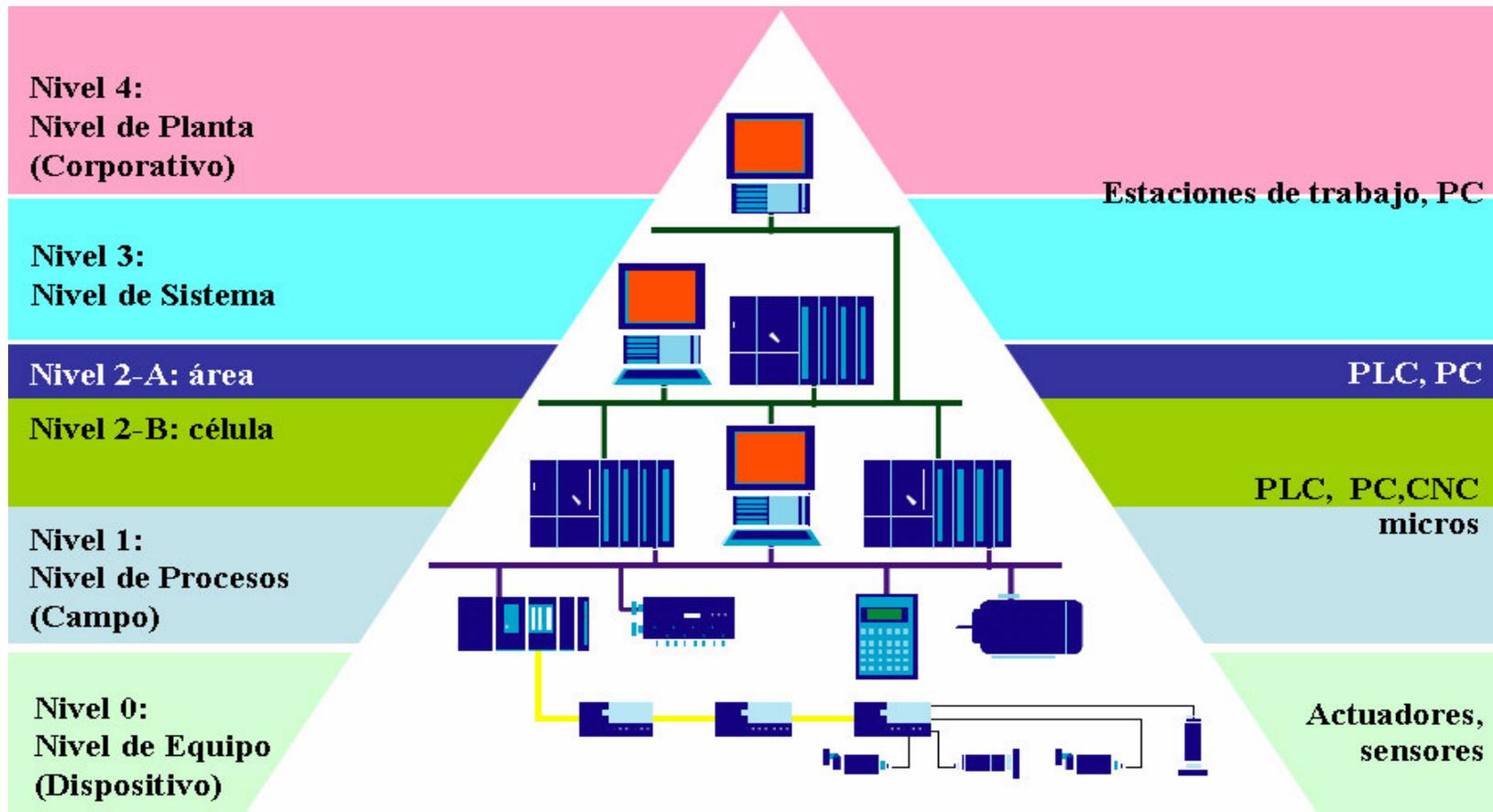
- **Situación actual de las redes industriales**





Buses de campo (Fieldbus)

● Pirámide de CIM



- Buses de campo (Actuador-Sensor)
- Buses de campo (Dispositivo)

- Buses de campo (Célula)
- Buses de campo (Dispositivo/Célula)



Buses de campo (Actuador-Sensor)

● **AS-interface**

- ▶ Iniciativa europea
- ▶ Subbus de muy bajo coste, bus de transductores.
- ▶ Reduce problemas de conexionado, diagnósticos a nivel de dispositivo.
- ▶ Pasarelas de interfaz con Profibus, DeviceNet, CAN, Interbus-S y WorldFIP,
- ▶ **Aplicaciones:**
 - Sistema de bajo coste
 - Sistema de que funcione de manera fiable y segura incluso en las condiciones más adversas
 - Sistema de que pueda trabajar en tiempo real
 - Sistema de aplicación universal
 - Sistema de instalación sencilla y flexible
 - Sistema de ampliación rápida y flexible.



<http://www.as-interface.net/>

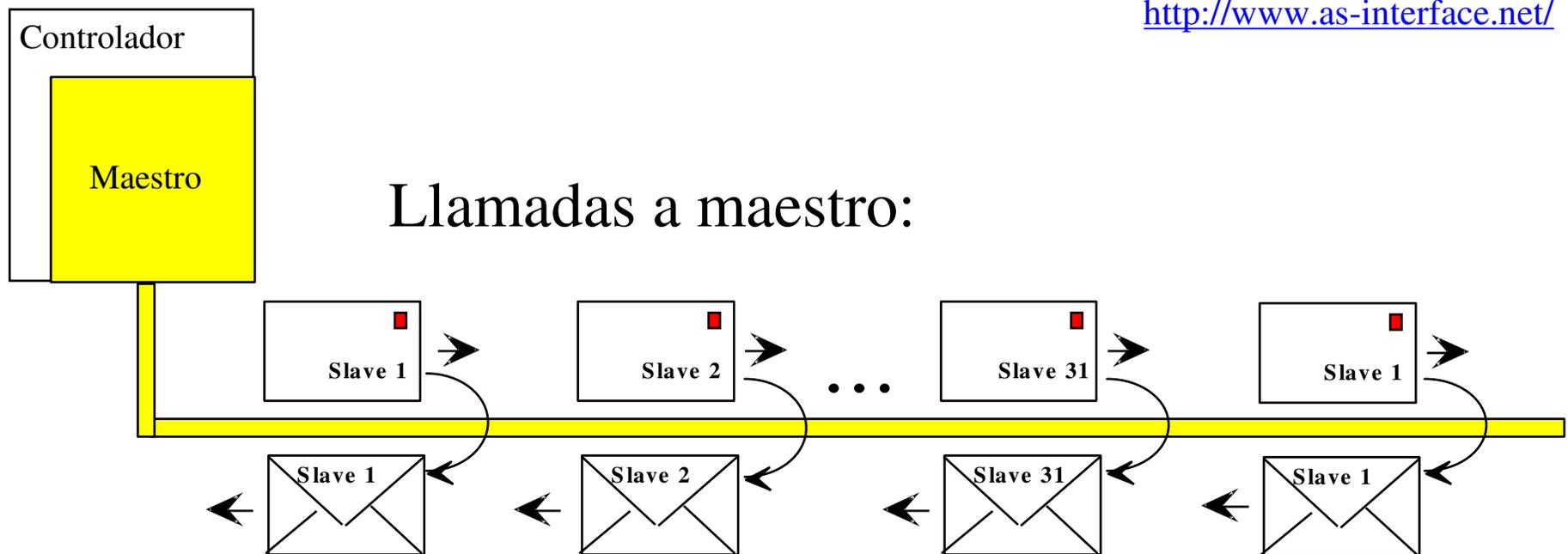


Buses de campo (Actuador-Sensor)

- AS-interface: Funcionamiento



<http://www.as-interface.net/>



Para el usuario: sin programación; sin configuración - ¡¡basta con la dirección!!



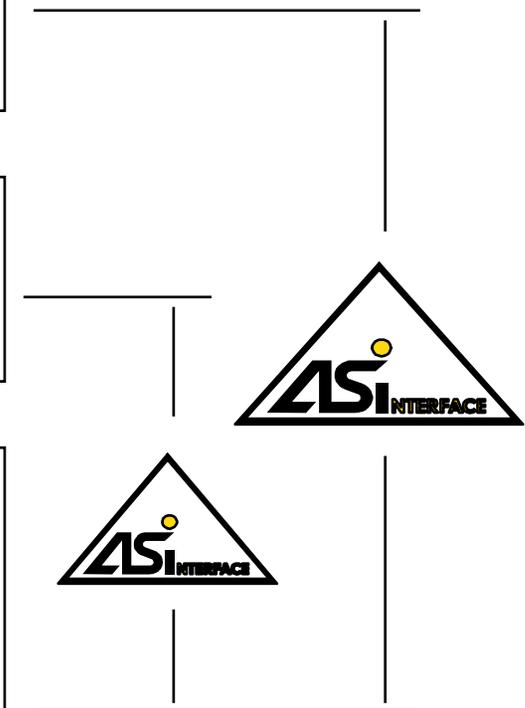
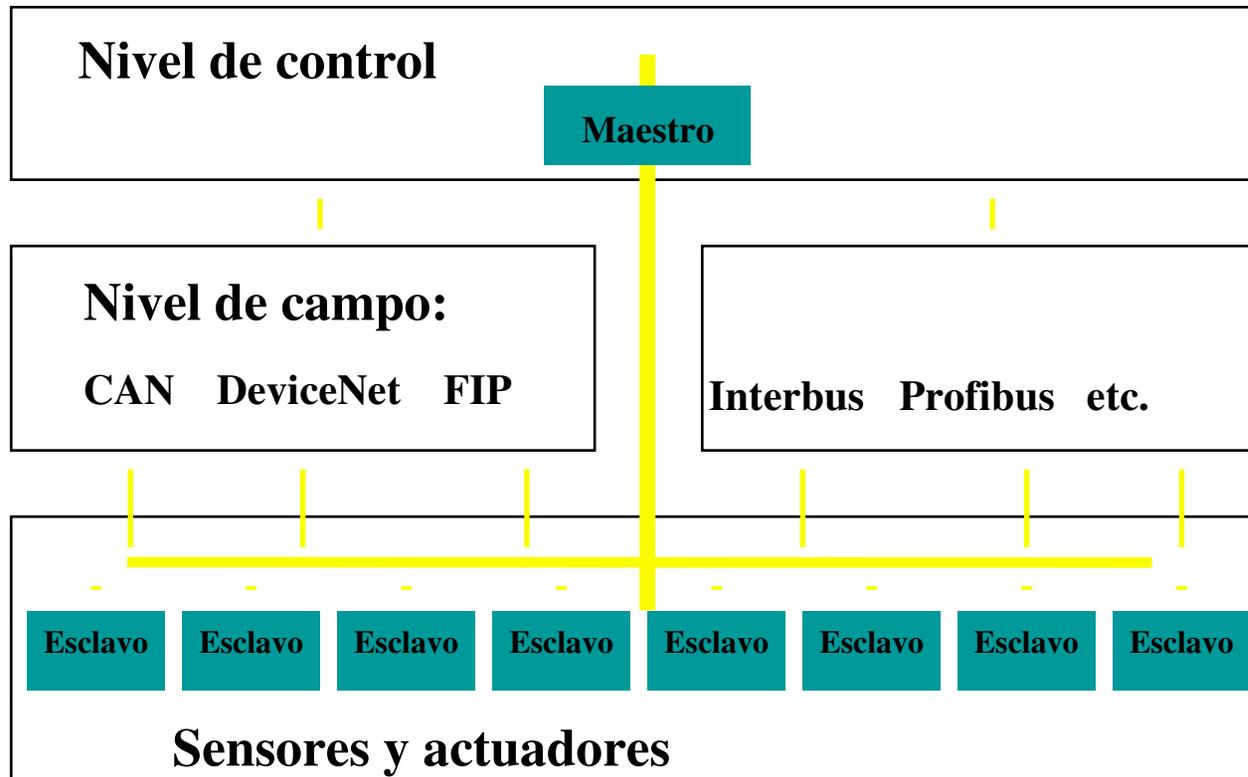
Buses de campo (Actuador-Sensor)

- **AS-interface**

- ▶ **AS-Interface en la jerarquía de automatización**



<http://www.as-interface.net/>





Buses de campo (Actuador-Sensor)



● **AS-interface: Principales datos**

- Principio maestro-esclavo
- Hasta 31 esclavos acoplados a un mismo cable
- Hasta 4 entradas +4 salidas por esclavo (hasta 248 entradas y salidas binarias por cada red)
- Además 4 bits de parámetros
- Son posibles E/S analógicas
- Configuración electrónica de la dirección a través del acoplamiento a bus
- Cable bifilar sin apantallar
- Información y alimentación eléctrica a través de un mismo cable
- Longitud de cable 100 m (300 m con repetidores/prolongadores)
- Estructura de árbol libre de la red
- Grado de protección hasta IP67
- Tiempo de ciclo < 5 ms
- Protección contra errores muy eficaz



Buses de campo (Actuador-Sensor)

- **AS-interface: ¿Qué se puede ahorrar con el AS-Interface?**



Hardware

- Tarjetas E/S en el PLC, PC
- Armario eléctrico más pequeño
- Prensaestopas PG
- Conectores múltiples
- Canales de cables, traviesas
- Anillos rozantes/cables arrastrables
- Cajas de bornes
- Bornes de organización
- Cables/cableado

Esfuerzos/costes

- Tiempo de instalación más corto
- Tiempo de inspección más corto
- Puesta en servicio más rápida
- Menos tiempo para la elaboración de esquemas
- Menos esfuerzos para identificación de cables y bornes
- Documentación más sencilla
- Diagnóstico más rápido
- Tiempos de mantenimiento más cortos



Buses de campo (Actuador-Sensor)



- **AS-interface: Beneficios adicionales...**
 - Prevención de errores de cableado
 - Ampliación sin modificación del armario eléctrico
 - Ampliación en cualquier punto de la red
 - Instalación, en parte, posible por personas sin conocimientos eléctricos
 - Preelaboración de la instalación en fábrica en vez de en el lugar de obra

¡Inicio avanzado de la producción, menos capital muerto!



Buses de campo (Actuador-Sensor)

● **CAN: Controller Area Network**

- ▶ Impulsado por Bosh (Alemania) y CiA (CAN in Automation)
- ▶ Recogido en la norma ISO 11898/11519
- ▶ Aplicación en la industria del automóvil
- ▶ Se caracteriza por su robustez.
- ▶ Medio físico de conexión: par trenzado
- ▶ Velocidad de transmisión: 50 Kbit/s a 1 Mbit/s
- ▶ Distancia segmento: 40- 1000 m
- ▶ Nodos por segmento: 127-64
- ▶ Topología: bus lineal
- ▶ Acceso al bus: CSMA/CD con arbitraje de bit



<http://www.can-cia.org/>



www.can.bosch.com/



Buses de campo (Actuador-Sensor)

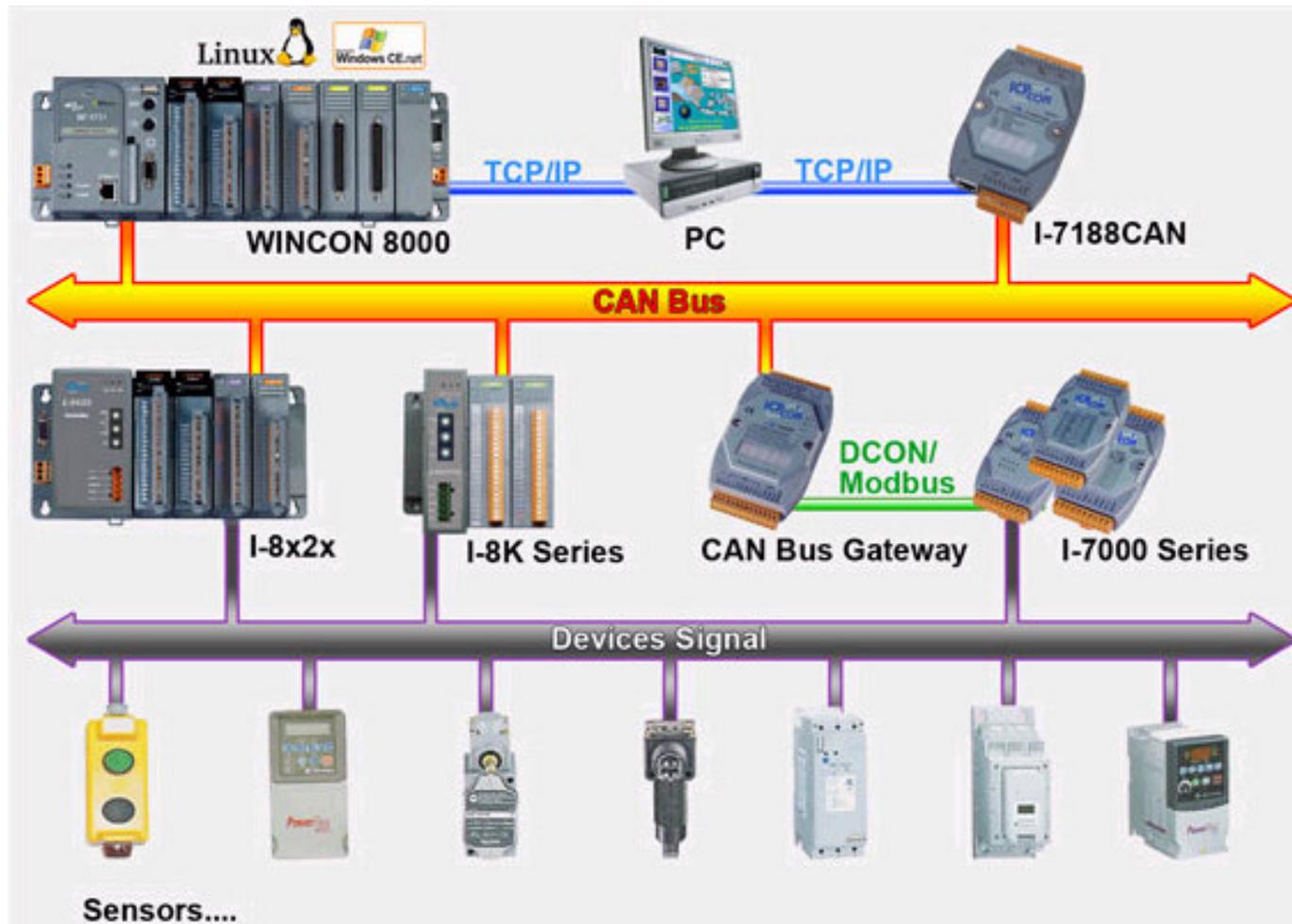
- **CAN: Controller Area Network**



<http://www.can-cia.org/>



www.can.bosch.com/



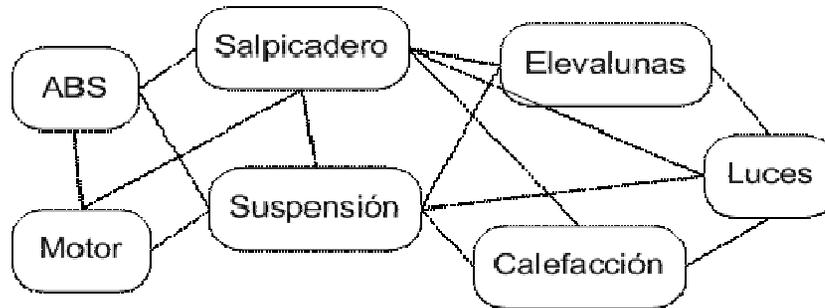


Buses de campo (Actuador-Sensor)

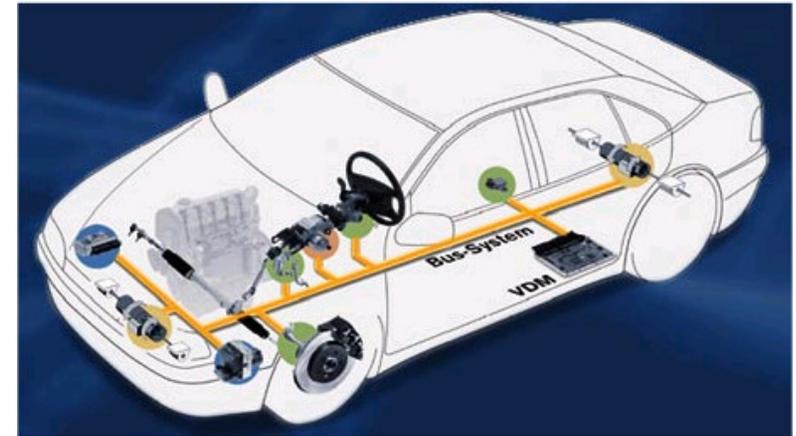
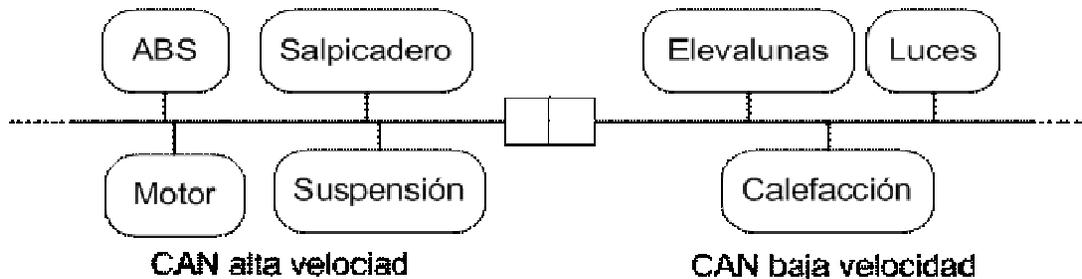
- CAN en los automóviles**

El uso de CAN en vehículos permite a cualquier estación comunicarse con cualquier otra sin que suponga gran carga adicional para el controlador.

Cableado tradicional



CAN





Buses de campo (Actuador-Sensor)

LONWORKS®

<http://www.ieclon.com/LonWorks/>

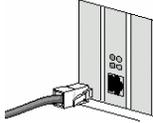
● **LonWorks: Local Operating Network**

- ▶ Fuertemente asentado en EE.UU., introduciéndose en Japón
- ▶ Gestión técnica de edificios
- ▶ Basado en control distribuido
- ▶ Extremada facilidad de utilización
- ▶ Motorola es el principal proveedor de circuitos integrados de interfase
- ▶ Modelo OSI completo
- ▶ Medio físico de conexión: par trenzado, fibra óptica, red eléctrica, coaxial, radio, infrarrojos
- ▶ Velocidad de transmisión: 78 Kbit/s a 1,25 Mbit/s
- ▶ Distancia segmento: 2700-130 m
- ▶ Nodos por segmento: 64
- ▶ Topología: bus, anillo, libre
- ▶ Acceso al bus: Predictive p-persistent CSMA



Buses de campo (Dispositivo)

● **DeviceNet**

- ▶ Impulsado por Allen Bradley (Rockwell)
- ▶ Recogido en la norma ISO 11898 y 11519 (basado en CAN)
- ▶ Mismo cable para alimentación y datos
- ▶ Longitud máxima de datos por trama de 8 octetos
- ▶ Medio físico de conexión: par trenzado 
- ▶ Velocidad de transmisión: 125, 250, 500 Kbit/s
- ▶ Distancia segmento: hasta 500, 250, 100 m
- ▶ Nodos por segmento: 64
- ▶ Topología: bus lineal
- ▶ Acceso al bus: CSMA/CD



<http://www.ab.com/>

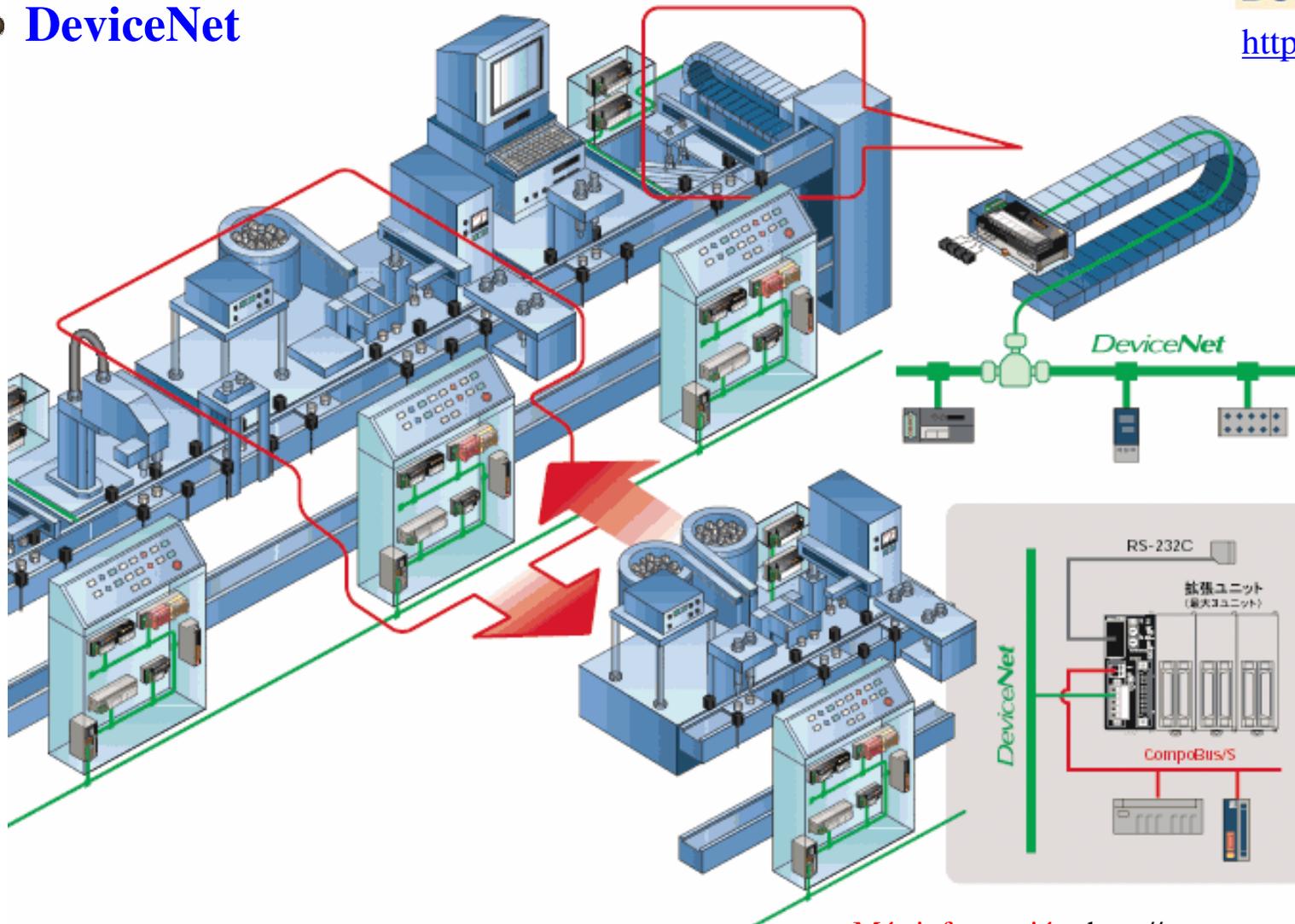


Buses de campo (Dispositivo)

- DeviceNet



<http://www.ab.com/>





Buses de campo (Dispositivo)



<http://www.phoenixcontact.com/>

● Interbus-S

- ▶ Impulsado por Phoenix Contact (Alemania), recogido en la norma DIN E19258 y EN 50254.
- ▶ Bus serie conecta módulos E/S distribuidas y dispositivos de automatización.
- ▶ Maestro/esclavo
- ▶ Longitud de mensaje fija.
- ▶ Longitud total 13 Km
- ▶ Aplicaciones: sensor/actuador, sistemas de producción, ...
- ▶ Medio físico de conexión: par trenzado
- ▶ Velocidad de transmisión: 500 Kbit/s
- ▶ Distancia segmento: hasta 400 m
- ▶ Nodos por segmento: 256
- ▶ Topología: anillo controlado por dispositivo central o tarjeta de control
- ▶ Acceso al bus: Paso de testigo





Buses de campo (Célula)



<http://www.modicon.com/>

● Modbus

- ▶ Diseñado por MODICON INC.
- ▶ Aparece para facilitar la conexión de forma distribuida de PLCs.
- ▶ Usado principalmente en América
- ▶ Velocidad de transmisión: 75 a 19,2 Kbps
- ▶ Medio físico de conexión: RS-485, RS-422 o fibra óptica.
- ▶ Un maestro y hasta 247 esclavos
- ▶ Distancia: hasta 1200 m sin repetidores
- ▶ Interconecta dispositivos de campo: PLCs
- ▶ Protocolo de comunicaciones maestro-esclavo
- ▶ Topología: bus lineal

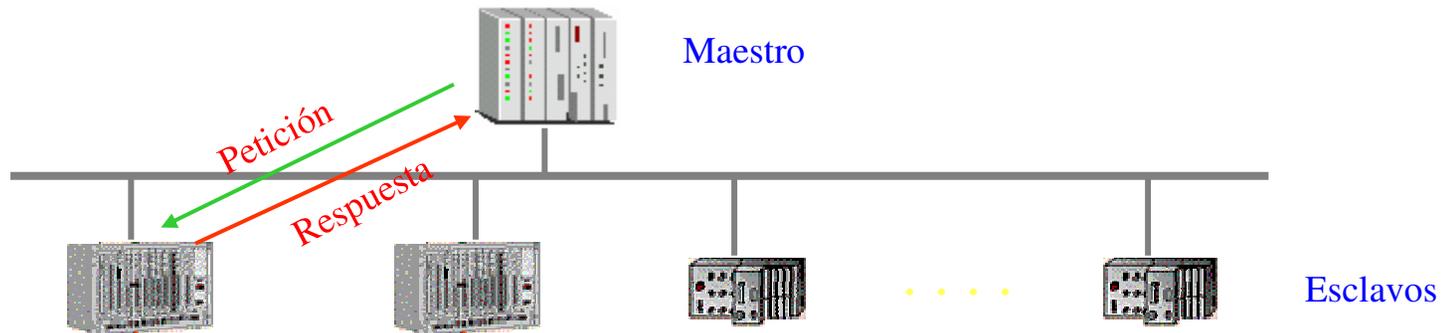




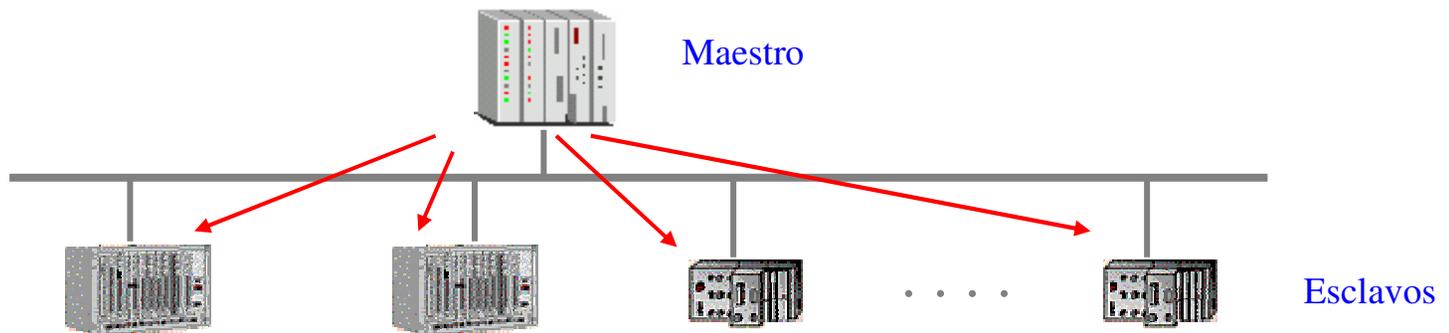
Buses de campo (Célula)

- **Modbus: Método de acceso al medio MODBUS**

Pregunta/Respuesta



Difusión





Buses de campo (Célula)

● **Modbus**

- ▶ Sólo el maestro puede iniciar la comunicación
- ▶ La comunicación se estructura en transacciones:
 - Pregunta/respuesta
 - Mensaje difundido
- ▶ El protocolo fija algunas características:
 - la forma del mensaje
 - secuencia de mensajes
 - gestión de errores
 - funciones a realizar
- ▶ Otras características son seleccionables por el usuario:
 - Medio de transmisión
 - Velocidad
 - Paridad
 - Número de bits de parada
 - Modo de transmisión de cada estación



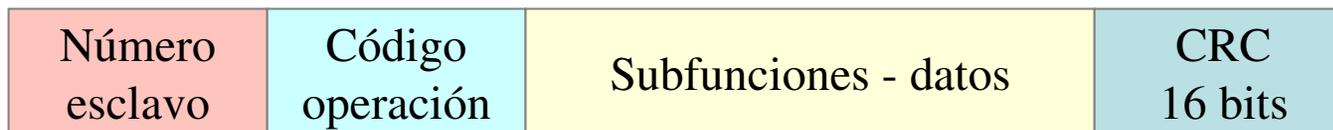
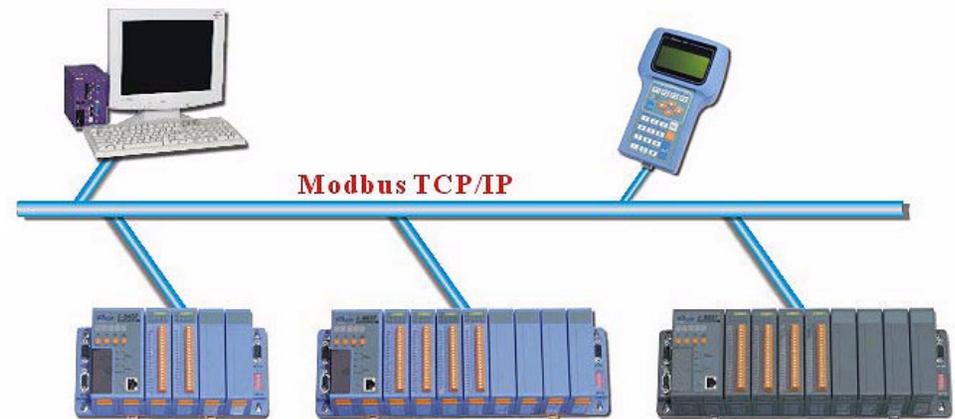
Buses de campo (Célula)

● Modbus: Trama de mensaje en MODBUS

▶ Mensaje dentro de un sobre:

- dirección del receptor
- función a realizar
- datos
- código de comprobación de errores

▶ Respuesta devuelta en el mismo sobre.



Más información: <http://www.automatas.org/modbus>



Buses de campo (Dispositivo/Célula)

WorldFiP

<http://www.worldfip.org/>

- **WorldFiP: World Factory Instrumentation Protocol**

- ▶ Impulsado por fabricantes franceses
- ▶ Recogido en la norma del CENELEC EN 50170 vol. 3
- ▶ Capa física según la norma IEC 1158-2
- ▶ Diseñado para establecer comunicaciones entre el nivel de sensores /actuadores y el nivel de unidades de proceso (PLC, controladores, ...)
- ▶ Medio físico de conexión: par trenzado apantallado
- ▶ Diseñado para establecer comunicaciones entre el nivel de sensores /actuadores y el nivel de unidades de proceso (PLC, controladores, ...)
- ▶ Distancia segmento: hasta 1900 m
- ▶ Nodos por segmento: 32
- ▶ Topología: bus lineal
- ▶ Acceso al bus: Centralizado (árbitro de bus)



Buses de campo (Dispositivo/Célula)



- **Profibus (Process Field Bus)**

- ▶ Es una red de altas prestaciones para niveles de campo y de célula.
- ▶ Impulsado por fabricantes alemanes en 1987.
- ▶ En 1991 estándar en Alemania DIN 19245.
- ▶ Norma europea EN 50170 en 1996.
- ▶ Amplia implantación en el mercado europeo.
- ▶ Profibus fue desarrollado para la automatización industrial por diferentes fabricantes y usuarios, organizados y es una norma europea EN 50 170.
- ▶ Implantada en el mercado (alrededor de 1.000.000 nodos instalados mundialmente, 41% del mercado de buses de campo en Europa, según Consultic).

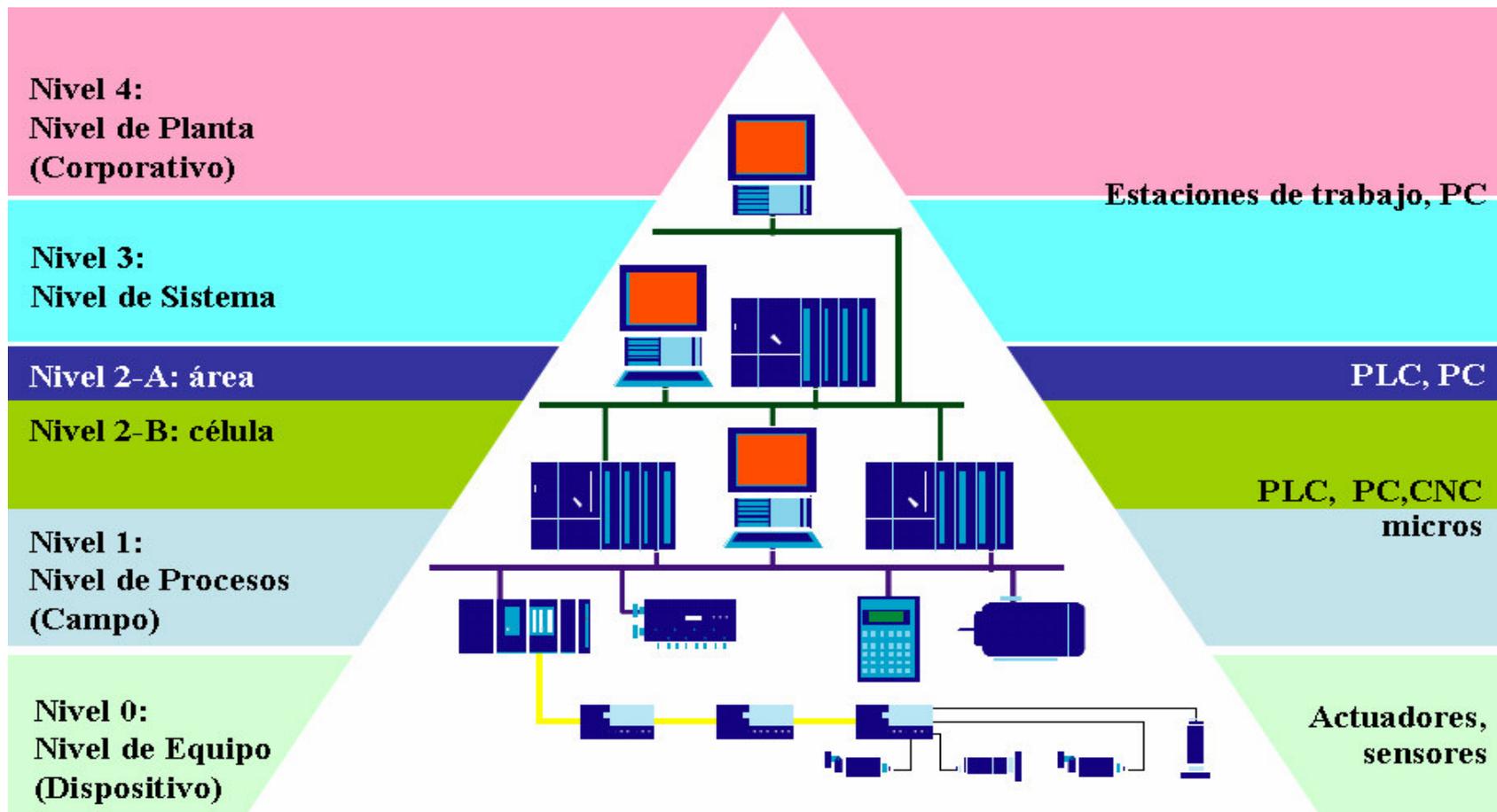


Buses de campo (Dispositivo/Célula)



- **Profibus (Process Field Bus)**

- ▶ Es una red abierta que opera en los niveles CIM 1 y 2-B.



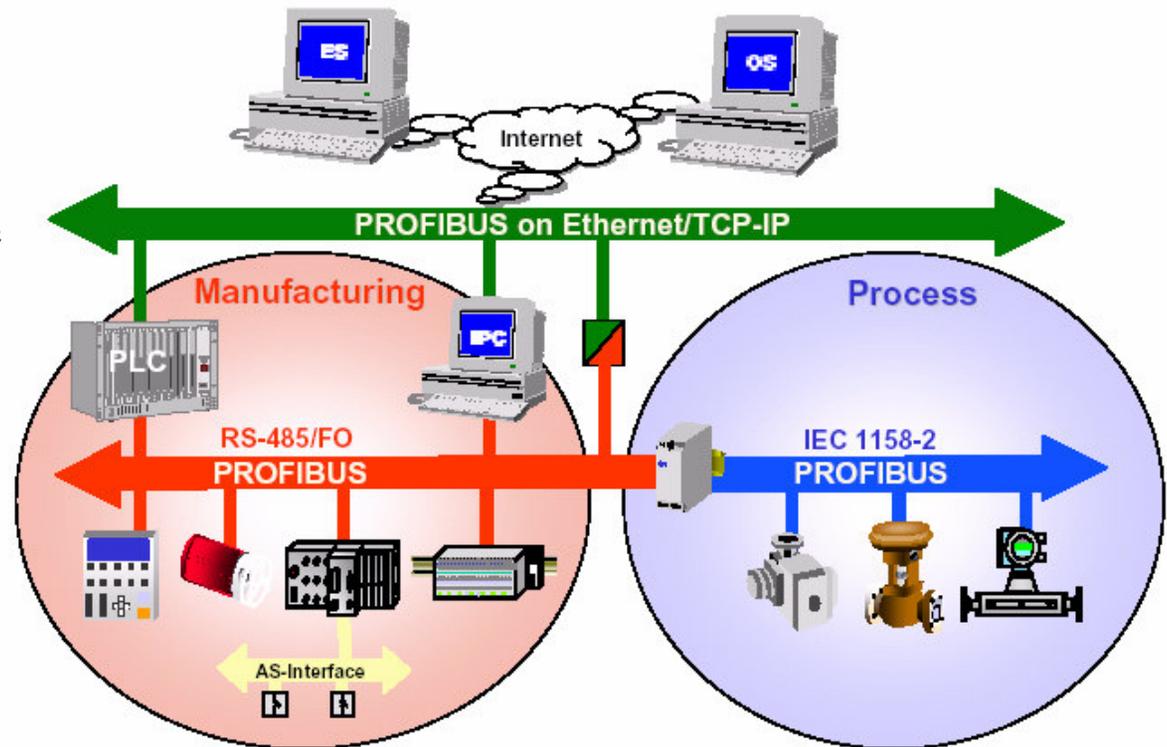


Buses de campo (Dispositivo/Célula)



● Profibus: Características

- ▶ Bus de campo serie
- ▶ Interconecta dispositivos de campo: sensores, actuadores, PLC, CN, etc.
- ▶ Dos tipos de estaciones:
 - Activas (master)
 - Pasivas (slave)
- ▶ Paso de testigo entre participantes activos
- ▶ Maestro-esclavo entre nodos asociados

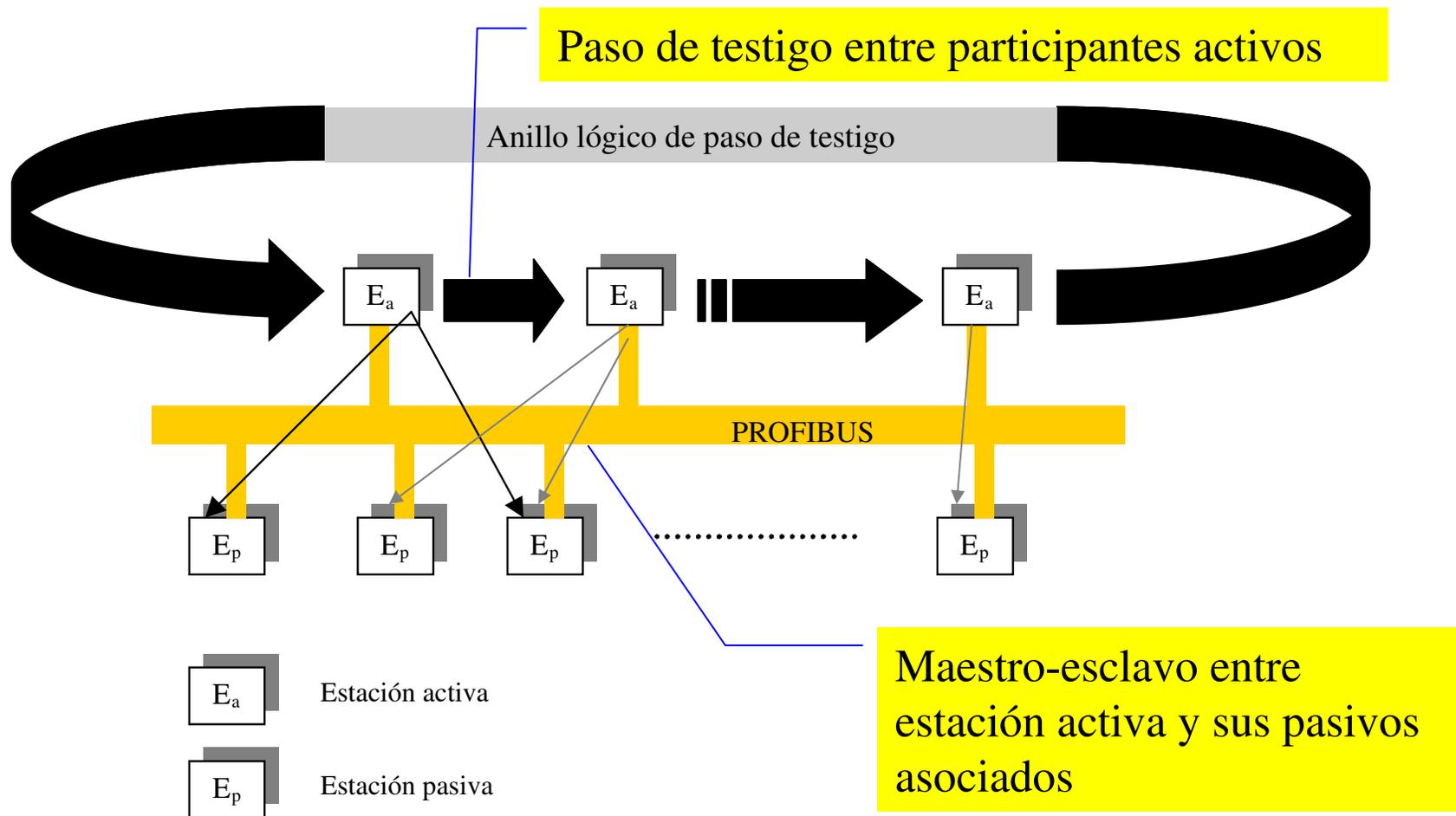




Buses de campo (Dispositivo/Célula)



- **Profibus: Método de acceso al medio Profibus**





Buses de campo (Dispositivo/Célula)

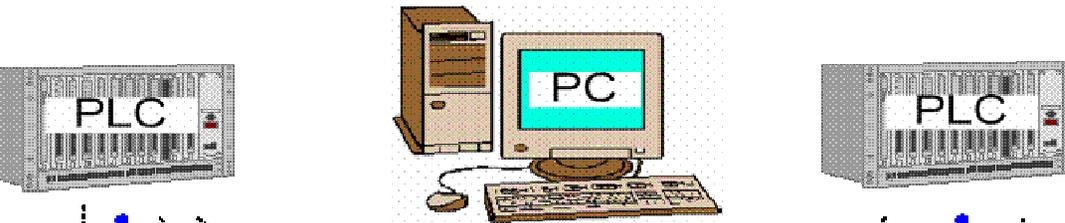


- **Profibus: Método de acceso al medio Profibus**

Anillo lógico entre dispositivos maestro



Estaciones activas, dispositivos maestro



Estaciones pasivas, dispositivos esclavo



Buses de campo (Dispositivo/Célula)



- **Profibus: Especificaciones Técnicas de Profibus**

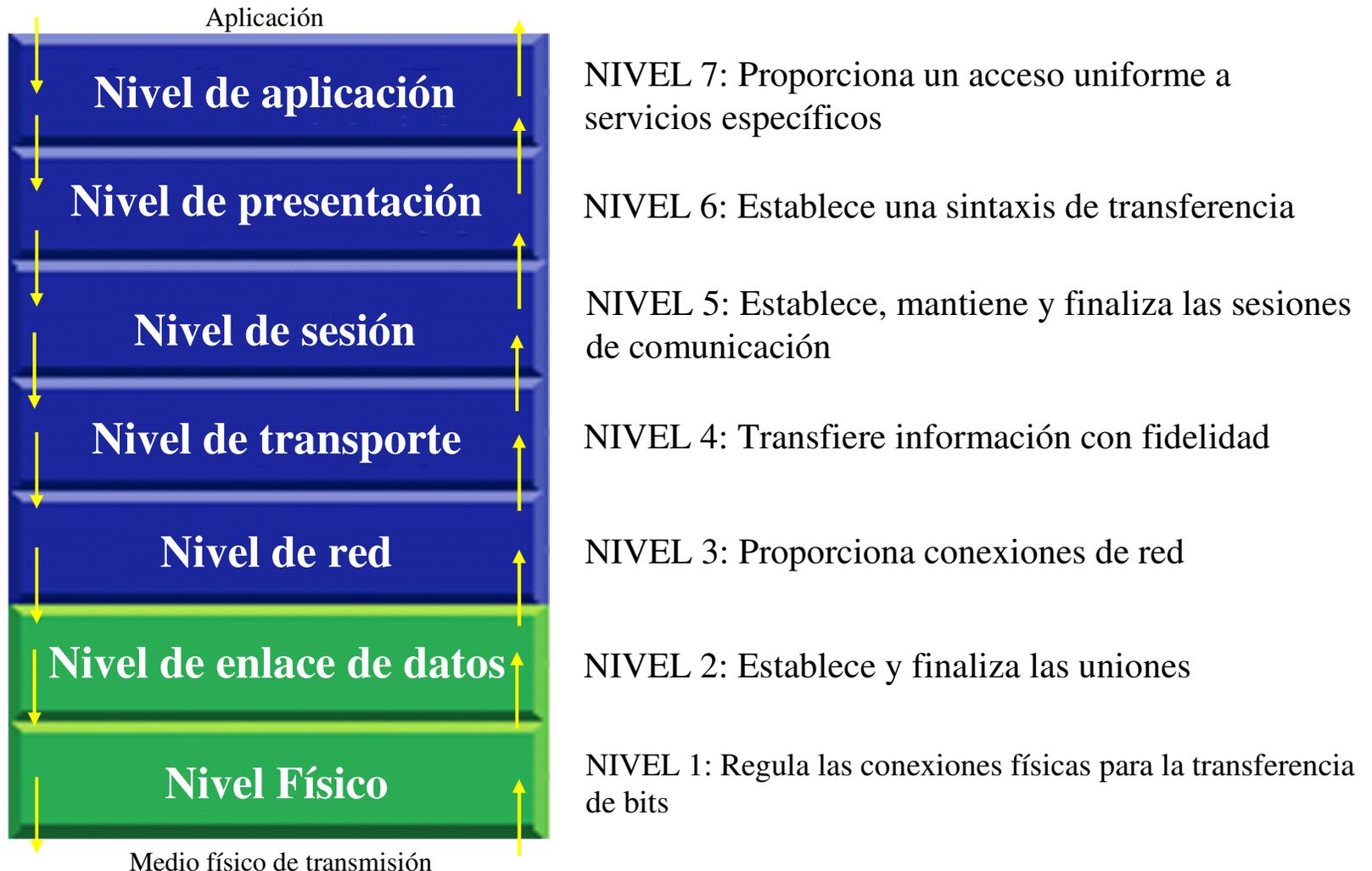
Estándar	PROFIBUS según EN 50170
Método de acceso	Paso por testigo junto con maestro-esclavo
Velocidad transfer.	9,6 Kbit/s - 12 Mbit/s
Medio transmisión	Eléctrico: par trenzado apantallado Óptico: fibra óptica
Distancia de red	Eléctrica: máx 9,6 Km Óptica: >100 Km
Máx nº nodos	127
Topología	Lineal, árbol, estrella, anillo
Aplicaciones	Niveles de campo y célula



Buses de campo (Dispositivo/Célula)



● Protocolos Profibus: Modelo de referencia ISO/OSI





Buses de campo (Dispositivo/Célula)



- Profibus: Protocolos



OSI Layer	Profibus Version		
	FMS	DP	PA
USER	FMS Device Profiles	DP Profiles	PA Profiles
		DP Functions	
7	FMS	Not Used	
6			
5	Not Used		
4			
3			
2	FDL		FDL
			IEC Interface
1	RS485 / Fiber Optic		IEC 1158-2



Buses de campo (Dispositivo/Célula)

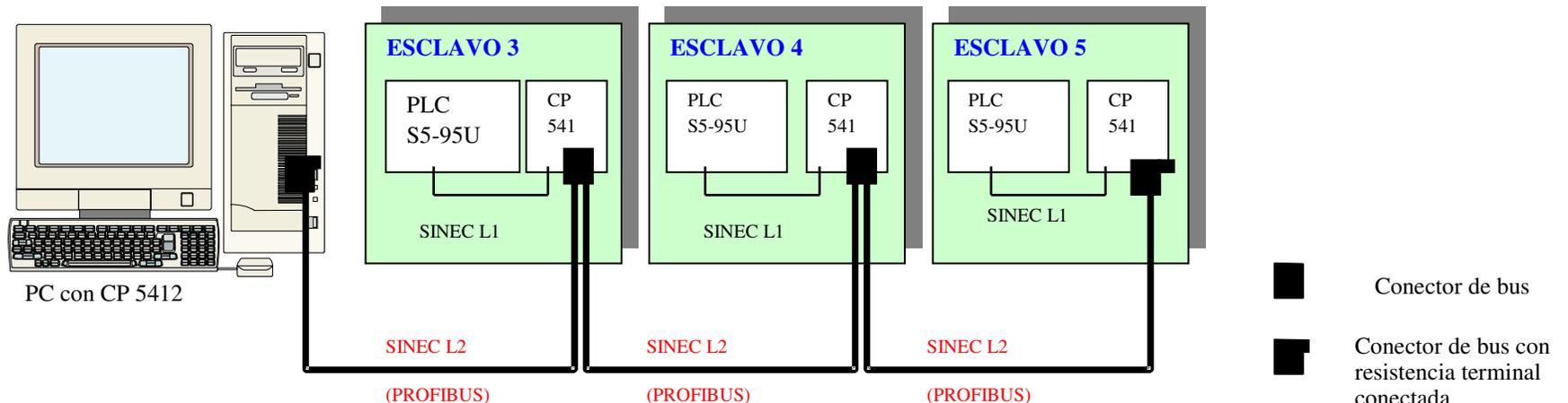


● Profibus: Protocolos

▶ PROFIBUS-DP (Distributed peripheral) o Periferia Distribuída

- Automatización de la fabricación
- Transferencia de datos a alta velocidad
- Optimizado para dispositivos de campo
- Asegura una transmisión de datos rápida (tiempos de reacción muy pequeños) y eficiente

MAESTRO DP





Buses de campo (Dispositivo/Célula)



- **Profibus: Protocolos**

- ▶ **PROFIBUS-PA (Process Automation)**

- Automatización de procesos en zonas restringidas
- Basado en PROFIBUS-DP
- Sistema de transmisión con seguridad intrínseca (IEC1158-2).

- ▶ **PROFIBUS-FMS (Fieldbus Message Specification)**

- Comunicación en redes de célula y con dispositivos de campo
- Comunicación universal . Orientada a objeto



Buses de campo (Dispositivo/Célula)



- **Profibus: Protocolos**

	PROFIBUS-DP	PROFIBUS PA	PROFIBUS-FMS
Aplicación	Área de campo	Área de campo	Área de célula
Estándar	EN 50 170	IEC 1158-2	EN 50 170
Dispositivos conectables	PLC, PC/disp. Prog., dispositivos de campo analógicos y digitales, controladores, válvulas, OPs	Dispositivos de campo analógicos y digitales PLC, PC/disp.prog./ OPs, controladores	PLC, PC/ disp. Prog.
T. Reacción	1 - 5 ms	Menos de 60 ms	Menos de 60 ms
Extensión	Sobre 100 km	Max. 1.9 km	Sobre 100 km
Velocidad de transmisión	9.6 Kbit/s 12 Mbit/s	31.25 Kbit/ s	9.6 Kbit/s 12 Mbit/s

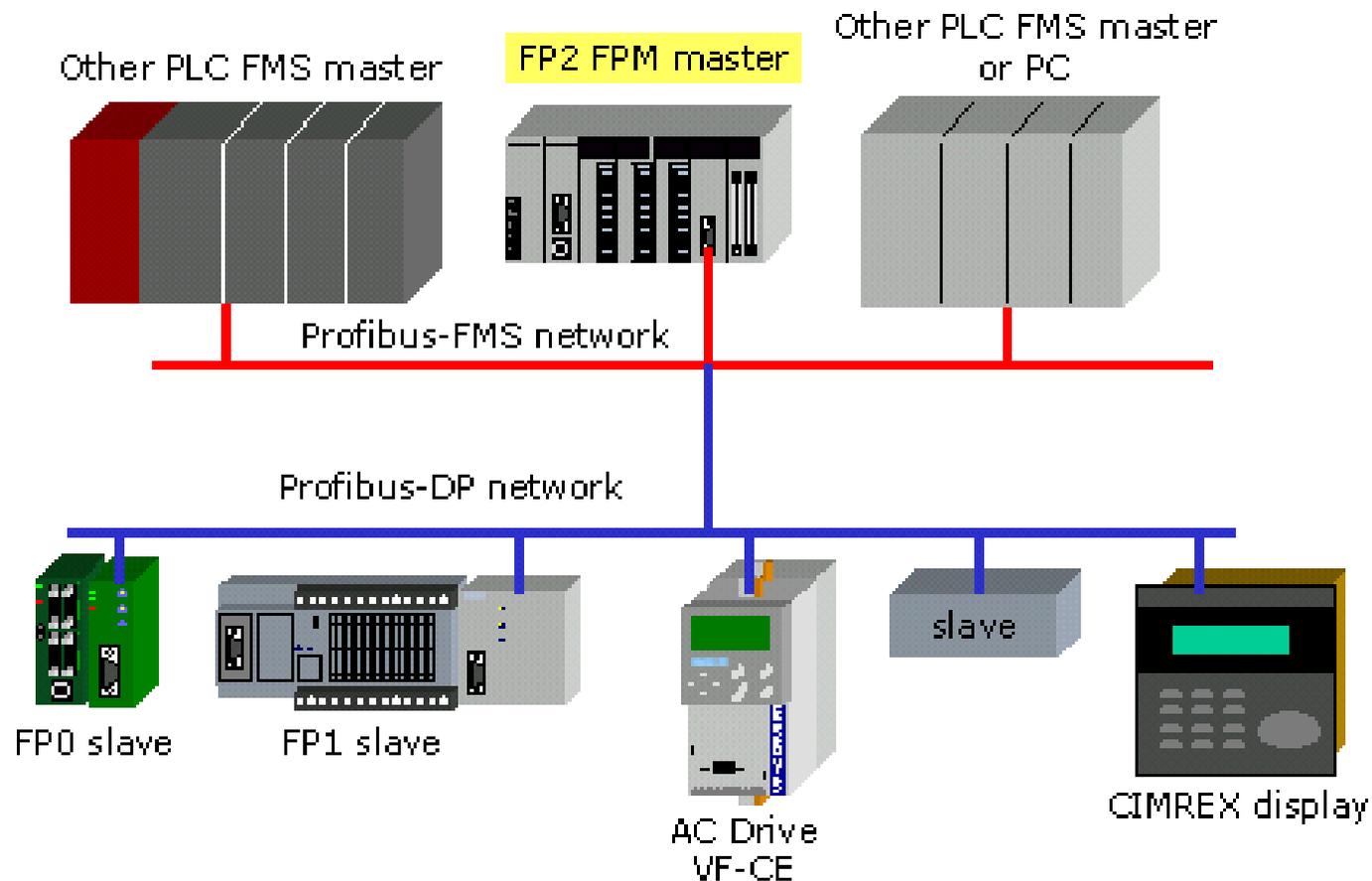


Buses de campo (Dispositivo/Célula)



- **Profibus: Protocolos**

FPM / DP mixed network





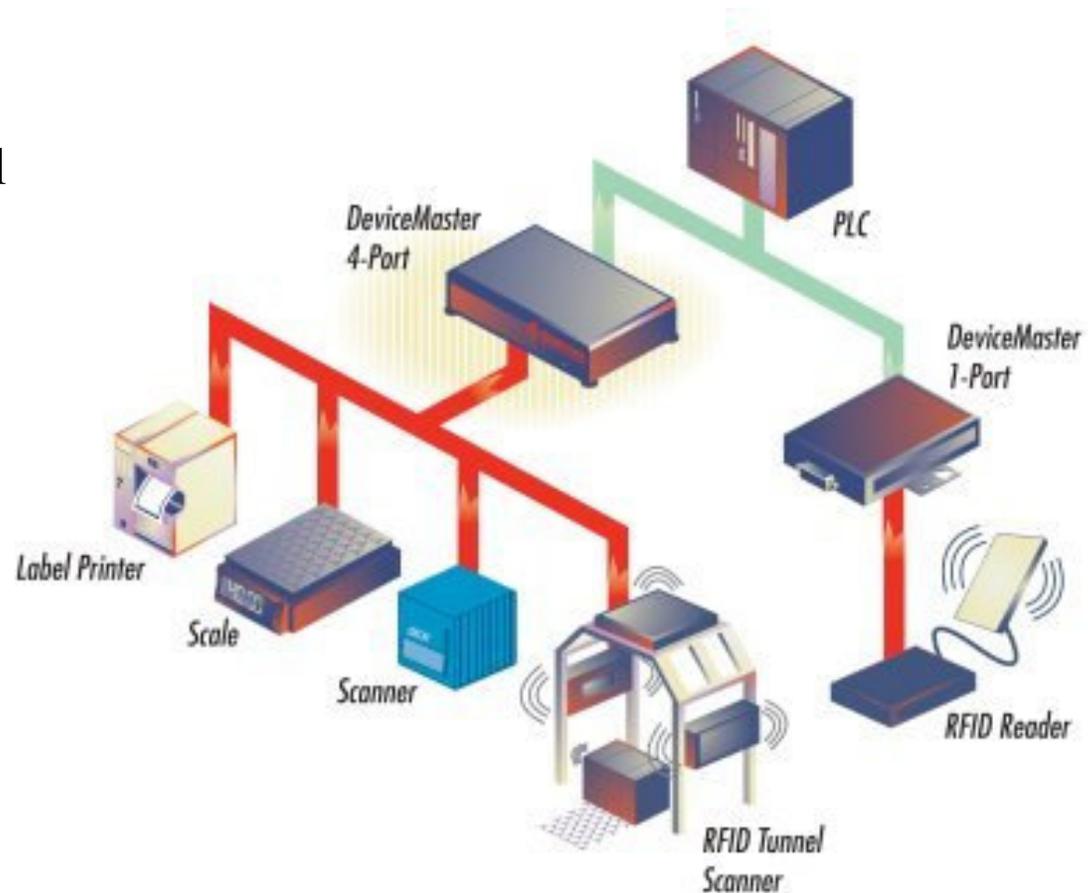
Ethernet Industrial



<http://us.profibus.com/profinet.aspx>

- **ProfiNet IO**

Un conjunto de protocolos de comunicación industrial definido según las normas IEC 61158 y IEC 61784 para empleo en sistemas de automatización.



Más información: <http://www.rtaautomation.com/profinetio/>

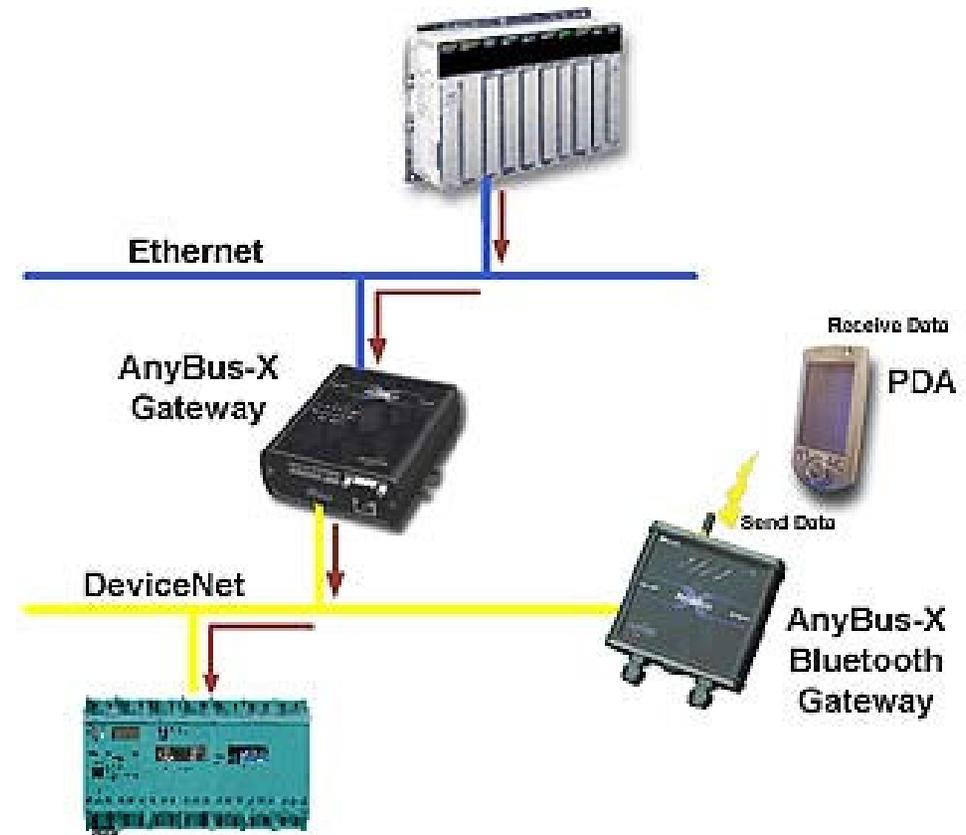


Ethernet Industrial

- **EtherNet/IP**

Es un protocolo de red en niveles para aplicaciones de automatización industrial.

Basado en los protocolos estándar TCP/IP, utiliza los ya bastante conocidos hardware y software Ethernet para establecer un nivel de protocolo para configurar, acceder y controlar dispositivos de automatización industrial.



Más información: http://www.siemon.com/la/white_papers/03-10-13-ethernet-ip.asp



Resumen:





Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 11 – Miércoles 24 de Mayo 2006

Buses Industriales

Ref:

- Los apuntes del Profesor: José Manuel Pastor García, “Comunicaciones en Fabricación”, Universidad Carlos III de Madrid.
- Carlo Cloet , “Ethernet for Industrial Automation: Have Fieldbus wars moved to a new battlefield?”, EE290-O Presentation.
- R. Moreno. Ingeniería de la Automatización Industrial. 2º Edición, Ra-Ma, 2004.



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 12 – Jueves 25 de Mayo 2006

Computer Integrated Manufacturing

Manufactura Integrada por Computador

CIM - Introducción

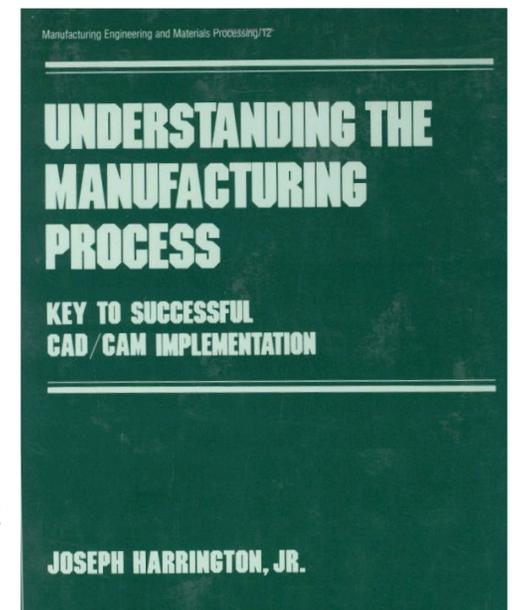


Significado y origen de CIM

- ▶ El acrónimo **CIM**, se ha sido utilizado para significar:

“La integración del negocio, ingeniería, manufactura y administración de la información, de las funciones de una compañía, desde el mercado hasta la distribución de sus productos”.

- ▶ Joseph Harrington, es acreditado como el primero en utilizar la expresión “**C**omputer **I**ntegrated **M**anufacturing”, 1973.
- ▶ La manufactura es considerada como el corazón de **CIM**. Así, Harrington considera en su libro tres elementos principales:
 - Procesos de planta
 - La planificación de la ingeniería de manufactura de esos procesos
 - La planificación y control de la producción en los procesos y los materiales utilizados





Significado y origen de CIM

- ▶ Otros autores tratan de capturar la esencia de CIM mediante el acrónimo **CIE** “**C**omputer **I**ntegrated **E**nterprise”.
- ▶ Algunos usan el término **CIME** “**C**omputer **I**ntegrated **M**anufacturing **E**nterprise”.
- ▶ La Comisión Europea promotora del programa SPRIT, utiliza el acrónimo **CIME** “**C**omputer **I**ntegrated **M**anufacturing and **E**ngineering”.

Ninguno de estos acrónimos alternativos se ha establecido, por si mismo, como lo está **CIM**.

- ▶ El concepto de Integración total de todas las funciones de una industria, está parcialmente recogido en 1985 por:

La rueda CIM (**CIM Wheel**) de Society of Manufacturing Engineers,



<http://www.sme.org>



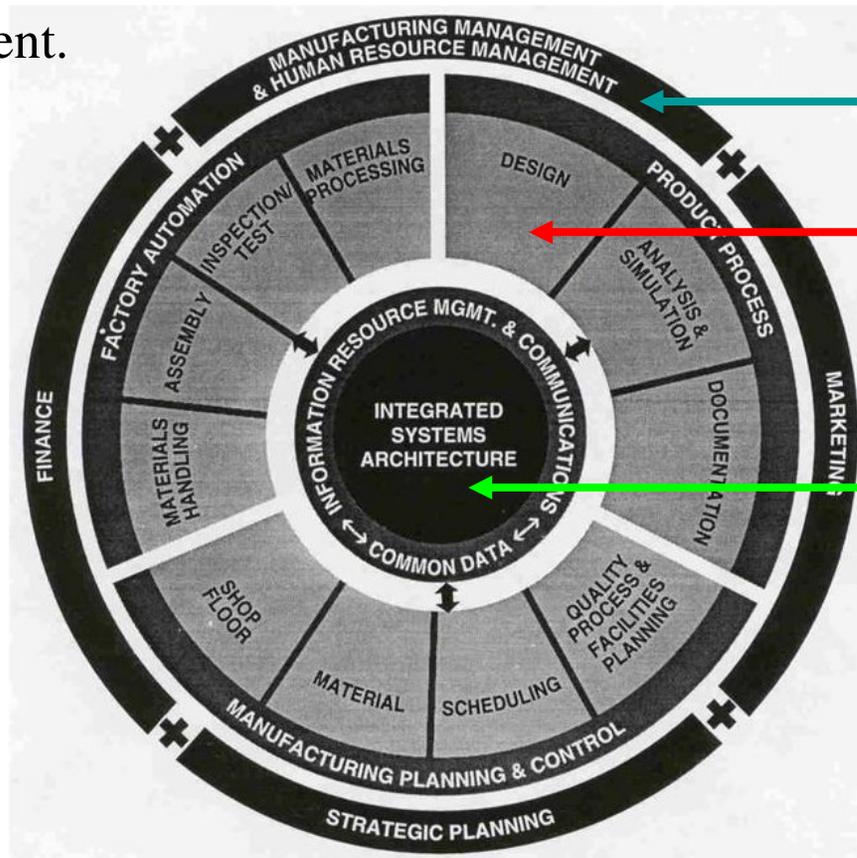
Automatización Industrial -II

CIM Wheel



<http://www.sme.org>

► Este diagrama conceptual tiene la ventaja de listar 21 de los aspectos de la operación de una compañía con diferentes interconexiones donde el centro de la rueda es conformado por Integrated systems architecture and information resource management.



El negocio

Diseño y Fabricación

Sistemas de Información

SME CIM Wheel

1985

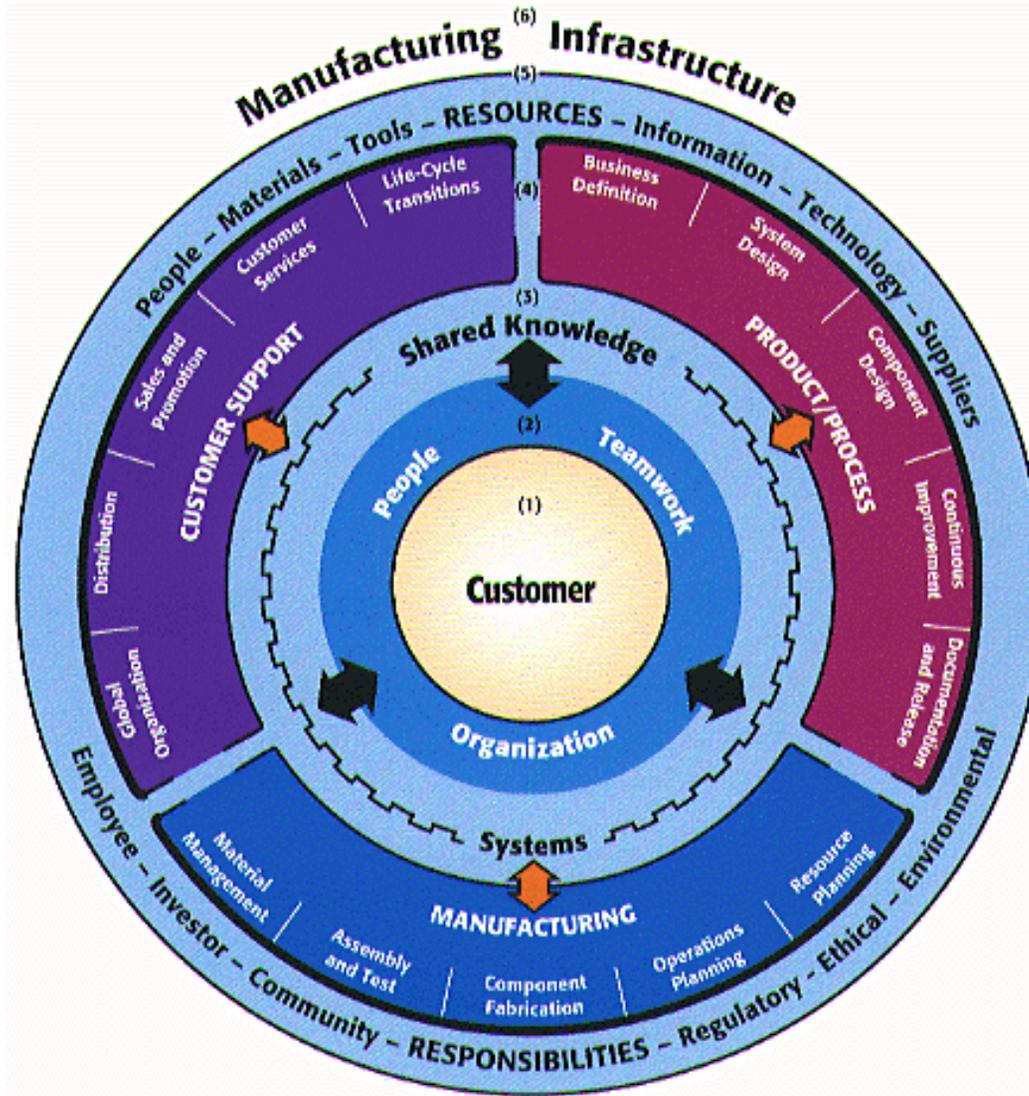


Automatización Industrial -II

Enterprise Wheel



<http://www.sme.org>



SME Enterprise Wheel

1993



Sistemas CIM



<http://www.sme.org>

● Definición

CIM: Es la **integración total** de una empresa de manufactura a través del uso de sistemas integrados y la comunicación de datos, junto con una nueva filosofía de gerencia, que mejora la eficiencia organizacional y de sus recursos humanos.

Computer and Automation Systems Association of SME (CASA/SME)



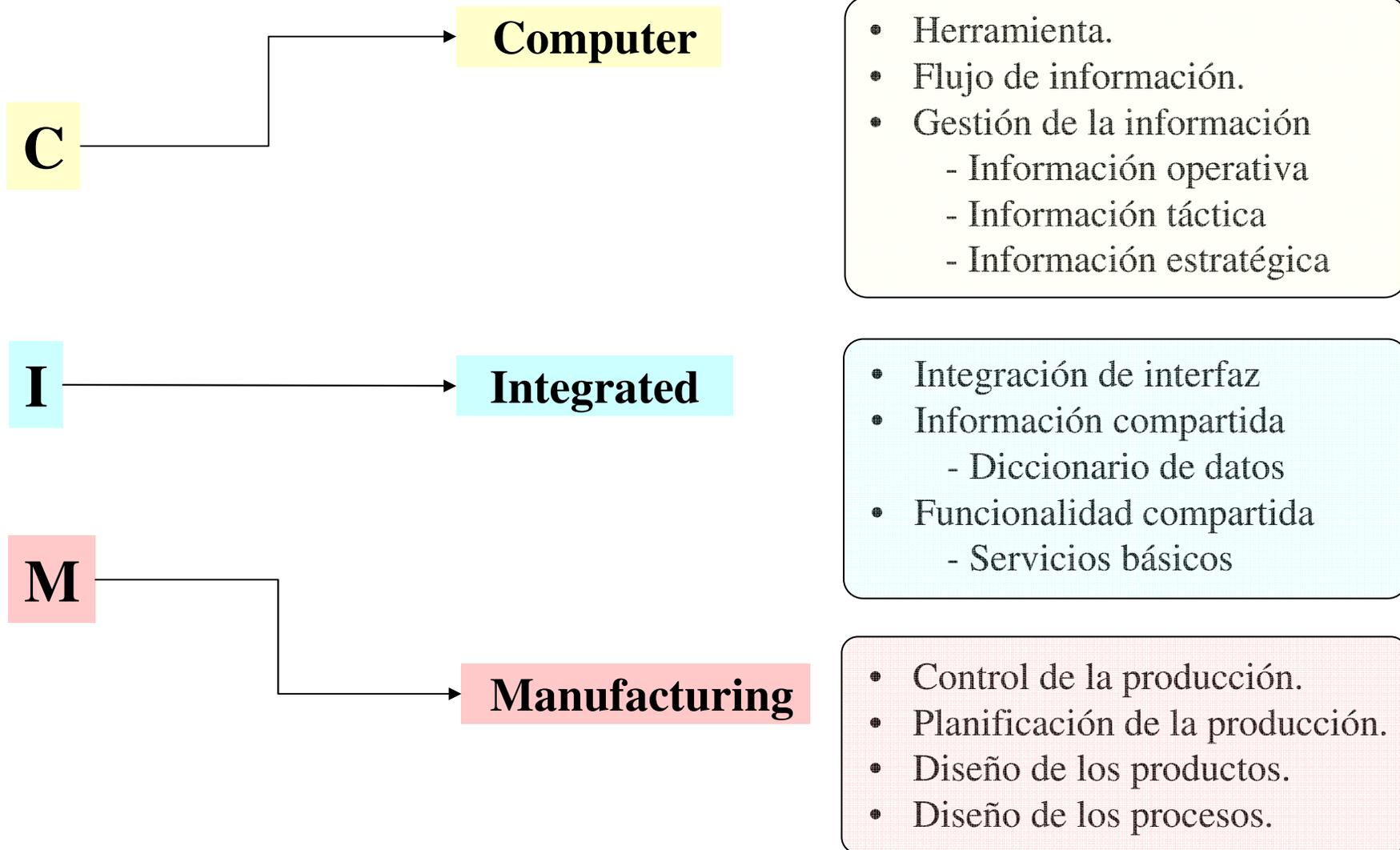
¿ CIM es una aplicación que puede comprarse como un software ?

¿ Las soluciones CIM son similares en las empresas ?

¿ Es CIM el resultado de interconectar ordenadores y máquinas ?

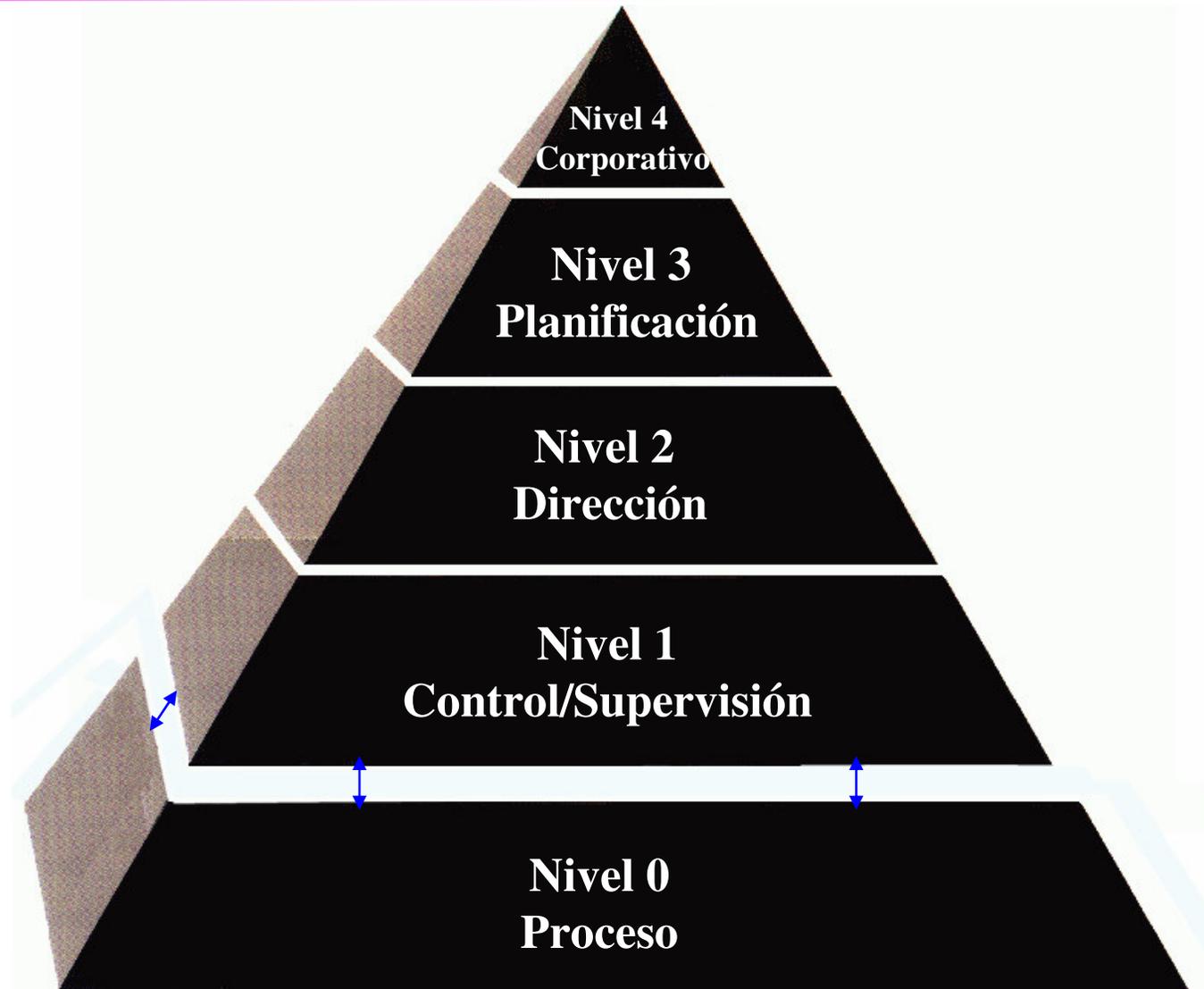


Sistemas CIM



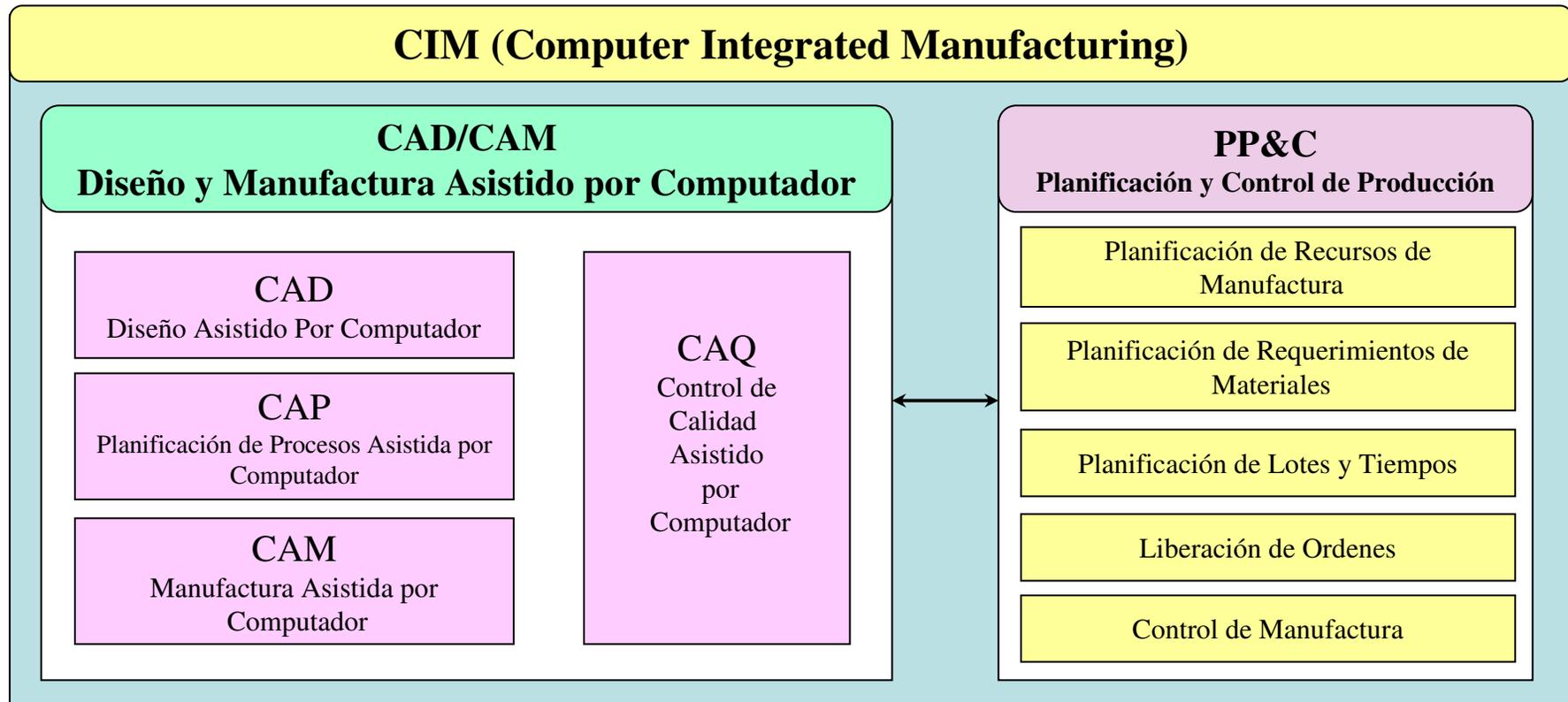


Pirámide CIM



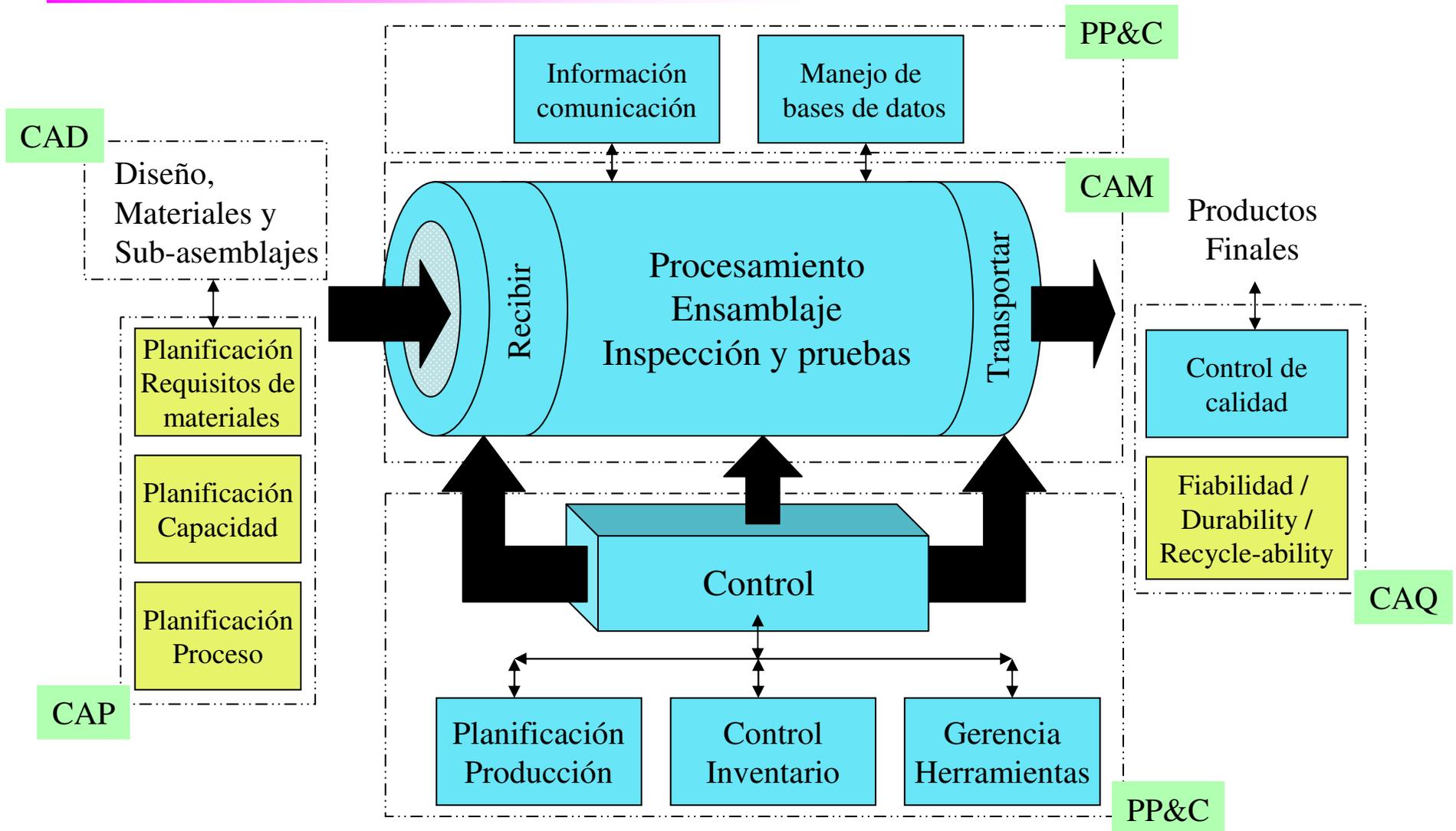


Esquema funcional de un sistema CIM



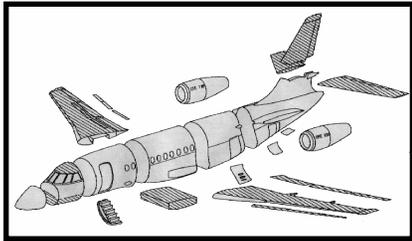


Esquema funcional de un sistema CIM

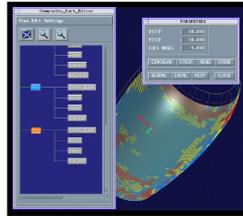




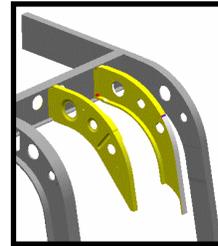
Sistemas CIM: Ejemplo CAD/CAM



Development by Section



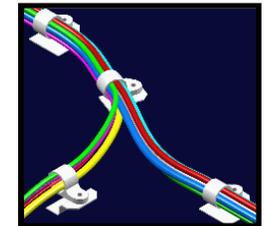
Composite Panels



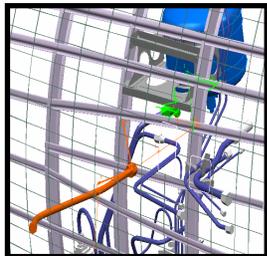
Structures



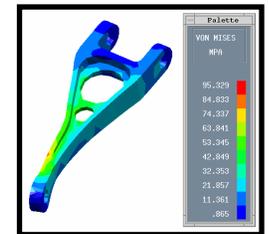
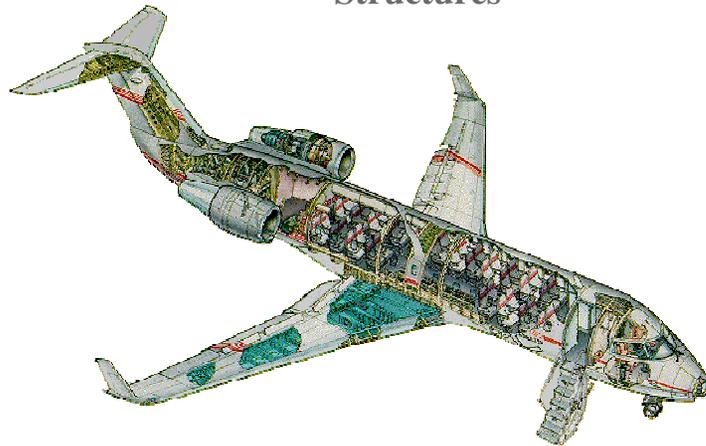
Tubing



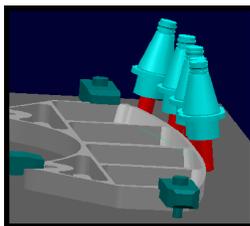
Electrical



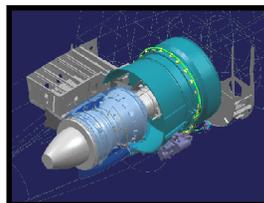
Maintenance



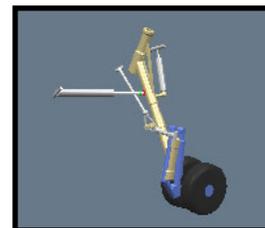
Part Analysis



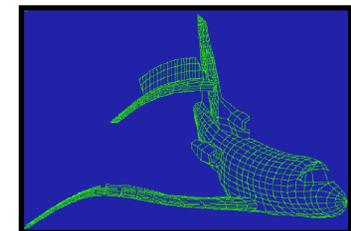
Manufacturing



Digital Pre-Assembly



Kinematics



Structure Analysis



Sistemas CIM: Ejemplo CAD/CAM



Nesting, Cutting

Manufacturing Process

Arc Welding

Virtual Product Management

P&ID

Equipment

Piping

Isometrics

Ductwork

Raceway

Electrical

Detail Design

Functional Design

Conceptual Design

Initial Design and Fairing

Structural Frame Analysis

FEA

Drafting

Drawing Production



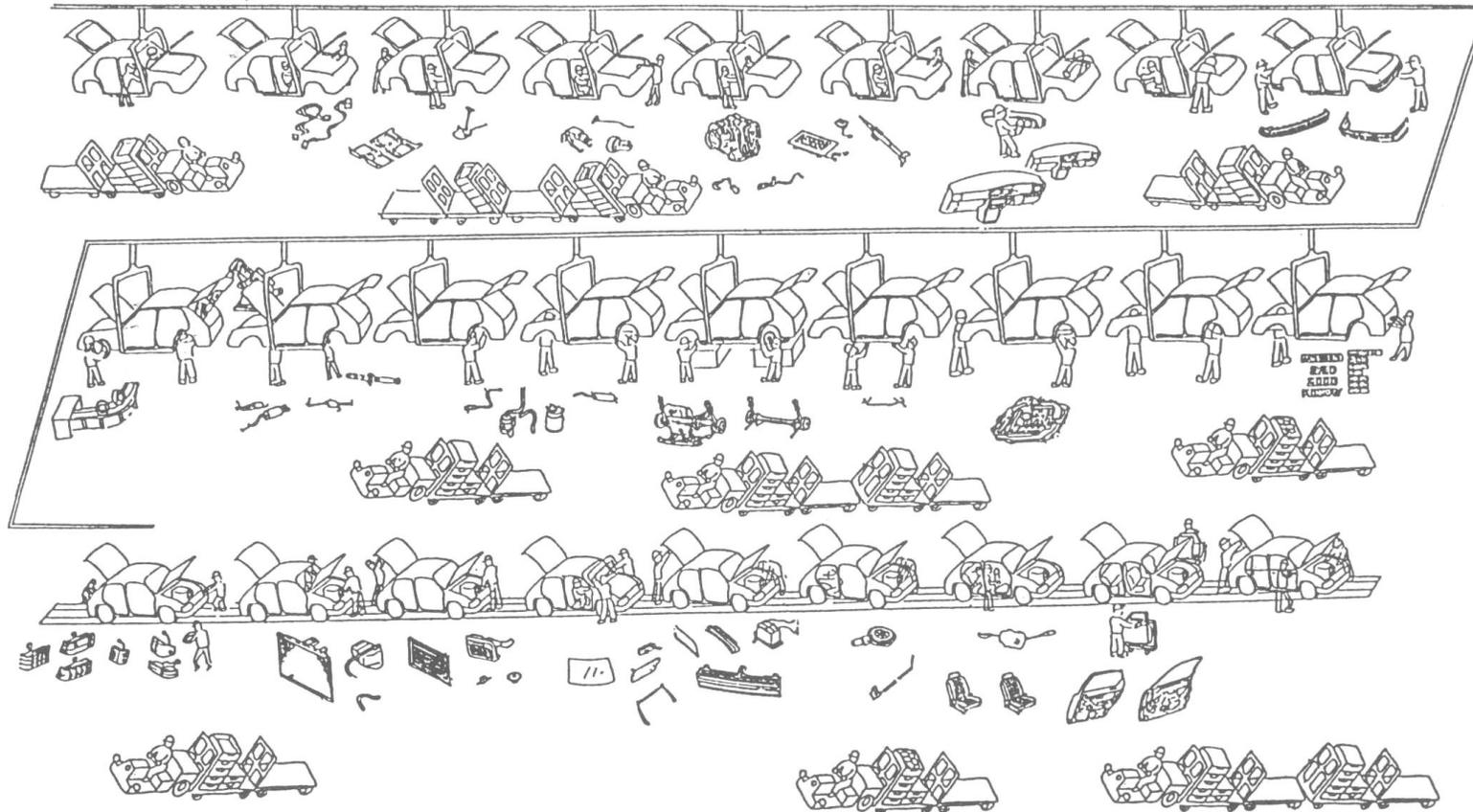
Sistemas CIM: Ejemplo CAP

- Ensamblaje de un coche





Sistemas CIM: Ejemplo CAP



Mixed Model
Production System

4~5 Models Per Line

High Flexibility
in Production



Complexity in

- Material Handling
- Processes
- Operations



Difficulty in
Automation



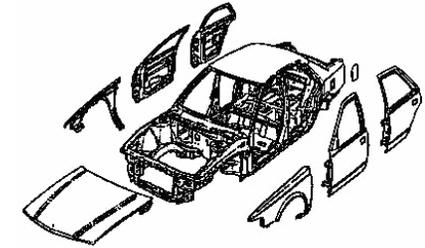
Manual Operation



Automatización Industrial -II

Sistemas CIM: Ejemplo CAP

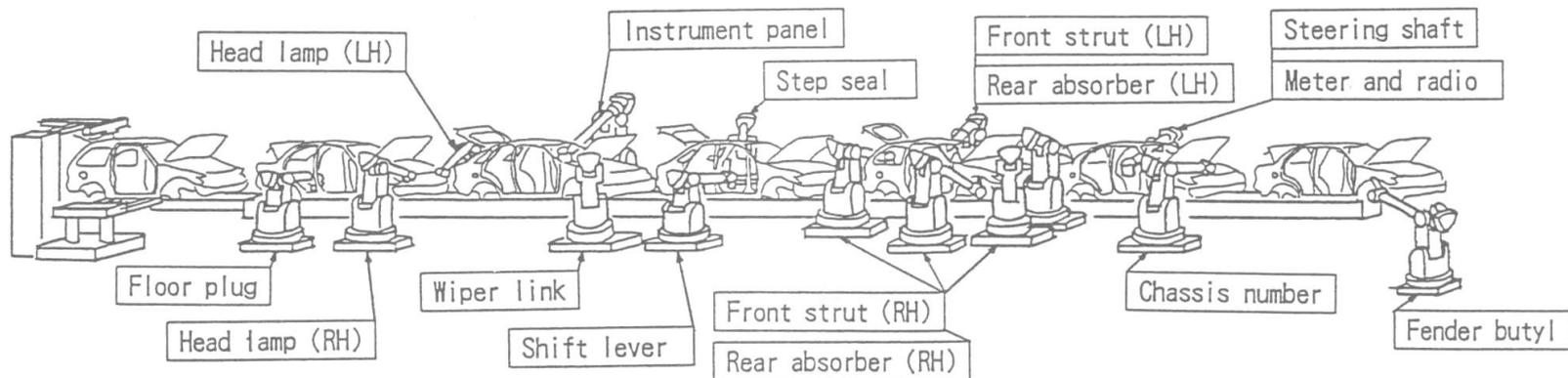
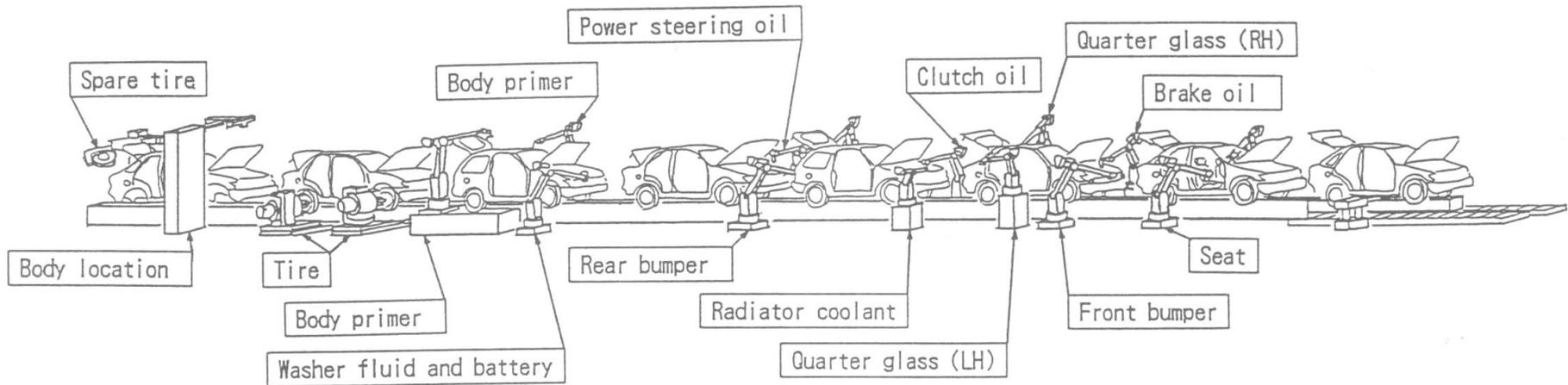
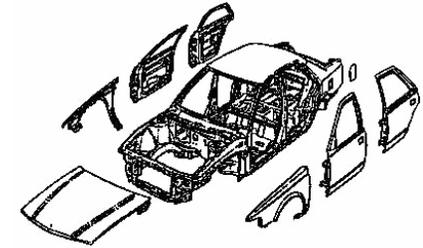
- Ensamblaje del chasis





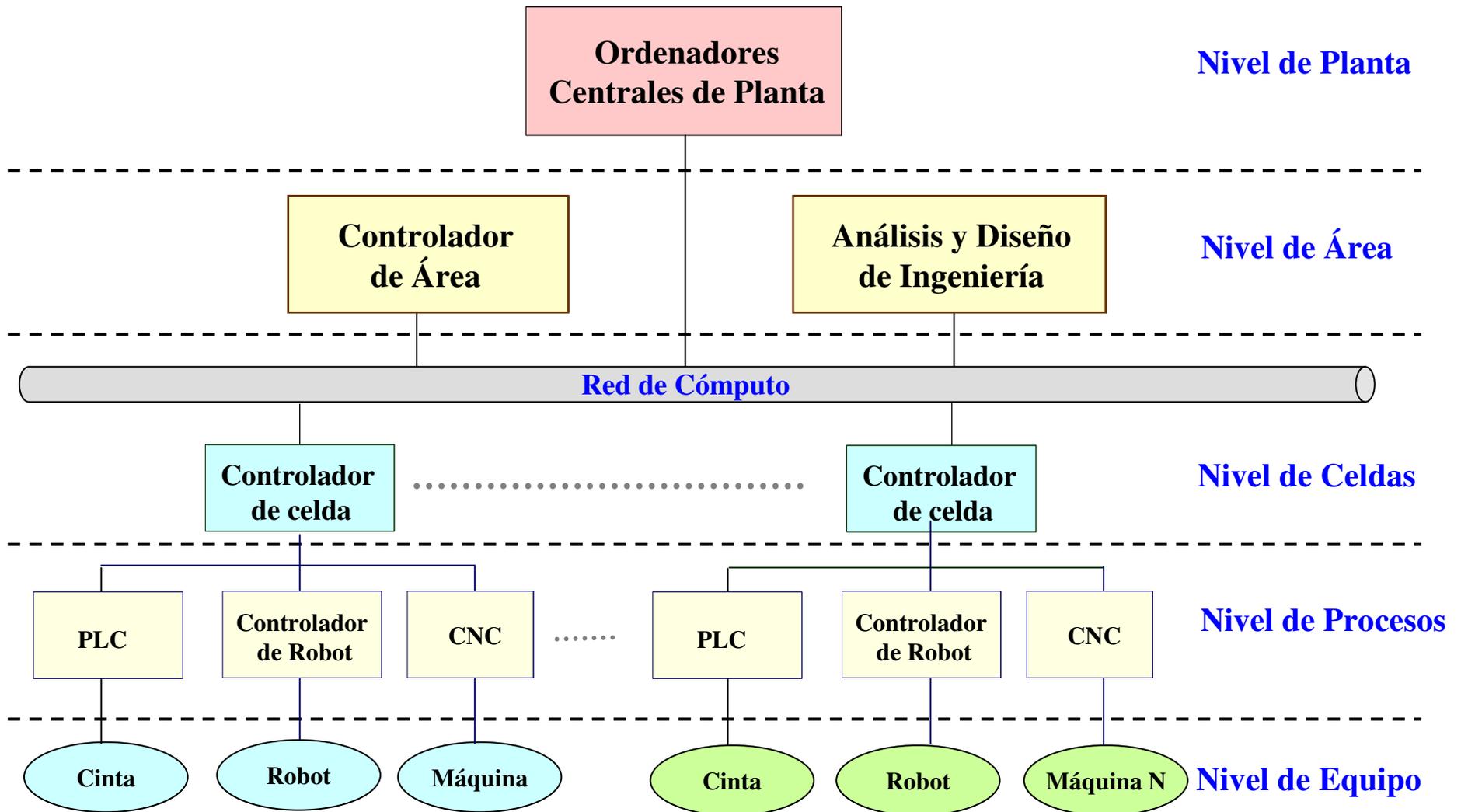
Sistemas CIM: Ejemplo CAP

- Ensamblaje del chasis





Niveles Jerárquicos de un CIM





Niveles Jerárquicos de un CIM

- **Nivel de Controlador de Planta**
 - Nivel más alto de la planta
 - Ordenadores centrales (mainframes)
 - Funciones corporativas
 - Administración de recursos
 - Planificación general





Niveles Jerárquicos de un CIM

● **Nivel de Controlador de Área**

- Minicomputadoras
- Control de operaciones de producción
- Responsable de coordinación y programación de las actividades de las celdas
- Entrada y salida de material
- Análisis y diseño de ingeniería
 - Diseño del producto
 - Análisis
 - Pruebas
- Planificación asistida por computador PAC
- Diseño asistido por computador CAD
- Planificación de requerimientos de materiales MRP





Niveles Jerárquicos de un CIM

- **Nivel de Controlador de Celda**

- Programación de las ordenes de manufactura
- Coordinación de actividades dentro de la celda
- Secuencia y control de los controladores de equipos.
- Ordenadores PC o estaciones de trabajo.

- **Nivel de Controlador de Procesos**

- Controladores de equipos
- Controladores de Robots RC
- Controladores Lógicos Programables PLC
- Máquinas de control numérico CNC
- Comunicaciones con máquinas del mismo nivel.



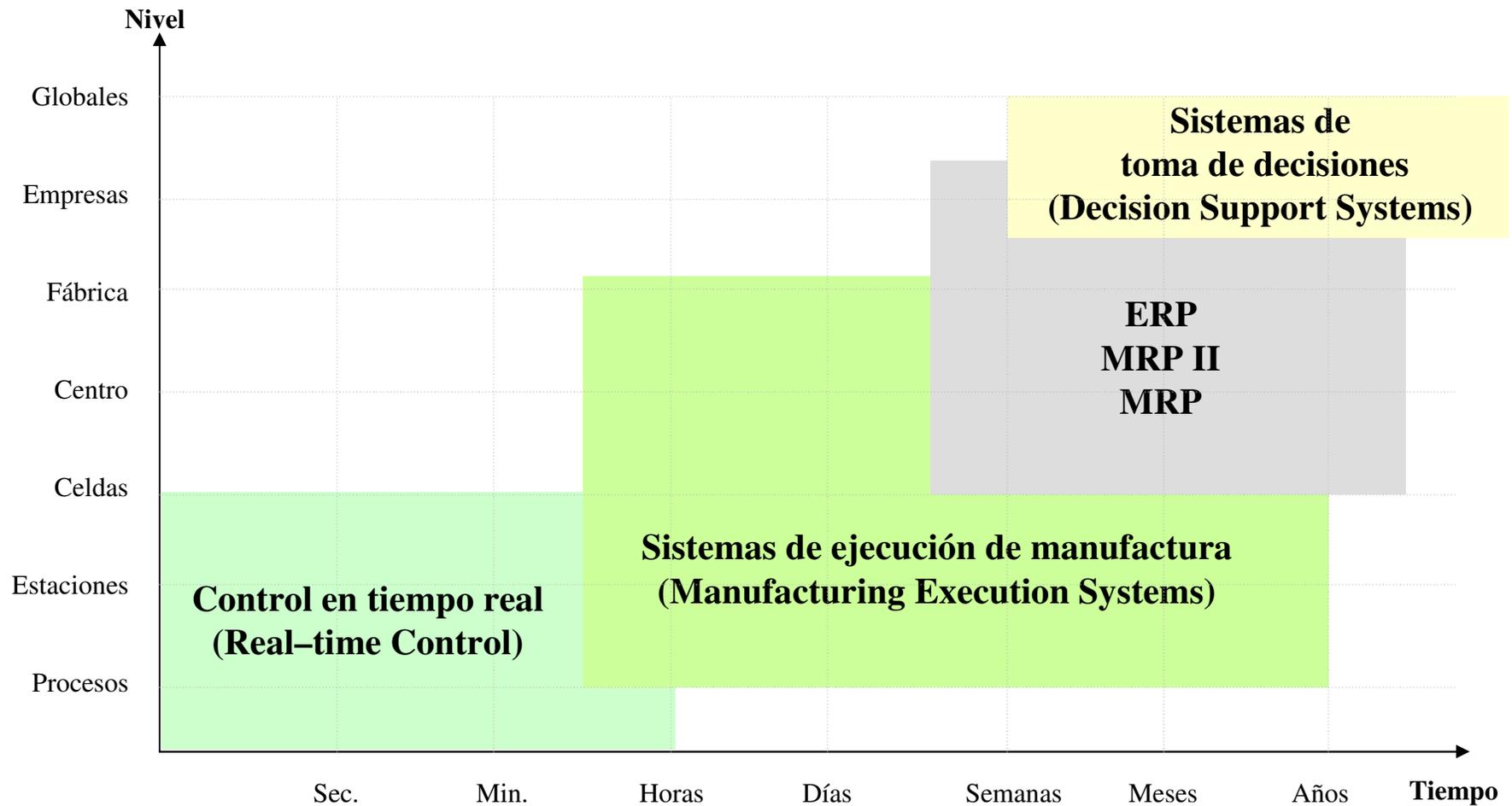
Niveles Jerárquicos de un CIM

- **Nivel Equipo**

- Nivel más bajo
- Dispositivos que ejecutan los comandos de control
 - Actuadores
 - Switches
 - Válvulas.



Niveles de Automatización CIM



ERP: Enterprise Resource Planning

MRP: Material Requirement Planning



Aspectos de un CIM

Aspecto	¿ Que se debe analizar en un CIM ?	Contenido
Procesos		<ul style="list-style-type: none"> • Procesos de la compañía. • Jerarquías de los procesos • Grupos funcionales • Secuencia de funciones
Información		<ul style="list-style-type: none"> • Tipo de información • Relaciones • Flujo de información • Estructuración
Recursos		<ul style="list-style-type: none"> • Recursos Tecnológicos <ul style="list-style-type: none"> - Capacidades - Infraestructura • Recursos Humanos <ul style="list-style-type: none"> - Habilidades - Experiencias, Conocimientos
Organización		<ul style="list-style-type: none"> • Enfoque de producto o proceso • Estructura organizacional • Enfoque de control



Aspectos de un CIM

- **Aspectos Administrativos: MRP (Material Requirement Planning)**
 - ▶ Método utilizado para derivar el calendario maestro de la producción conocido como (**MPS**), a partir de pronósticos y /o órdenes de ventas.
 - ▶ Está basado en las listas de materiales *Bill of Materials* (**BOM**) para la producción que se especifica en el MPS y el inventario actual con salidas de órdenes de compra y órdenes liberadas en la planta de producción.
 - ▶ La lista de materiales (BOM) representa las partes requeridas y el material utilizado en la manufactura de un producto en el sistema MRP.
 - ▶ El MRP funciona como:
 - 1- El MRP basado en el plan maestro de producción MPS, obtiene una lista de materiales y componentes de acuerdo con el BOM.
 - 2- Calcula cuando se tiene que comenzar a realizar los productos considerando los tiempos de entrega y de manufactura.



Aspectos de un CIM

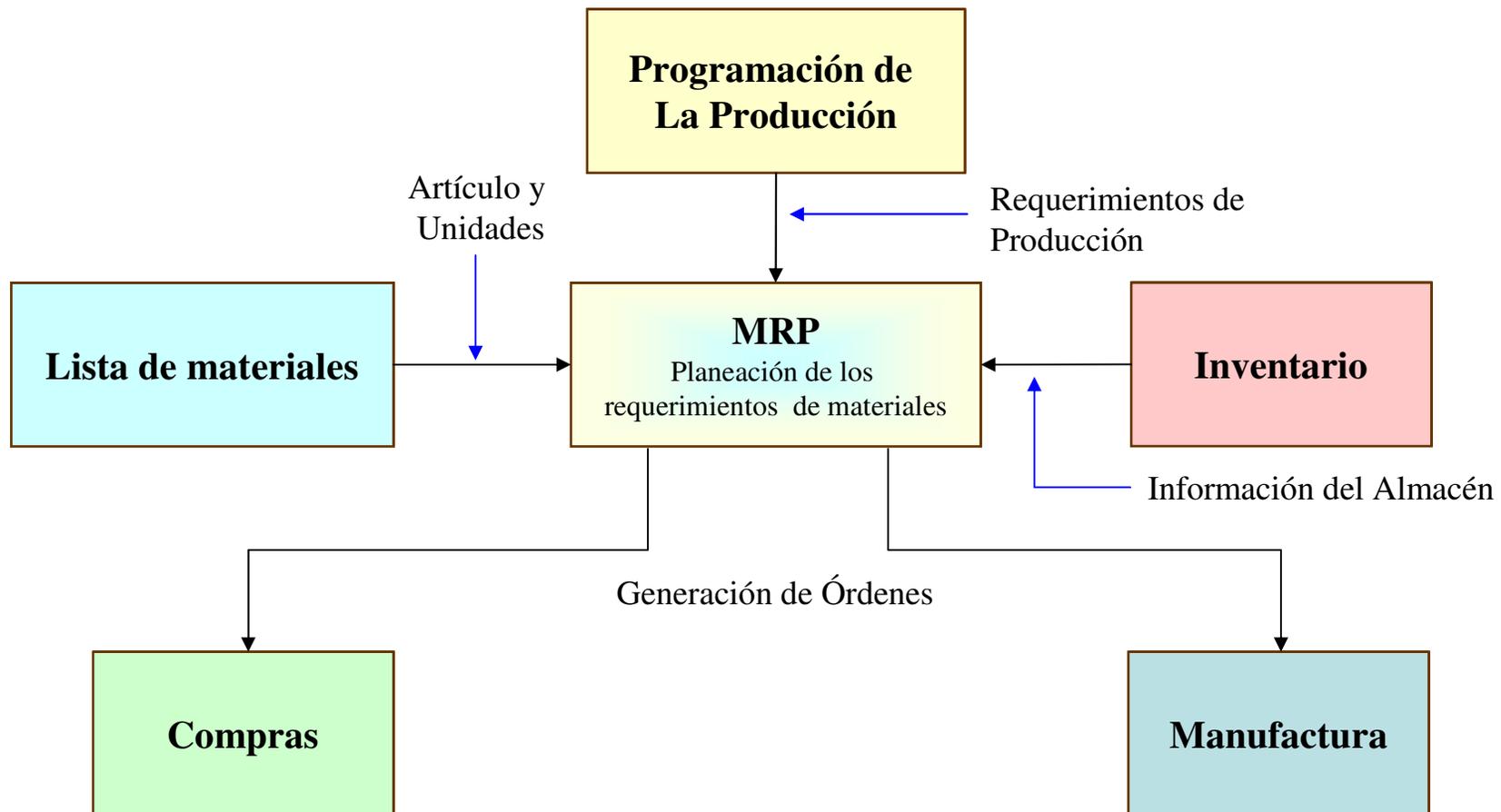
- **Aspectos Administrativos: MRP II**

- ▶ El **MRP** evolucionó a un sistema totalmente integrado de planeación de recursos de manufactura **MRP II**, que incluye los aspectos de MRP y también integra la capacidad de planeación de requerimientos, planeación de la producción y control de las actividades de producción.
- ▶ MPR II cuenta con dos características :
 - 1. Un sistema financiero y operacional** que cubre los aspectos de negocios de la compañía como ventas, producción, ingeniería, inventarios y contabilidad.
 - 2. Un simulador** que permite evaluar planes de producción como apoyo a la toma de decisiones.



Aspectos de un CIM

- **Aspectos Administrativos: MRP II**





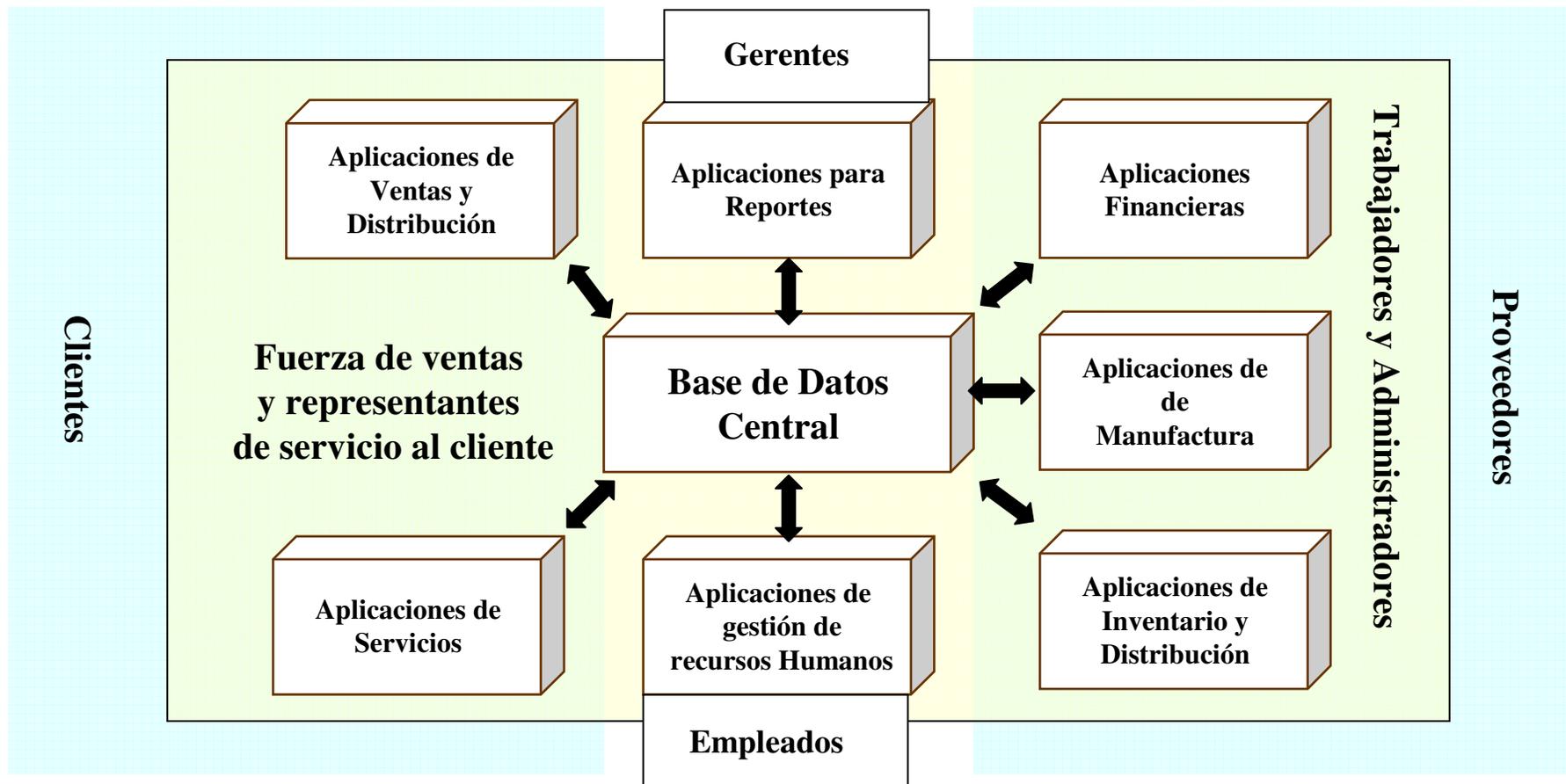
Aspectos de un CIM

- **Aspectos Administrativos: ERP (Enterprise Resource Planning)**
 - ▶ ERP concierne en asegurar que las decisiones de las firmas de manufactura no sean hechas sin tomar en cuenta su impacto en la cadena de suministro tanto hacia arriba como hacia abajo.
 - ▶ Las principales áreas en los negocios (ingeniería, contabilidad, mercadotecnia, etc.), son afectadas y afectan las decisiones de producción.
 - ▶ ERP, es un software que integra finanzas, distribución y manufactura con interfases con otras aplicaciones
 - ▶ Mientras MRP II programa una planta, ERP programa múltiples plantas.
 - ▶ Opera con bases de datos integradas.



Aspectos de un CIM

- Aspectos Administrativos: **ERP (Enterprise Resource Planning)**





¿Por qué CIM?

- **Para ejercer la competencia**

- ▶ **Reducir Tiempo:** Desde que un concepto es introducido en el diseño, hasta que es enviado al mercado (tiempo para mercado).
- ▶ **Reducir Costos:** Aplica a todos los aspectos de la producción, pero especialmente a costos de materiales y personal.
- ▶ **Reducir Inventario:** Esto implica reducción de espacio de planta, espacio de almacén, manejo de material y personal asociado con estas actividades
- ▶ **Incrementar calidad:** Calidad en los procesos, productos y servicios de la empresa. Calidad en el bienestar de los recursos humanos, etc.
- ▶ **Incrementar la sensibilidad para los clientes:** Hacer lo que el cliente requiere. Manufactura centrada en el cliente.



¿Por qué CIM?

- **Para coordinar y organizar datos**

- ▶ La principal clave de cómo CIM ayuda a la compañía a responder a las presiones de la competencia es por un mejor uso de los datos.
- ▶ Primero se requiere que los datos estén coordinados y organizados, lo cual, puede ser logrado por medio de Bases de Datos.
- ▶ Segundo se requiere que los datos estén realmente accesibles, CIM puede lograr esto por medio de redes adecuadamente diseñadas.

- **Para eliminar errores y su costo asociado**

- ▶ Eliminar los potenciales errores de transcribir datos.



¿Por qué CIM?

- **Para facilitar ingeniería simultánea**
 - ▶ Realizar tareas en paralelo más que de manera secuencial.
 - ▶ Tareas de optimización como reducción de tiempos de operación, retardos, etc.
 - ▶ Todo esto en paralelo con la operación de la planta y utilizando datos de operación.
- **Para automatizar la comunicación y aumentar su rapidez..**
 - ▶ Las redes de una implantación CIM, permiten envío y recepción de mensajes, (memos, documentos, etc.), por medio del correo electrónico.
 - ▶ Comunicaciones con clientes y proveedores.
 - ▶ Rapidez de comunicación entre planta y otros departamentos de la empresa.
 - ▶ EDI “Electronic Data Interchange”.



Características CIM

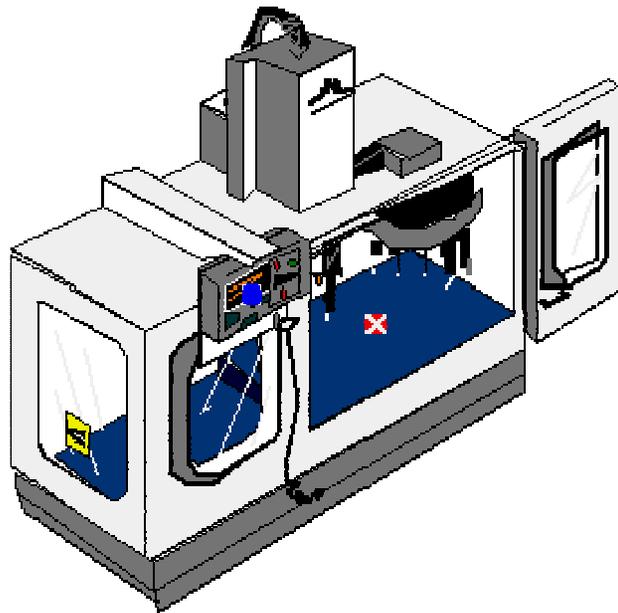
- ▶ Software y hardware para automatización.
- ▶ Un sistema de manufactura adquirido e instalado en la empresa.
- ▶ Estrategias de producción adaptables (JIT [Just-in-time], MRP).
- ▶ Filosofía de gerencia
 - Soporte a las decisiones en base a la satisfacción del cliente.
 - Integra principios de calidad total.
 - Identidad corporativa a todo nivel.
 - Búsqueda continua de mejoras.
- ▶ Existencia de una base de datos corporativa.
- ▶ Hardware y software como agentes catalizadores de los procesos de producción.



Integración en los sistemas CIM

El término integración puede ser interpretado de dos diferentes maneras:

- **Cuando distintas operaciones o funciones son integradas**

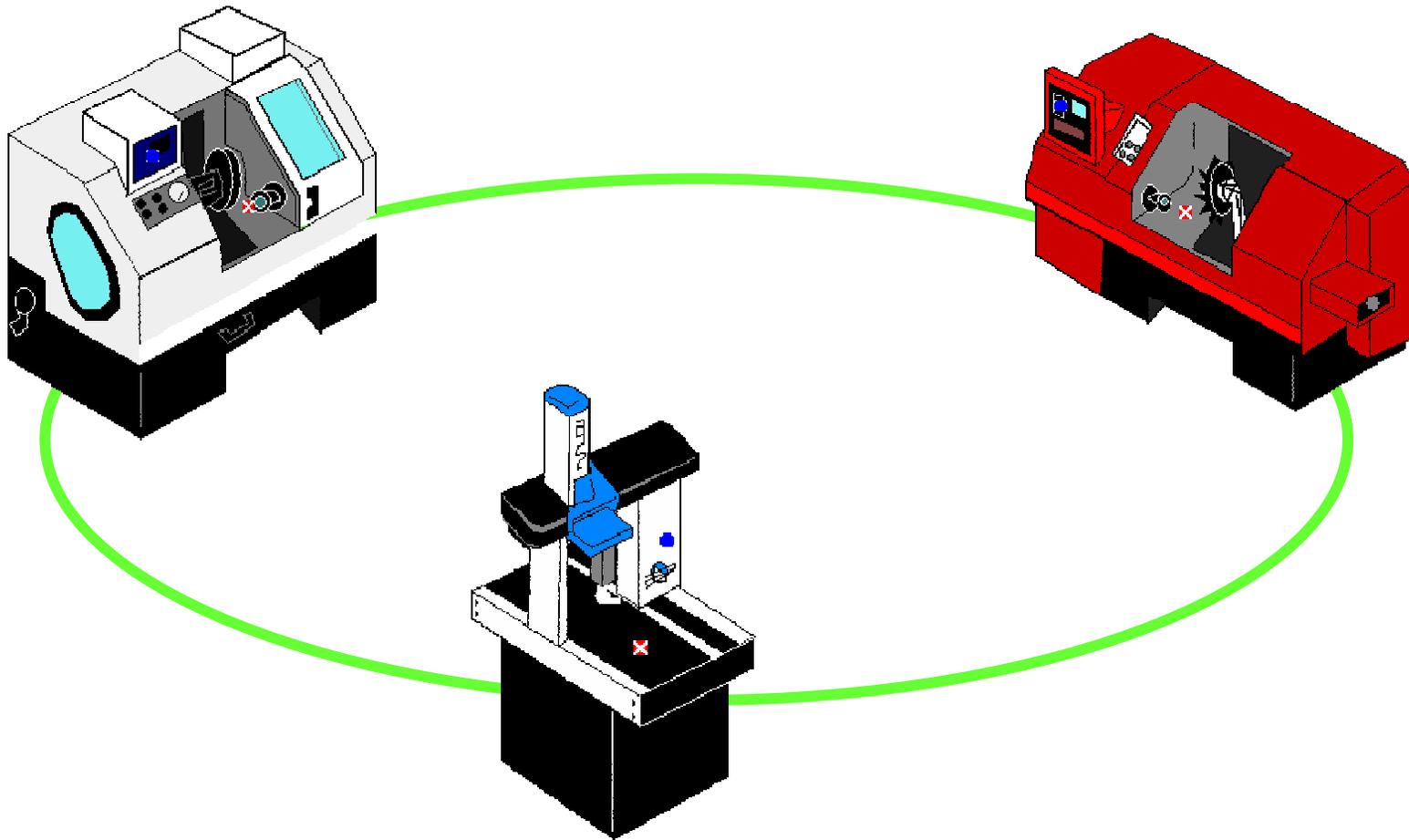


Las funciones “*drilling, milling and boring*” son integradas en una misma máquina.



Integración en los sistemas CIM

- **En CIM el término integración es referido a encadenar**





Integración en los sistemas CIM

● **Islas de Automatización (Island of automation)**

- ▶ La noción de islas de automatización surge en la década de los 80, desde los desarrollos de una serie de soluciones técnicas a problemas en la industria de la manufactura.
- ▶ De esta forma CIM en sus inicios era visualizado por muchos autores en términos de interfases entre las diferentes Islas de Automatización.
- ▶ Realizar una interfase para las islas de automatización, cuando éstas han sido concebidas de manera aislada, es siempre complejo y problemático.
- ▶ Los investigadores en CIM, para aliviar estos problemas reconsideran el diseño de los sistemas de manufactura con una visión en la dirección de integración total (Full Integration).



Integración en los sistemas CIM

Miller et al. (1986)* en “Taking Stock of CIM”, identifican tres tipos de integración:

- **Integración Técnica (Technical Integration)**

- ▶ Referida a la actividad de establecer comunicación electrónica entre las diferentes áreas funcionales, cuestiones como: canales de comunicación, protocolos y normas.
- ▶ Surgen protocolos como MAP (*Manufacturing Application Protocol*) y TOP (*The Office Protocol*), con gran facilidad para la integración entre las islas de automatización.



Integración en los sistemas CIM

- **Integración de Procedimientos (Procedural Integration)**
 - ▶ Referida a diferentes grupos funcionales, los cuales comparten datos, tienen una visión consistente de cómo interpretar los datos y de esa manera son capaces de usar apropiadamente procedimientos compartidos.
- **Integración de Objetivos (Goal Integration)**
 - ▶ Es el más alto nivel de integración que se considera alcanzado cuando diferentes áreas funcionales (Islas), utilizan datos e información compartida para alcanzar objetivos comunes compartidos.



Resumen:

- **CIM** es una industria manejada por tecnología, con cada brazo de la industria condicionado por su propio y particular conjunto de experiencias, requerimientos y circunstancias.
- Es claro que una industria que busca alcanzar CIM, debe tener en consideración su propia inversión existente en las islas de automatización, con frecuencia referida a la herencia de sus sistemas, antes de planificar una evolución hacia CIM
- Todas estas circunstancias hacen que la descripción de CIM tenga una tendencia altamente dependiente de la experiencia y perspectiva de las condiciones individuales que cada industria ofrece.
- La comprensión del problema CIM, depende de la experiencia y perspectiva desde puntos de vista individuales.
- No existen recetas universales para alcanzar el CIM.
- Cada industria debe desarrollar su propio CIM, en función de su estado, procesos y objetivos deseados



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 12 – Jueves 25 de Mayo 2006

Computer Integrated Manufacturing

Manufactura Integrada por Computador

CIM - Introducción



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 13 – Miércoles 31 de Mayo 2006

Computer Integrated Manufacturing

Manufactura Integrada por Computador

CIM - Estrategias



Estrategia para la automatización en la empresa

● Empresa

▶ Una empresa es un sistema estructurado de alguna manera, con el fin de cumplir una misión

▶ Estructura:

- **Unidades de producción:** Asociada a la cadena de valor de la empresa.

- **Unidades de soporte:** Facilitan los recursos a las unidades de producción para que cumplan su misión

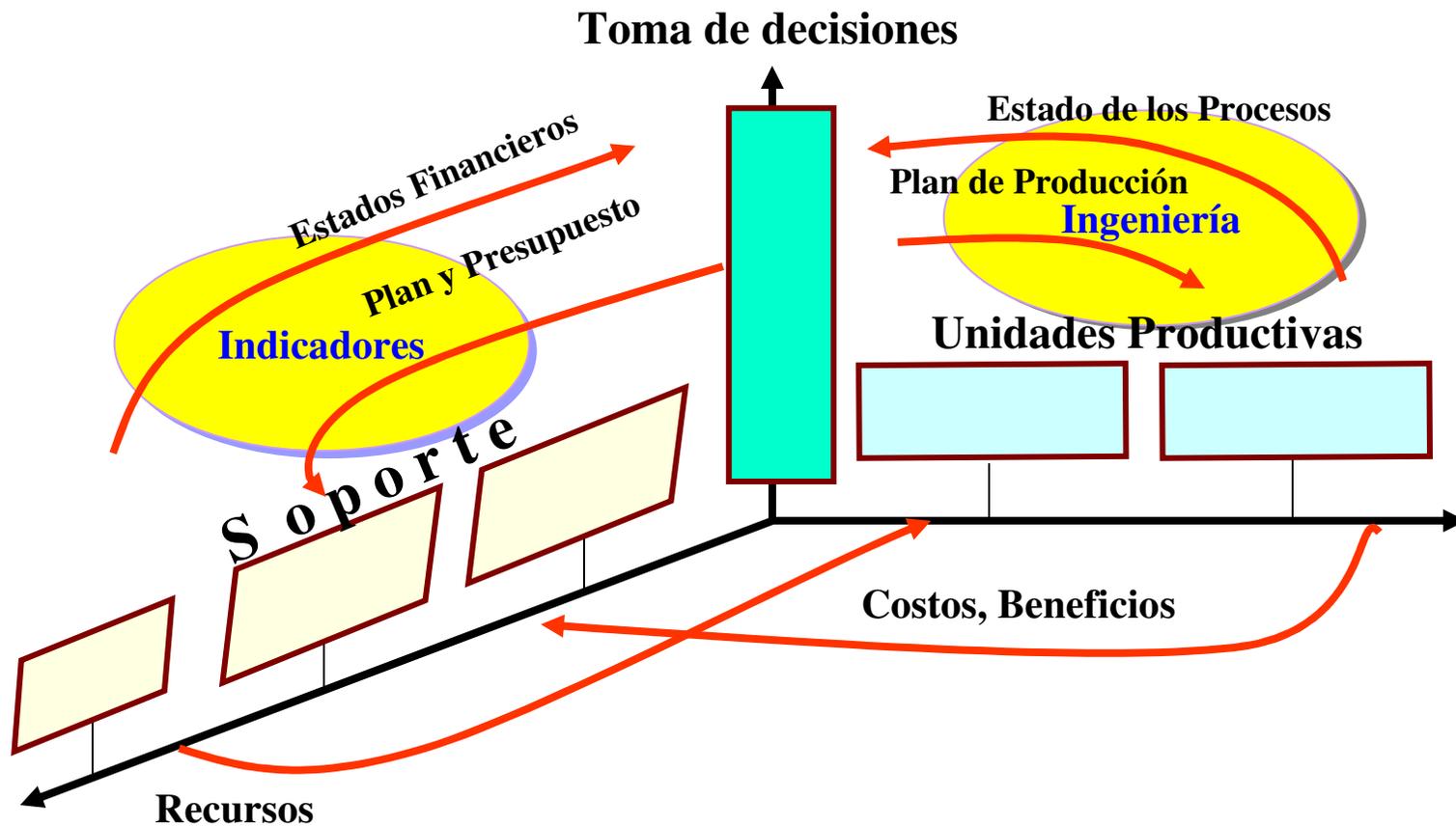
- **Estructura de planificación y coordinación:** Coordina las actividades de producción, Planifica dichas actividades, Asegura los recursos para el cumplimiento de las actividades.





Estrategia para la automatización en la empresa

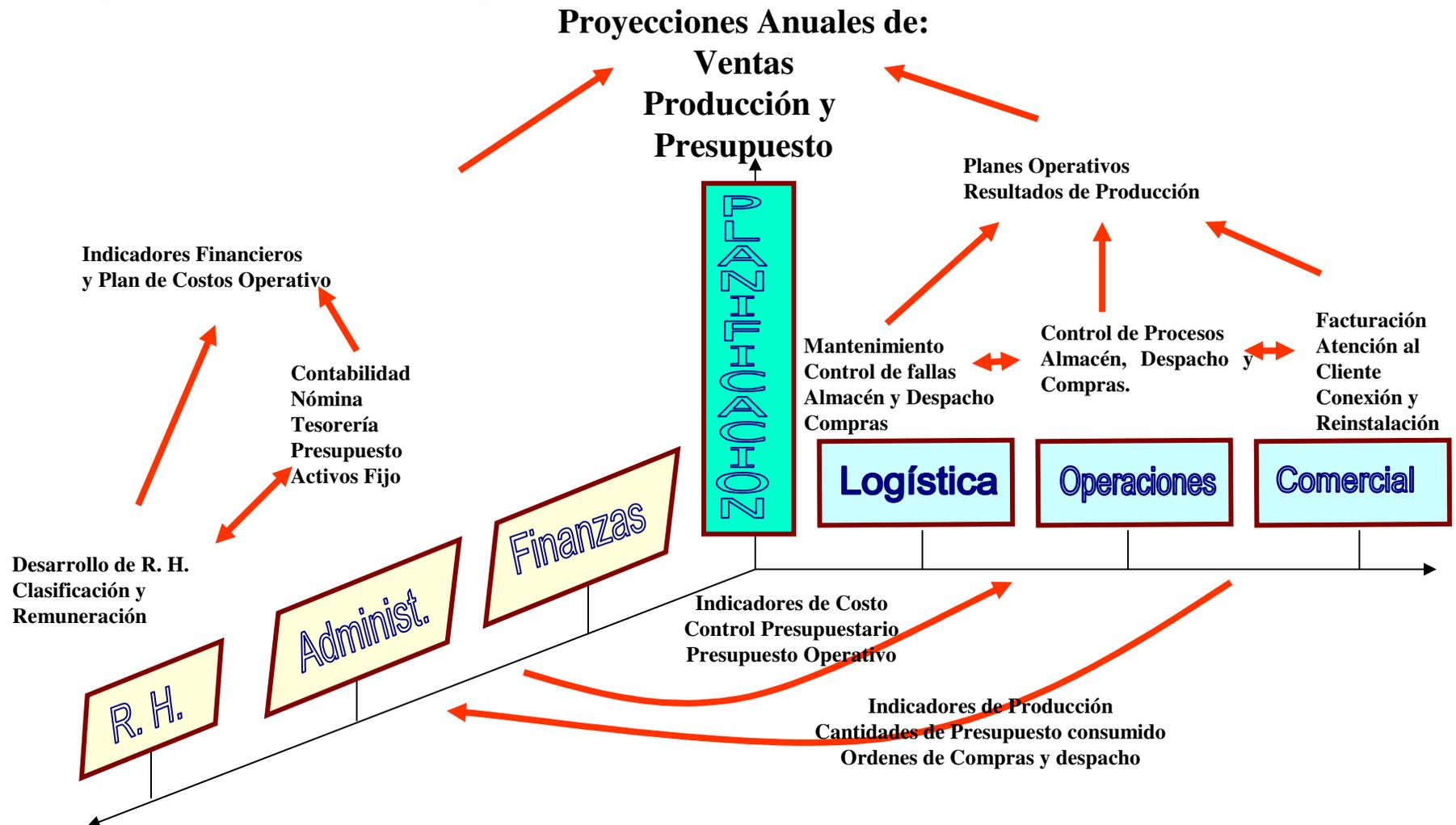
- **Empresa:** Modelos empresariales tres ejes





Estrategia para la automatización en la empresa

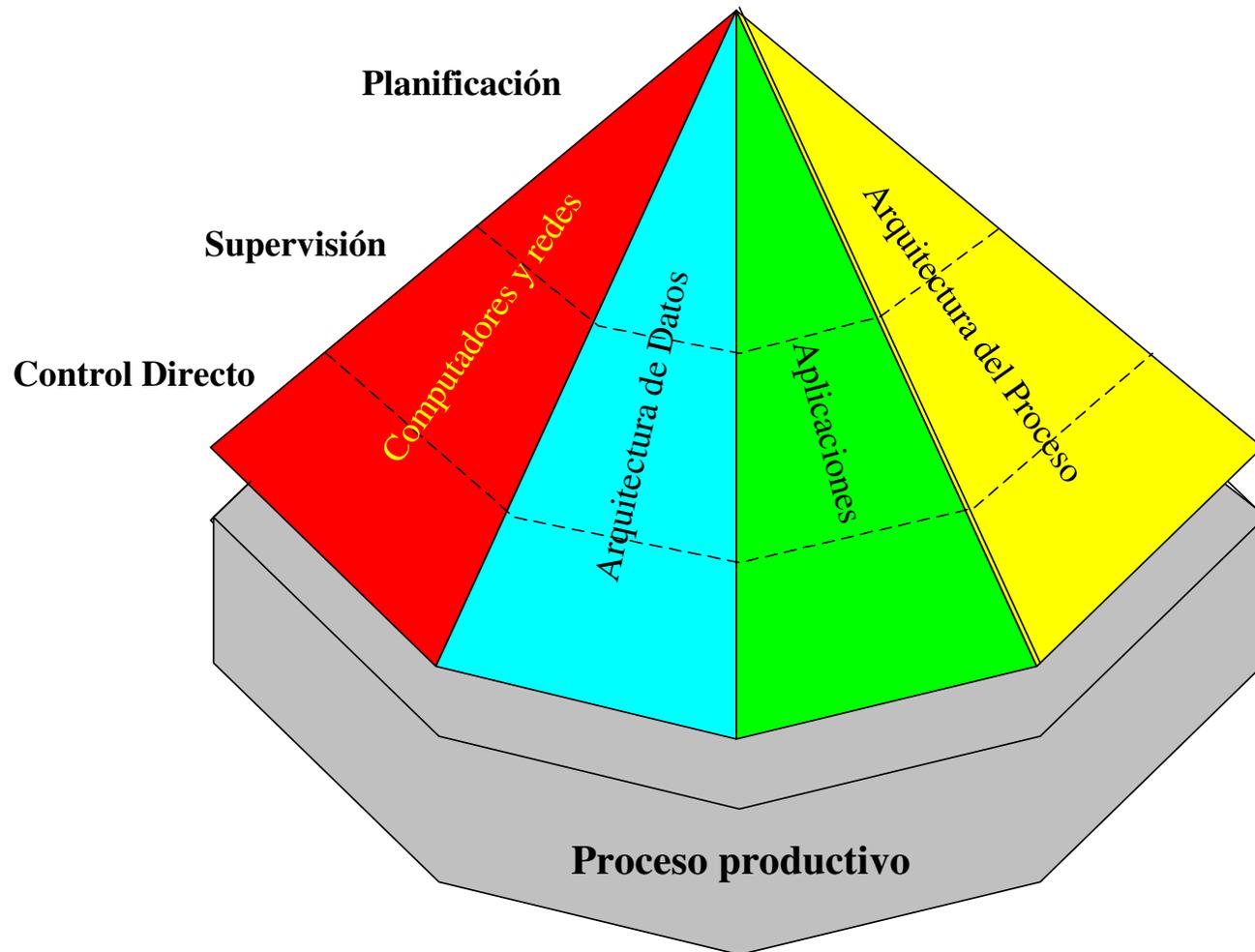
- **Empresa: Modelos empresariales tres ejes**





Estrategia para la automatización en la empresa

- **Aspectos a evaluar para definir el nivel de automatización**





Estrategia para la automatización en la empresa

- **Acepciones al término automatización**

- ▶ **Control de procesos**

- **Conceptual:** Capacidad de regular un proceso de manera automática (Medir, Inferir el estado, Calcular la acción a realizar, Actuar)
- **Instrumental:**
 - Selección de equipos de medición, cálculo y actuación.
 - Determinación de las estrategias de supervisión.
 - Control y Selección de la forma de interactuar los operadores con el proceso.



Estrategia para la automatización en la empresa

- **Acepciones al término automatización**

- ▶ **Planificación y programación de la producción**

- **Conceptual:**
 - Capacidad de ajustarse para cumplir compromisos de producción (Conocimiento de la capacidad de producción, Manejo de los recursos de producción: Disponibilidad, Rendimiento, Costos de producción, relaciones entre equipos)
 - Tareas (Medir, evaluar, programar).

- **Instrumental:**
 - Selección de equipos de control y elementos de transferencia de información.
 - Determinación de los métodos y estrategias de planificación.
 - Determinación de las formas de programación.



Estrategia para la automatización en la empresa

● **Tecnologías para cada nivel**

▶ **Niveles:**





Estrategia para la automatización en la empresa

- **Tecnologías: Nivel de control**

- ▶ **Medición de variables**

- **Control:** Medición de las variables necesarias al proceso.
- **Detección de fallos:** Basados en modelos o basados en condición de los instrumentos.
- **Control inteligente:** Capacidad de tomar decisiones. Control adaptativo
- **Control supervisor:** Detección de eventos.
- Uso de energía, insumos, y otros elementos

- ▶ **Actuación:** Actuadores inteligentes. Ejemplo. Válvulas con PID

- ▶ **Control:** Directo (PID, control avanzado, control neuromimético), Supervisores (modelo de Wonham–Ramadge), Híbridos (Lygeros, Lennarson).



Estrategia para la automatización en la empresa

- **Tecnologías: Nivel de control**

- ▶ **Sistemas inteligentes**

- Sistemas que se adaptan a situaciones diferentes.
- Aseguran el control de manera autónoma (llegar a un punto en la trayectoria del sistema, de manera segura, en tiempo finito, y permanecer en él).
- El punto puede cambiar en el tiempo

- ▶ **Remotas inteligentes**

- Comunicación con campo.
- Capacidad de ejecución de Algoritmos.
- Comunicación con Sistema de Supervisión.



Estrategia para la automatización en la empresa

- **Tecnologías: Nivel de control**
 - ▶ **Centros de control**
 - **Supervisión clásica:**
 - IHM, Servidores de datos, Control de las remotas
 - Manejo de alarmas.
 - **Sistemas expertos de apoyo en línea:** Recomendaciones al operador.
 - **Integración con los sistemas de optimización, historiadores**
 - **Coordinación de sistemas de producción “complejos”**



Estrategia para la automatización en la empresa

- **Tecnologías: Nivel de control**
 - ▶ **Centros de control**
 - **Control supervisor**
 - Detección de fallas
 - Coordinación de los procesos
 - Ajuste de parámetros de controladores.
 - **Técnicas**
 - Sistemas expertos en línea
 - Control supervisor de eventos
 - Control avanzado.



Estrategia para la automatización en la empresa

- **Tecnologías: Nivel de Gestión de la producción**

- ▶ Uso de Equipos de procesamiento generales.

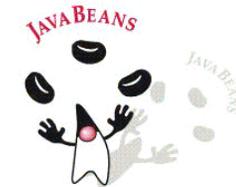
- ▶ Empleo de Middlewares (Corba, Tuxedo)



BEA Tuxedo

<http://www.omg.org/corba/> <http://www.bea.com>

- ▶ Modelos de integración empresarial (Workflow Enterprise Java Beans, CIMOSA, SP-95).



- ▶ Modelos de Gestión (Balance Score Card).



<http://www.balancedscorecard.org/>



- ▶ Bases de datos de Integración (Datawarehouses, Data minning).



Estrategia para la automatización en la empresa

● **Una visión de futuro**

- ▶ Dificultad para tener modelos globales de producción.
 - Integración de modelos
 - Modelos basados en transacciones. SDED y Sistemas Híbridos

- ▶ Manejo integral del proceso
 - Modelo integral de cada unidad de producción.
 - Sistemas Holónicos y Holarquías

- ▶ Incremento de la productividad
 - Ingenieros de procesos y automatización entonando los procesos
 - Procesos autónomos.



Estrategia para la automatización en la empresa

- Una visión de futuro: **Holonic Manufacturing Demos**

Agent Oriented Software Limited © Cambridge University

Introduction

Factory Layout

Holons

Scenario

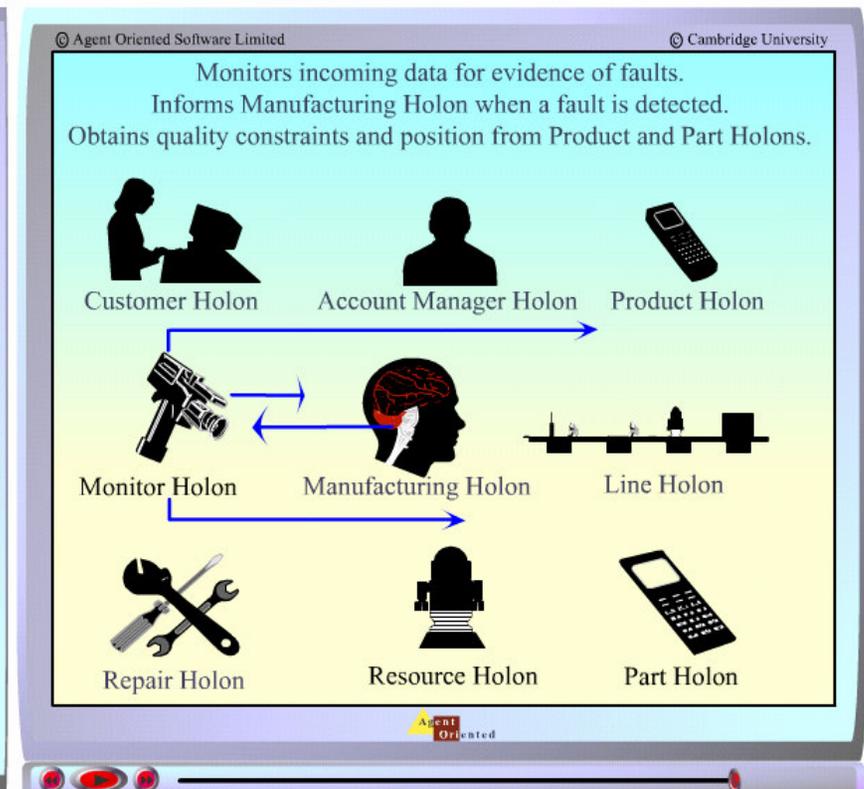
Breakdown

Recovery

Holonic Manufacturing of Mobile Phone Products

UNIVERSITY OF CAMBRIDGE

<http://www.agent-software.com/shared/resources/holonicDemos.html>





Estrategia para la automatización en la empresa

● **Una visión de futuro**

▶ **Dificultades**

- Cambiar la mentalidad de los empleados, del hombre de rutina al hombre pro-activo. Responsable del proceso y no de la operación.
- Temor (muchas veces justificado) a perder el trabajo.

▶ **Facilidades**

- Tecnología disponible (SCADAs basados en JAVA, Sistemas abiertos)
- Cooperación entre especialidades.



<http://www.jsent.co.za>



http://www.controlstar.com/tech_java.php

SCADA Simulator

<http://members.iinet.com.au/~ianw/simulate.html>



Resumen:

- La automatización de procesos conduce a tener altos índices de satisfacción del cliente.
 - ▶ Productos con costo inferior.
 - ▶ Productos entregados a tiempo.
 - Tecnología disponible
 - ▶ La informática y las comunicaciones permiten integrar ambientes de trabajos heterogéneos.
 - ▶ Los costos de automatización son relativamente bajos.
 - Dificultad
 - ▶ Modelar de manera integrada el proceso.
 - ▶ Definir las aplicaciones necesarias para las diferentes actividades no es sencillo.
-



Automatización Industrial - II

56 – 10569 2º Cuatrimestre 2006

Clase 13 – Miércoles 31 de Mayo 2006

Computer Integrated Manufacturing

Manufactura Integrada por Computador

CIM - Estrategias