

# “ACTUALIZACIÓN DEL SISTEMA OPERATIVO, MANUAL DE OPERACIÓN Y GUÍA DE PRÁCTICAS PARA EL SISTEMA DIDÁCTICO ROBÓTICO MÓVIL DEL LABORATORIO DE MECATRÓNICA DE LA FIMCP-ESPOL”

Osmar G. Lema C. <sup>(1)</sup>, Ing. M.Sc. Eduardo H. Orces Pareja <sup>(2)</sup>  
Facultad de Ingeniería en Mecánica y Ciencias de la Producción  
Escuela Superior Politécnica del Litoral (ESPOL)  
Campus Gustavo Galindo, Km 30.5 vía Perimetral  
Apartado 09-01-5863. Guayaquil-Ecuador

<sup>(1)</sup>Ingeniero en Mecánica, con preparación en diseño, automatización y robótica, email: [Olema@espol.edu.ec](mailto:Olema@espol.edu.ec)

<sup>(2)</sup>Master en Ciencias de Ingeniería Mecánica, California Institute of Technology- USA, email: [eorces@espol.edu.ec](mailto:eorces@espol.edu.ec)

## Resumen

*Con la aparición de la nueva carrera en la Escuela Superior Politécnica del Litoral, Ingeniería en Mecatrónica, la Facultad de Ingeniería en Mecánica y Ciencias de la Producción (FIMCP) hizo un nuevo laboratorio para esta rama.*

*El laboratorio de Mecatrónica, inició con la adquisición de equipos entre los cuales se encuentran: computadoras, un brazo robótico, estaciones de automatización y el sistema didáctico robótico móvil (Robotino® de Festo®).*

*Esta tesis se centra en el sistema didáctico robótico móvil, que después de ser actualizado con un nuevo sistema operativo, nace la necesidad de realizar nuevos manuales de uso y prácticas de laboratorio, debido a que las anteriores quedaron obsoletas*

*Con estas nuevas guías de laboratorio se espera suplir las necesidades académicas del estudiante y que logre asimilar los conocimientos adquiridos en el aula mediante análisis y pruebas.*

*Este trabajo, en primer lugar, presenta una descripción completa de sistema didáctico robótico móvil y sus componentes, denotando sus características y especificaciones técnicas e indicando también el procedimiento usado para su actualización.*

*Seguidamente, se da una introducción al nuevo software de programación de la unidad: Robotino®view 2 y otro software alterno “MatLab 7”, que posteriormente será complementada con las guías prácticas.*

*A continuación, se exponen cada una de las nuevas prácticas que se implementaran para comprensión y uso de la unidad robótica.*

*Por último, se espera llevar a cada estudiante a la comprensión total y familiarización de cada uno de los componentes y sensores contenidos en la unidad robótica.*

**Palabras Claves:** guías, prácticas, manual, laboratorio, mecatrónica, Robotino®, actualización, manejo, uso

## Abstract

*With the advent of new career at the ESPOL, Mechatronics Engineering, the Faculty of Mechanical Engineering and Production Sciences (Spanish acronym: FIMCP) implemented a new laboratory for this branch.*

*The Mechatronics Laboratory began with the acquisition of equipment, among which are: computers, a robotic arm, automation stations and a didactic mobile robotic system (Robotino® from Festo®).*

*This thesis focuses on the didactic mobile robotic system, which after being updated with a new operating system, comes the need for new manuals and laboratory practice, because the former were obsolete.*

*With these new laboratory's guides, is expected to meet the academic needs and that the students achieve assimilate the knowledge acquired in the classroom through analysis and testing.*

*First, this paper presents a didactic mobile robotic system's complete overview and its components, denoting its features and specs and indicating also the procedure used for its updating.*

*Following, it gives an introduction to the new software programming's guide: Robotino ®view 2 and other software alternate "MatLab 7", which will later be complemented with practical guides.*

*Then presents each of the new practices were implemented for the understanding and use of the robotic unit.*

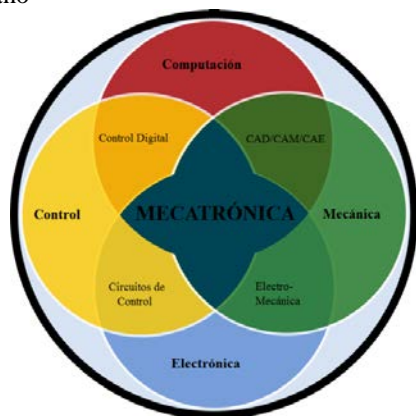
*Finally, we expect each student to take full understanding and familiarity of each of the components and sensors on the robotic unit.*

**Keywords:** guides, practices, manual, laboratory, mechatronics, Robotino ®, updating, management, use.

# 1. Introducción.

## 1.1. Mecatrónica.

La mecatrónica y su ingeniería está compuesta por la unión de las disciplinas: mecánica, electrónica, control e informática (*Figura 1.1-1*); enfocándose en el diseño y desarrollo de productos que involucren estas áreas. Con lo que se busca crear máquinas más complejas y precisas para facilitar las actividades del ser humano



**Figura 1.1–1: Ramas que conforman la ingeniería mecatrónica**

Esta rama tiene sus inicios en los estudios realizados por Alan Mathison Turing (*Figura 1.1-2*) en el área de la cibernética, quien es considerado como uno de los padres de la ciencia de la computación y precursor de la informática moderna, proporcionando una influyente formalización de los conceptos de algoritmo y computación.

Las primeras máquinas creadas las controlaba su operador, pero con el comienzo de la industrialización, estas necesitaban ser más rápidas y eficientes por lo que su control se fue dificultando, empezando a necesitar cada vez actuadores, sensores más sensibles y precisos, los mismos que se han ido desarrollando hasta la actualidad donde se están usando micro-sensores y micro-actuadores, los cuales han servido para miniaturizar aún más las máquinas modernas, volviéndolas más versátiles y portables con bajo costo de energía, teniendo como consecuencia un aumento considerable de su eficiencia.

## 2. Generalidades

El propósito de este trabajo es elaborar nuevos manuales de uso y guías de práctica para el sistema didáctico robótico móvil, debido a que las anteriores quedaron obsoletas por una actualización del sistema operativo que recibió la unidad, que forma parte del laboratorio de Mecatrónica de la Facultad de Ingeniería en Mecánica y Ciencias de la Producción (FIMCP-ESPOL).

También en el presente escrito se exponen las generalidades del sistema robótico de nombre *Robotino® de Festo®*, su uso y manipulación.

Además se explica la programación para el sistema en Robotino®view 2 y Matlab® 7

Por último se incluyen las guías actualizadas y reformuladas en Robotino®view 2 y las prácticas nuevas programando en MatLab® para el laboratorio.

### 2.1. Objetivos generales

Proporcionar a la FIMCP-ESPOL un manual de uso y prácticas de laboratorio que le permita afianzar a cada estudiante los conocimientos adquiridos en clase.

### 2.2. Objetivos específicos

- ✓ Presentar de manera clara, explicativa e interactiva el uso y manejo de la unidad y cada uno de los componentes de la unidad robótica.
- ✓ Explotar cada una de las cualidades que tiene la unidad.
- ✓ Afianzar en cada estudiante los conocimientos adquiridos en el aula de clase.
- ✓ Establecer las aplicaciones practicas que tiene en la industria, sensores y actuadores en la robótica y automatización de los procesos.

### 2.3. Metodología aplicada

Para realizar las prácticas se toma en cuenta el nivel académico del estudiante, debido a que, se requiere que el estudiante tenga conocimientos previos de programación con diagramas de bloques y pseudo-código.

No obstante se ha elegido una secuencia en las guías, de tal manera que sea adaptativa en la curva de aprendizaje del estudiante.

### 2.4. Estructura general del trabajo



### 3. Marco Teórico

#### 3.1. Sistema didáctico robótico móvil.

La Facultad de Ingeniería en Mecánica y Ciencias de la Producción (FIMCP-ESPOL) en el equipamiento de su nuevo laboratorio de mecatrónica, adquirió, un sistema robótico denominado Robotino® de la marca FESTO® (veáse figura 3.1-1), el cual ha sido desarrollado netamente con el propósito de la formación y perfeccionamiento profesional en el área de automatización, robótica y tecnología.



Figura 3.1-1: Sistema didáctico robótico móvil

Con este sistema, se persigue que cada estudiante se familiarice con las diferentes clases de sensores y actuadores que éste contiene. Al igual de su uso y aplicaciones prácticas que se pueden hacer con cada uno de ellos.

Robotino® se propone como un sistema amigable y abierto, donde cada alumno o usuario tiene acceso a sus componentes y tecnología, para ser modificados o dispuesto de la manera más conveniente o requerida, volviéndose divertido. También, al promover la competencia o concurso para establecer maneras más óptimas y/o eficientes en su programación y control, obteniendo soluciones semejantes a las que se utilizan en la industria, ya que está compuesto por elementos que se emplean en la misma.

#### 3.2. Reconocimiento de los componentes del sistema didáctico robótico móvil.

El S.D.Ro.M. Robotino® al igual que robots industriales incluye diversos sistemas parciales o componentes que tienen funciones diversas e independientes.

En la Tabla 3.2-1 se muestran los componentes del S.D.Ro.M. Robotino® y en la figura 3.1-1 donde están situados para su rápido reconocimiento.

1. Cámara
2. Puente de mando
3. Motor
4. Encoder incremental
5. Sensor anti-colisión
6. Sensor de medición de distancia
7. Interfase E/S
8. Rodillos omnidireccionales
9. Unidad de fibra óptica
10. Cargador de batería
11. Batería
12. Fibra óptica
13. Chasis
14. Plug de carga
15. Conector VGA
16. Interfase Ethernet
17. Puerto USB

Tabla 3.2-1: Principales Componentes del S.D.Ro.M. Robotino®

#### 3.3. Uso, control y puesta en marcha del sistema didáctico robótico móvil.

El S.D.Ro.M. Robotino® dispone en el puente de mando, de un teclado de membrana y una pantalla de diodos luminosos (Figura 3.3-1), donde es posible manejar y revisar varios parámetros del mismo; como: el estado de carga de las baterías, dirección IP de Robotino®, versión de sistema operativo, entre otros.

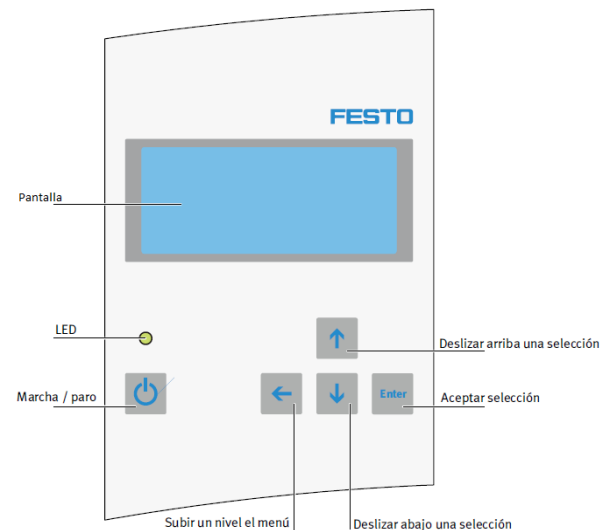


Figura 3.3-1: Control de mando manual de Robotino®

Además el S.D.Ro.M. activa una red inalámbrica “Wi-fi” una vez encendido, lo que permitirá una conexión remota entre un computador y el robot, para poder controlar la unidad desde el ordenador, mediante el programa Robotino®view 2 u otro compatible con el sistema.

### 3.4. Actualización sistema operativo del sistema didáctico robótico móvil

Cuando la FIMCP adquirió el S.D.Ro.M. Robotino®, vino con la versión 1.7 en su sistema operativo y con las continuas mejoras en software que se dan con el transcurso del tiempo, Festo®didactic la empresa que fabrica los Robotinos® ofreció una nueva versión (2.xx) del S.O. Con lo que se comenzó a trabajar en esta actualización, que no solo involucraba el cambio de sistema operativo sino también la modificación de las prácticas establecidas hacia al nuevo programa de control del S.D.Ro.M.

Esta actualización se hizo, siguiendo las instrucciones de la página del fabricante, debido a que se tenían hacer pasos estrictos con un programa especial.

Este método especial para actualizar solo se lo realizará una vez, puesto que la empresa Festo®, cambio el procedimiento de actualización para las nuevas versiones que se presenten; ahora estas se harán directamente desde el programa Robotino®View 2.

## 4. Software De Comunicación Y Control Del Sistema Didáctico Robótico Móvil.

El S.D.Ro.M. Robotino®, trae consigo un software llamado Robotinoview®, mediante el cual puede ser programado y controlado, pero para usuarios más experimentados existen varios lenguajes de programación y software, como son: C, C++, Java, .NET, MatLab®, Simulink, Labview y Microsoft Robotics Developer Studio, con los que también, se puede lograr la comunicación y control de la unidad robótica

Este trabajo se dirige a enseñar el uso general y programación en Robotinoview® (versión 2.x) y Matlablab® 7 que es un programa ampliamente usado por la ESPOL.

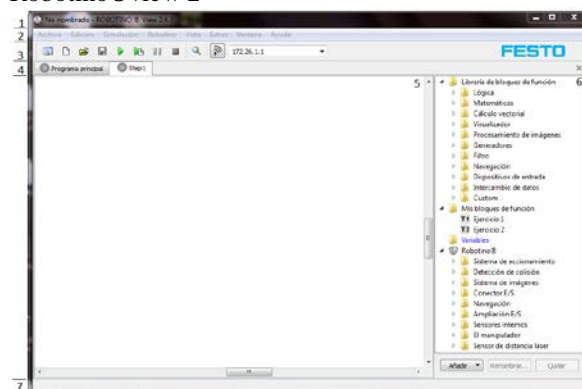
### 4.1. Robotino®view 2.

Robotino®view ofrece una interface o espacio de trabajo amigable con el usuario, que permite una rápida familiarización del mismo.

La nueva versión (2.x) contiene cambios muy importantes como: actualizaciones automáticas, ejecución de programas en esquemas Graficet, control simultáneo de varios Robotinos, nuevos bloques funcionales, ejecución de programas de Robotino®view 2 directamente desde el S.D.Ro.M. Robotino®

Además el programa de control de secuencia es reemplazado por un programa de control “real”, conocido de la programación de PLC según DIN EN 61131.

Al correr el programa, éste abre un espacio vacío para iniciar un proyecto nuevo, listo para comenzar a trabajar en la programación y control de la unidad. La Figura 4.1–1 muestra una impresión de pantalla de Robotino®view 2



**Figura 4.1–1: Interface de usuario de Robotino®view: (1) Barra de título; (2) Barra de menú; (3) Barra de herramientas; (4) Pestañas de selección de programas; (5) Espacio de trabajo del subprograma; (6) Librería de bloques de funciones; (7) Barra de estado**

### 4.2. MatLab® 7 para del sistema didáctico robótico móvil.

MatLab® 7, un programa ampliamente usado en la Escuela Politécnica del Litoral y en muchas otras universidades a nivel nacional y mundial, en ámbitos académicos y de igual manera, por personas en diferentes áreas empresariales.

MatLab® 7 derivado de la abreviatura MATrix LABORatory (Laboratorio de matrices) es un software matemático con lenguaje de programación propio (lenguaje M), ofreciendo un entorno de desarrollo integrado o IDE por sus siglas en inglés, lo que significa que es un programa informático compuesto por un conjunto de herramientas de programación, que consisten en un “editor de código”, “un compilador”, “un depurador” y un “constructor de interfaz gráfica (GUI)”. y esta disponible para las plataformas Unix Windows y Mac OS

Entre sus prestaciones básicas se hallan: la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuarios y la comunicación con programas en otros lenguajes y con otros dispositivos hardware. Dispone también de dos herramientas que expanden sus prestaciones: “Simulink” (plataforma de simulación multidominio) y “GUIDE” (editor de interfaces de usuario). Además se puede ampliar sus capacidades con cajas de herramientas “Toolboxes” o paquetes de bloques “blocksets” para Simulink.

Y dada a la importancia que tiene MatLab®, se han creado los controladores que permiten establecer

la comunicación y operar la unidad por medio del mismo.

La programación en MatLab®, se hará con pseudocódigo y la ayuda de paquetes de bloque y cajas de herramientas, que se instalan junto con los drivers de Robotino® para MatLab®

## **5. Desarrollo De Nuevas Prácticas De Laboratorio Consideradas.**

### **5.1. Robotino®view 2: Control de las comunicaciones, puesta en funcionamiento y programación de movimientos lineales en sentidos indistintos del sistema robótico.**

Cuyos objetivos son:

- ✓ Conocer los componentes más importantes de un Sistema Didáctico Robótico Móvil (S.D.Ro.M.).
- ✓ Poner en funcionamiento, probar y explicar los movimientos que ejecuta el S.D.Ro.M. Robotino® y los grados de libertad que posee, mediante Robotino®View 2.
- ✓ Describir y programar movimientos sencillos con bloques de funciones y secuencias en Robotino®View 2; tomando en cuenta aspectos de seguridad en el caso de una colisión del S.D.Ro.M.

### **5.2. Sensores de reflexión directa: programación del movimiento guiado del sistema robótico.**

Cuyos objetivos son:

- ✓ Aprender sobre el montaje y conexión de los sensores de reflexión directa.
- ✓ Evaluar y calibrar las señales de un sensor de reflexión directa.
- ✓ Desarrollar una estrategia para la ejecución de movimientos guiados de un sistema de transporte sin conductor, elaborando un programa para aplicar, probar y optimizar las ideas planteadas

### **5.3. Detector analógico inductivo: comportamiento frente a diversos tipos de materiales y programación del movimiento guiado del sistema robótico.**

Cuyos objetivos son:

- ✓ Aprender sobre el montaje y conexión de un detector analógico inductivo.
- ✓ Evaluar y calibrar las señales de un detector analógico inductivo.
- ✓ Desarrollar un método para reconocer la diferencia entre tipos de materiales metálicos
- ✓ Desarrollar una estrategia para la ejecución de movimientos guiados de un sistema de transporte sin conductor.

- ✓ Elaborar un programa secuencial para aplicar, probar y optimizar las ideas planteadas en el punto anterior.

### **5.4. Detectores de distancias infrarrojos: línea característica (curva de calibración), avance en función de distancias precisas y mantenerlas, bordear o evitar obstáculos con el sistema robótico.**

Cuyos objetivos son:

- ✓ Conocer el comportamiento y el funcionamiento de los detectores de distancia por luz infrarroja del S.D.Ro.M.
- ✓ Determinar y registrar la línea de calibración de los detectores de distancia y saber como usarla para relacionarla con la medición distancias.
- ✓ Elaborar un programa de regulación donde la unidad sea capaz de acercarse a un obstáculo y mantener una distancia definida respecto al mismo.
- ✓ Elaborar varias estrategias para que Robotino® mantenga distancias ante un obstáculo y/o lo bordee.

### **5.5. Motores: control, calibración movimiento lineales y posicionamiento del sistema robótico.**

Cuyos objetivos son:

- ✓ Obtener conocimientos básicos acerca de la técnica de regulación de motores eléctricos.
- ✓ Ajustar y optimizar los parámetros de un regulador P.I.D. mediante el software correspondiente.
- ✓ Describir y evaluar los efectos que tiene el ajuste de las constantes del control P.I.D. en los movimientos ejecutados por el robot.

### **5.6. Cámara digital: detección de una pieza de diferentes colores y ejercicios de aplicación.**

Cuyos objetivos son:

- ✓ Conocer el funcionamiento y utilización de la cámara de Robotino®.
- ✓ Configurar las funciones de detección de colores de Robotino®View y saber usarlas.
- ✓ Conocer las limitaciones y condiciones que se tienen la detección de colores y objetos.

### **5.7. Movimiento guiado del sistema robótico mediante el uso de una cámara digital**

Cuyos objetivos son:

- ✓ Conocer el funcionamiento y utilización de la cámara de Robotino®.
- ✓ Configurar y usar el bloque de función "Detector de líneas".



- ✓ Lograr movimientos guiados usando la cámara de Robotino®

### **5.8. MatLab® 7: control de comunicaciones, puesta en funcionamiento y programación de tareas básicas del sistema robótico.**

Cuyos objetivos son:

- ✓ Establecer la conexión y el control de comunicaciones de Robotino® mediante MatLab®7
- ✓ Poner en funcionamiento, probar y explicar los movimientos que ejecuta el S.D.Ro.M. Robotino® y los grados de libertad que posee, mediante MatLab®7.
- ✓ Describir y programar movimientos sencillos con las cajas de herramientas “ToolBoxes” en MatLab®7.

### **5.9. Detectores de distancia infrarrojos: programación de sistema de exploración a través de MatLab® 7.**

Cuyos objetivos son:

- ✓ Conocer la programación y utilización de los sensores de distancia de Robotino® en MatLab® 7.
- ✓ Configurar y usar las cajas de herramientas “DistanceSensor”.
- ✓ Elaborar un programa de exploración donde Robotino® se acerque a los obstáculos y los evite.

### **5.10. Detector analógico inductivo: programación de movimiento guiado en MatLab® 7.**

Cuyos objetivos son:

- ✓ Conocer la programación y utilización de los sensores de inducción de Robotino® en MatLab® 7.
- ✓ Configurar y usar las cajas de herramientas “AnalogInput”.
- ✓ Elaborar un programa de exploración donde Robotino® se acerque a los obstáculos y los evite rodeándolos.

### **5.11. Sensores de reflexión directa: programación de movimiento guiado en MatLab® 7 mediante bloques en Simulink.**

Cuyos objetivos son:

- ✓ Conocer la programación y utilización de bloques en Simulink para los sensores de distancia de Robotino® en MatLab® 7.
- ✓ Configurar y usar los bloques de Simulink “DigitalInput0”.

- ✓ Elaborar un programa de exploración donde Robotino® siga el trayecto de una línea negra previamente marcada.

## **6. Conclusiones y Recomendaciones**

1. En conclusión, este trabajo deja como resultado, guías de laboratorio, ya probadas por estudiantes de la asignatura “Sistemas de Control”, logrando un mayor entendimiento y afianzamiento de ideas, de lo dictado en clases.
2. Además se desea promover que el alumno se motive con el autoaprendizaje, a buscar mayores conocimientos y desarrollos para su carrera.
3. Se recomienda seguir, el orden que se ha establecido en las guías, debido a que van acorde a un ritmo escalado de complejidad; además que, siguen el plan actual de lo que se dicta en clases de Sistema de Control.
4. Robotino® es un dispositivo robusto, fabricado de materiales de alta resistencia, pero de todas formas es necesario extremar sus cuidados; que por ser un equipo electrónico, poseedor de elementos sensibles y delicados, se aconseja que antes de su manipulación, se dedique un tiempo a examinar la unidad, identificar sus componentes y proponer acciones a tomar frente a los mismos.
5. Finalmente, esta tesis fue hecha para introducir al alumno al uso y control de un sistema robótico, por lo que se recomienda, explotar al máximo todas las posibilidades que tiene la unidad, realizando nuevas prácticas a manera de proyectos o combinando las ya existentes, para ampliar o complementar las tareas y aplicaciones que se le puede dar la unidad.

## **13. Agradecimientos**

A Dios, en primer lugar, por otorgarme la existencia y permitirme llegar a donde estoy.

A mi madre, por ser una guía a lo largo de mi vida y por toda la paciencia y abnegación que me tiene.

A mi padre por todos sus sacrificios, dedicación y la atención hacia la familia.

Y por último a todas las personas que de una u otra forma colaboraron con la realización de este trabajo, en especial a mi Director de Tesis Ing. M.Sc. Eduardo Orces Pareja, por su invaluable ayuda.

## **14. Referencias**

- [1] Ruiz Rojas, Paola Andrea. “Mecatrónica, Revolución para el siglo XXI”  
[http://www.metalactual.com/revista/8/tecnologia\\_mecatronica.pdf](http://www.metalactual.com/revista/8/tecnologia_mecatronica.pdf)

- [2] Arbeláez Salazar; Mendoza Vargas, Osiel; Jairo A. (16 de Julio de 2007). “La ingeniería Mecatrónica por ciclos en Colombia”  
<http://www.utp.edu.co/php/revistas/ScientiaEtTechnica/docsFTP/122935421-426.pdf>
- [3] Alan Turing: Life and Legacy of a Great Thinker, C. Teuscher (Ed.), ISBN 3-540-20020-7 (Springer-Verlag, 2004)
- [4] IES Alfonso Romero Barcojo, Departamento de Tecnología, Unidad Didáctica: Control y robótica. Sección: Sensores.  
<http://cmapspublic2.ihmc.us/rid=1H2F1807L-JP0SG2-J1J/encoder.pdf>
- [5] <http://wiki.openrobotino.org/>
- [6] Introducción a la programación con Matlab por Ángel Garcimartín-Departamento de Física y Matemática Aplicada  
<http://fisica.unav.es/~angel/matlab/matlab1.html>
- [7] Ajuste empírico de controladores PID: método de ziegler-nichols (.pptx)  
Electrónica y sistemas de control - Dpto. Ingeniería mecánica; frsf-utn
- [8] Richard O. Duda, Peter E. Hart, David G. Stork (2001): Pattern classification (2ª edición), Wiley, New York
- [9] Dietrich Paulus and Joachim Hornegger (1998): Applied Pattern Recognition (2ª edición), Vieweg.