





## Unidad 6 Control y robótica

### 1. Sensores

Los **sensores** son dispositivos que se encargan de detectar las condiciones del entorno: temperatura, humedad, luz, movimiento, caudal, etc.

Clasificación de los sensores			
SENSOR	VARIABLE DEL SENSOR	MAGNITUD QUE DETECTA	SÍMBOLO
Final de carrera	Posición de una lengüeta	Presión de contacto	
LDR	Resistencia	Intensidad luminosa	
PTC	Resistencia	Temperatura	
Higrómetro	Resistencia	Humedad	
Infrarrojos	Radiación infrarroja	Luz infrarroja	

### 2. Sistemas de control y robots

Los **sistemas de control** son los encargados de tomar determinadas acciones en función de una magnitud determinada. Estos sistemas pueden ser sin realimentación (no consideran las variaciones del entorno) y con realimentación (tienen en cuenta las variaciones del entorno mediante sensores).

#### Sistemas de control sin realimentación o de lazo abierto



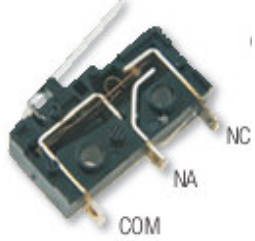
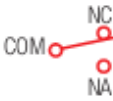
#### Sistemas de control con realimentación o de lazo cerrado



Un **robot** es una máquina automática con capacidad de movimiento que puede responder y reaccionar a las condiciones del entorno y desarrollar diferentes tareas según sea programado.



### 3. Sensor de contacto. Interruptor de final de carrera

Los **sensores de contacto** se utilizan para detectar el final del recorrido de determinados componentes mecánicos móviles, como pueden ser puertas, barreras o ascensores, o para determinar la posición límite de elementos móviles de máquinas.

Interruptor de final de carrera	Símbolo	Funcionamiento	Aplicaciones
		Los interruptores de final de carrera se disponen en los circuitos de tal manera que los mecanismos móviles, cuya posición se quiere controlar, presionan la lengüeta cuando llegan a sus posiciones límites.	Por ejemplo, para detener el movimiento de un ascensor en un piso.



### 4. Sensor magnético. Interruptor *reed*

Los **sensores magnéticos** se utilizan para detectar campos magnéticos producidos por corrientes eléctricas o por imanes. Uno de los sensores magnéticos más utilizados es el interruptor *reed*.

Interruptor <i>reed</i>	Símbolo	Funcionamiento	Aplicaciones
		Cuando el interruptor <i>reed</i> se encuentra en las cercanías de un campo magnético, este hace que los extremos de las láminas metálicas adquieran polaridades de distinto signo, lo que provoca que las láminas se atraigan y cierren el circuito.	Para detectar las posiciones de partes móviles de mecanismos sin necesidad de contacto. En la detección del nivel de líquidos en depósitos.





### 5. Sensor de humedad

Los **sensores de humedad** se utilizan para detectar cambios de humedad.

Sensor de humedad	Símbolo	Funcionamiento	Aplicaciones
		Los cables se comportarán como un interruptor abierto y no circulará corriente cuando estén en contacto con el aire. En contacto con el agua harán que circule una pequeña corriente, cerrando el circuito.	Para el control de nivel en depósitos de agua, en un sistema de riego o en un sistema de alarma para evitar inundaciones.



## 6. Sensor de temperatura. Termistores NTC y PTC

Los **sensores de temperatura** detectan las variaciones de esta. Se construyen a partir de unos materiales cuya resistencia varía en función de la temperatura.

Sensores de temperatura	Símbolo	Funcionamiento	Aplicaciones
<p><b>NTC</b></p> 		<p>El valor de la resistencia varía de modo inversamente proporcional a la temperatura, es decir, si aumenta la temperatura, disminuye la resistencia, y viceversa.</p>	<p>Para sistemas de control de temperatura en sistemas de calefacción y aire acondicionado, o en sistemas de alarma contra incendios.</p>
<p><b>PTC</b></p> 		<p>El valor de la resistencia varía de modo directamente proporcional a la temperatura, es decir, si sube la temperatura, también aumenta la resistencia.</p>	


## 7. Sensor de luz. Resistencia LDR

Los **sensores de luz** se construyen a partir de unos componentes (selenio, sulfuro de cadmio y sulfuro de plomo) cuya resistencia varía en función de la cantidad de luz que reciben. Su nombre es **LDR**.

Sensor de luz	Símbolo	Funcionamiento	Aplicaciones
		<p>El valor de la resistencia varía de modo inversamente proporcional al nivel de luz que haya en su entorno, es decir, si aumenta la luz, baja la resistencia, y viceversa.</p>	<p>Para el control de apertura y cierre de puertas, para alarmas o para el control de iluminación.</p>

## 8. Sensores de infrarrojos. Optocopladores

Los **sensores de rayos infrarrojos** se utilizan para la detección de una zona del espectro electromagnético que no es visible para el ojo humano.

Sensor de infrarrojos	Funcionamiento	Aplicaciones
	<p>Está formado por un emisor y un receptor de luz infrarroja que pueden estar encapsulados en un mismo componente (optocoplador) o ser independientes.</p>	<p>Para mandos a distancia o como detector de paso para abrir o cerrar puertas.</p>

## 9. El lenguaje de programación MSWLogo

**MSWLogo** es un *software* gratuito que permite hacer programas en un lenguaje sencillo mediante órdenes que el programa entiende y ejecuta.

Las **primitivas** son las órdenes o instrucciones básicas para hacer los programas en MSWLogo. Cada primitiva va asociada a una acción mediante una sintaxis determinada.

PRIMITIVA	ABREVIATURA	ACCIÓN
<b>avanza</b>	<b>av</b>	Dibuja una línea.
<b>retrocede</b>	<b>re</b>	Dibuja una línea hacia atrás.
<b>giraderecha</b>	<b>gd</b>	La tortuga gira a la derecha el número de grados que se indica.
<b>giraizquierda</b>	<b>gi</b>	La tortuga gira a la izquierda el número de grados que se indica.
<b>subelapiz</b>	<b>sl</b>	La tortuga no dibuja.
<b>bajalapiz</b>	<b>bl</b>	La tortuga dibuja por donde pasa.
<b>pongrosor [0 n]</b>	<b>pong [0 n]</b>	Dibuja una línea de grosor <i>n</i> .
<b>circulo</b>	-	Dibuja un círculo del tamaño indicado.
<b>goma</b>	-	Borra por donde pasa la tortuga.
<b>ponlapiz</b>	<b>pla</b>	Después de utilizar la goma, para que la tortuga vuelva a escribir hay que poner de nuevo el lápiz.
<b>ocultatortuga</b>	<b>ot</b>	Oculto la tortuga.
<b>muestratortuga</b>	<b>mt</b>	Muestra la tortuga.
<b>repite n [a]</b>	-	Repite la acción entre corchetes <i>n</i> veces.

## 10. El color en MSWLogo

- Aplicar color con primitivas

PRIMITIVA	ABREVIATURA	ACCIÓN
<b>poncolorlapiz</b>	<b>poncl</b>	Color con el que dibujará la tortuga.
<b>poncolorpapel</b>	<b>poncp</b>	Cambia el color del fondo.
<b>rectangulorrelleno an al [color]</b>	-	Dibuja un rectángulo de ancho <i>an</i> y alto <i>al</i> del color indicado.
<b>poncolorrelleno</b>	<b>poccr</b>	Rellena de color el espacio en el que se encuentra la tortuga. Es necesario poner después del color la primitiva rellena.

## 11. Ventanas y botones

En la ventana de MSWLogo se pueden **crear ventanas y botones** que realizan acciones de control.

- Crear ventanas y botones.

PRIMITIVA	ACCIÓN	EJEMPLO
<b>creaventana</b>	Crea una ventana.	creaventana "botones [control] 50 30 60 80 []
<b>creaboton</b>	Crea un botón.	creaboton "botones "línea [recta] 30 10 20 20 [av 50]

- Borrar ventanas y botones

PRIMITIVA	ACCIÓN	EJEMPLO
<b>borraventana</b>	Borra una ventana.	borraventana "botones
<b>borraboton</b>	Borra un botón.	borraboton "línea

## 12. Procedimiento, variable y recursividad

Un **procedimiento** es un pequeño subprograma formado por un conjunto de primitivas.

Una **variable** es un dato, numérico o de texto, que toma distintos valores.

La **recursividad** es la capacidad que tiene un programa creado con MSWLogo para llamarse a sí mismo en un bucle sin fin.

## 13. Simulación de luces

PRIMITIVA	ABREVIATURA	ACCIÓN
<b>espera n</b>	-	El programa se para durante un tiempo determinado $n$ . El número 1 corresponde a $\frac{1}{60}$ segundos y 60 corresponde a 1 segundo.
<b>activa</b>	<b>ac</b>	El programa permite trabajar con varias tortugas. La que utiliza por defecto es la número 0. Cuando trabajemos con varias tortugas, cada vez que queramos que una de ellas haga algo hay que escribir la primitiva <b>activa</b> y a continuación su número.

#### 14. Simulación de movimiento

PRIMITIVA	ACCIÓN
<b>azar x</b>	Genera un número aleatorio* entre 0 y $x$ .
<b>si condicion=[a] [b]</b>	Si se cumple la condición [a], entonces realizará la acción [b].
<b>siempre []</b>	La tortuga realiza indefinidamente la acción que hay entre corchetes.

#### 15. Movimiento por circuitos

PRIMITIVA	ABREVIATURA	ACCIÓN	ORDEN COMPLETA PARA PROBAR
<b>cargadib "nombre"</b>	-	Carga un dibujo en la ventana de gráficos.	cargadib "f1a.bmp"
<b>ponfoco []</b>	-	Nombre de la ventana que queremos tener activa.	ponfoco [pantalla de MSWLogo]
<b>ponteclado []</b>	-	Al pulsar cualquier letra del teclado se ejecuta la acción que hay entre corchetes.	ponteclado [av 100]
<b>ponrumbo g</b>	<b>ponr</b>	Orienta la tortuga en función de los grados que se asignen a $g$ .	ponrumbo 90
<b>caracter</b>	<b>car</b>	Devuelve el carácter que corresponde a un código ASCII (número), entre 0 y 127.	car
<b>leecar</b>	<b>lc</b>	Esta primitiva devuelve el valor de la tecla presionada o liberada.	lc

#### 16. Trabajar con una controladora

La **controladora** es un dispositivo que permite la comunicación entre el ordenador y el mecanismo que se quiere controlar, como puede ser un puente levadizo, un ascensor, etc.

Las primitivas para controlar las acciones de la controladora con MSWLogo son las siguientes:

PRIMITIVA	ACCIÓN
<b>conecta n</b>	Conecta la salida digital $n$ .
<b>apaga n</b>	Desconecta la salida digital $n$ .
<b>esperaon n</b>	Espera a que se conecte la entrada digital $n$ .
<b>esperaoff n</b>	Espera a que se desconecte la entrada digital $n$ .
<b>leeanalogica n</b>	Lee el valor de la entrada analógica $n$ .
<b>salidaanalogica n x</b>	Da a la salida analógica $n$ el valor $x$ .