# TUTORIAL

Pin8, sistema de educación robótico









# ACERCA DE ROBÓTICA ALTERNATIVA



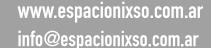


Robótica Alternativa es un proyecto que indaga en una nueva mirada sobre la educación robótica, poniendo el énfasis en la idea de la autoproducción de las herramientas robóticas como un proceso pedagógico y emancipador de la cultura de consumo desmedido.

Este proyecto es en parte resultado de la investigación "Una alternativa robótica para la Educación", proyecto de la convocatoria Untref 2018-2019.













Robótica Alternativa surge de la experiencia transitada en espacios alternativos de educación en arte y tecnología, donde el foco esta puesto en incentivar las capacidades creativas y diversas de cada uno de los estudiantes. Es por ello que abogamos por un proyecto donde se libere el desarrollo creativo personalizado, fomentando el conocimiento y la apropiación crítica de las herramientas tecnológicas. En contraposición a muchos productos robóticos del mercado que pretenden homogeneizar la imaginación con sistemas desarrollados en otras culturas, este proyecto indaga en la posibilidad del desarrollo de un diseño robótico con identidad regional. Solamente experimentando con nuestra propia cultura material y simbólica se lograra en un futuro alcanzar nuestros propios desarrollos tecnológicos.

Este proyecto plantea la posibilidad de desarrollar actividades para la construcción de robots con distintos niveles de dificultad, favoreciendo la inclusión socieducativa a un amplio rango de edades de estudiantes y con una diversidad de contenidos educativos. Puede experimentarse con el Hardware, con el Software y con la materialidad de su diseño. Estos tres pilares son abiertos, modificables, mejorables, personalizables y compartibles. Es una búsqueda estética en todos sus aspectos.

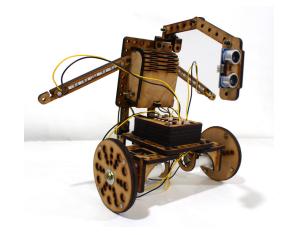
Con esta visión, desarrollamos un sistema robótico flexible y permite incluir a estudiantes de los todos los niveles (primario, secundario y universitario), estimulando el pensamiento computacional, la lógica, el manejo de la complejidad y el trabajo colaborativo. Además puede ser utilizado en las aulas para afianzar conocimientos de física, matemática y mecánica, como también de plástica, biología y contenidos vinculados a los campos tradicionales del saber. Es una herramienta creativa de aprendizaje transversal a los contenidos de la currícula. Un espacio de aprendizaje, juego, encuentro y colaboración dentro y fuera del aula.





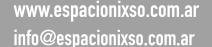


## **ACERCA DE** Pin8





Es un sistema de educación robótico aplicable a distintos niveles educativos. El proyecto se establece dentro del pensamiento Open Design, Open Hardware y Open Software. Propone la búsqueda de una identidad robótica. PIN8 es el último modelo del proyecto. Desarrollado completamente en placa de madera MDF, contiene todos los componentes básicos para la construcción del robot. (controladores, puente H, motores, led, sensores US, IR y LDR, fuente de energía). Los planos, listado de materiales y software se pueden descargar de la web del proyecto. Todas las piezas estructurales son intercambiables con otro tipo de materiales y objetos en desuso (caños, basura electrónica, electrodomésticos, botellas, etc), lo que habilita a una infinidad de configuraciones formales y materiales.

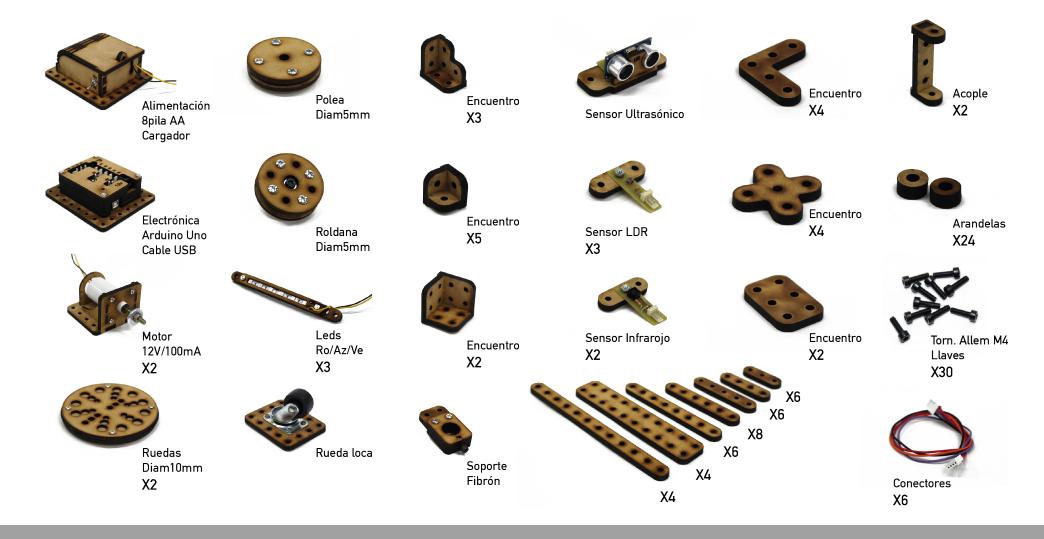








#### Componentes básicos de Pin8





## ¿Cómo construir Pin8?









# PASO Nº 1 MATERIALES

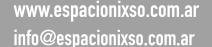
#### TUTORIAL SISTEMA DE EDUCACIÓN ROBÓTICO PIN8

Para construir Pin8, necesitarás conseguir los siguientes materiales

#### **Componentes - Placas Pin8**

- Arduino Uno R3, Cantidad 1
- Placa de cobre de 10 x 10 cm. Cantidad 1
- Integrado Uln 2803. Cantidad 1
- Integrado l293D. Cantidad 1
- Zócalo de 18 patas. Cantidad 1
- Zócalo de 16 patas. Cantidad 1
- Capacitor Cerámico 0.1uf (104). Cantidad 5
- Capacitor 100uf . Cantidad 1
- Led 3mm. Cantidad 1
- Micro switch (tact switch), con 4 terminales y alto. Cantidad 1
- Switch de palanca SPTD 1. Cantidad 1
- Resistores 1K 1/4 Watt. Cantidad 1
- Resistores 100K 1/4 Watt. Cantidad 7
- Resistores 0 o puentes. Cantidad 4
- Diodo 4007, Cantidad 1
- Jack DC Hueco para Placa (2.1mm o 2.5 mm). Cantidad 1











- Tira de pines Macho Recta Simple Larga (25 mm). Cant. 2
- Conector Wafer 0.1 Recto Macho(Molex). 2 vías. Cant. 5
- Conector Wafer 0.1 Recto Macho (Molex). 3 vías. Cant. 5
- Conector Wafer 0.1 Recto Macho (Molex). 4 vías. Cant. 2
- Motorreductor 12v. Consumo máximo 500 ma. Cant.2
- Portapilas para 4 AA. Cantidad 2
- Pila recargables AA. Cantidad 8
- 1/2 metro tira leds. Cantidad 3

#### **Componentes - Conexiones**

- Conector Wafer 0.1 Hembra (Molex). 2 vías. Cant. 5
- Conector Wafer 0.1 Hembra (Molex), 3 vías, Cant.10
- Conector Wafer 0.1 Hembra (Molex). 4 vías. Cant. 2
- Terminales Crimpeable para Wafer 0.1. Cant. 60

#### **Componentes - Sensores LDR**

- Placa de cobre de 5 x 5 cm. Cantidad 1
- LDR 5mm, Cantidad 2
- Resistores 10K 1/8 Watt, Cantidad 2
- Conector Wafer 0.1 en L Macho (Molex). 3 vías. Cant. 2

#### **Componentes - Sensores Infrarojos**

- Placa de cobre de 5 x 5 cm. Cantidad 1
- CNY70. Cantidad 2
- Resistores 0.330K 1/4 Watt. Cantidad 4
- Resistores 100K 1/4 Watt, Cantidad 2
- Conector Wafer 0.1 en L Macho (Molex). 3 vías. Cant. 2

#### **Componentes - Sensor Ultrasónico**

- Conector Wafer 0.1 en L Macho (Molex), 3 vías, Cant. 1
- Sensor Ultrasonico Hc-sr04, Cant. 1

#### Herramientas:

Soldador, estaño, alicate, taladro eléctrico de banco, mecha de 1mm y 3mm, cloruro férrico, batea, plancha, papel transfer para circuitos electrónicos, impresora laser, computadora.









# PASO Nº 2 Armado de placas electrónicas

Para la construcción del proyecto **Pin8**, es necesario hacer las placas electrónicas, tanto para el control de los motores como para el de los sensores.

Los diseños estan disponibles dentro del material de descarga del proyecto. También encontrarán el orden de los componentes en la placa.

Para construir las placas les compartimos algunos tutoriales de cómo hacerlo. Son algunas referencias de miles que hay en la red:

https://www.youtube.com/watch?v=mdOkVgij6xQ

https://www.youtube.com/watch?v=sMFswuSh4FY

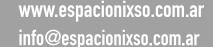
https://www.youtube.com/watch?v=MX9\_9U6Wnu8

https://www.youtube.com/watch?v=HFd5Gp93dFl

http://usuaris.tinet.cat/fmco/download/Tutorial\_placas.pdf

http://www.pablin.com.ar/electron/cursos/pcb









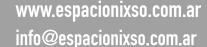
PASO N° 3

Soldar componentes
electrónicos

Realizar la soldaduras de los componentes en la placa es una tarea sencilla. Para aquellos que no tienen experiencia, les compartimos algunos videos tutoriales:

https://www.youtube.com/watch?v=qxlglm8d14c https://www.youtube.com/watch?v=6SrXD5OJj5c https://www.youtube.com/watch?v=Rvypcu0YIPA https://www.youtube.com/watch?v=Il5rsSkocP8 https://www.youtube.com/watch?v=ZrtpAn82yc0 https://www.youtube.com/watch?v=LSpmmEFwGKA

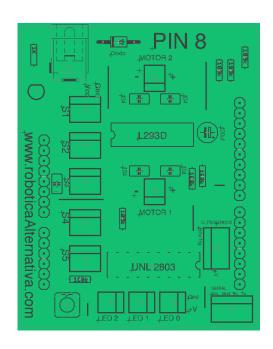
Se recomienda empezar con los componentes más pequeños ya que los más grandes pueden incomodar en el proceso de soldado.



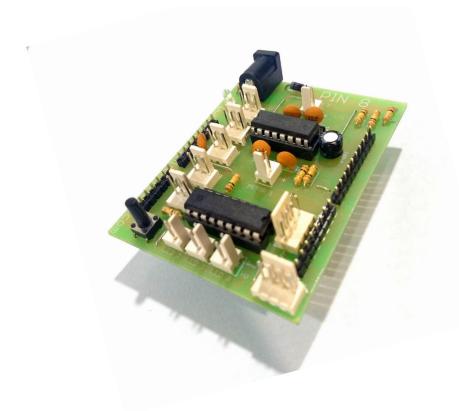








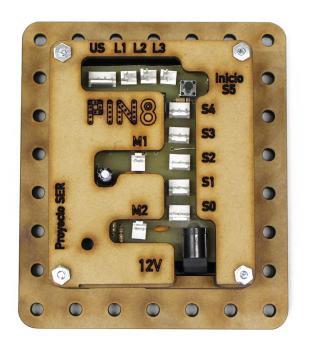
Vista frontal del esquema de distribución de los Componentes de la Placa de Control Pin8.

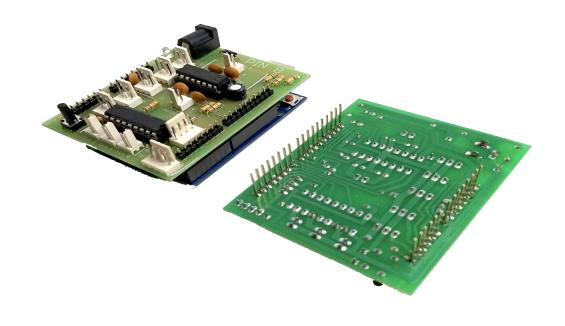


Vista frontal de la placa terminada









Vista frontal y posterior de la Placa de Control Pin8 terminada

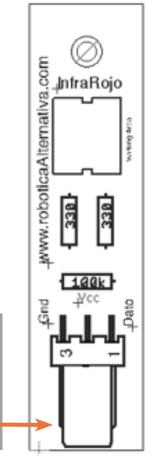








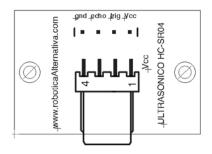
Vista frontal del esquema de distribución de los Componentes de Sensor Infrarrojo





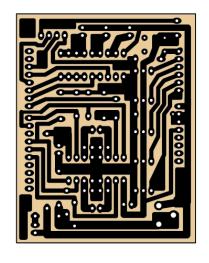


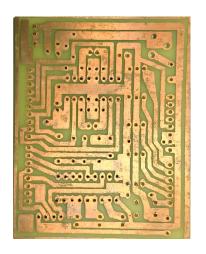




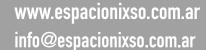


Vista frontal del esquema de distribución de los Componentes del Sensor Ultrasónico





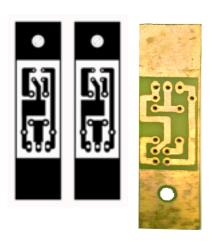
Vista posterior del esquema de pistas para imprimir y soldar la Placa de Control Pin8

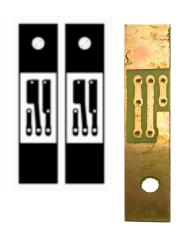








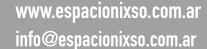








Vista posterior de la imagen para imprimir y soldar el Sensor Infrarrojo Vista posterior de la imagen para imprimir y soldar el Sensor LDR Vista posterior de la imagen para imprimir y soldar el Sensor Ultrasónico







# PASO Nº 4 MOTORES



#### Motorreductor 12Volt. - Eje roscado

Nosotros utilizamos motor tipo <u>AP&S</u> **SP4**, de 50rpm, con un consumo nominal de 100ma y picos de 400ma. Los mismos se consiguen con eje roscado de 6mm lo que facilita la vinculación con cualquier tipo de rueda.

Se puede utilizar cualquier motor que responda a los requisitos electrónicos, que no supere los 500ma de consumo y también hay que tener en cuenta el voltaje que asigmanos al sistema (en nuestro caso 12Volt.









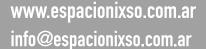
## PASO Nº 4.1.1 Armado de cables.

#### Crimpeado de cables

En este paso, usaremos unas pinzas de punta plana para crimpear los cables al Terminal metálico Molex tipo HUXX. Debemos colocar el terminal en uno de los extremos de ambos cables del motor.



La medida de los cables debe ser de unos 25 cm aproximadamente y en preferencia de distintos colores para identificar facilmente las conexiones.





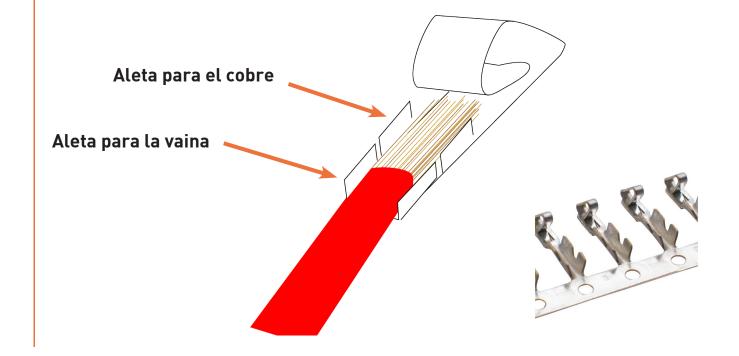


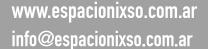


## PASO Nº 4.1.2 Armado de cables.

#### Crimpeado de cables

La vaina del cable debe estar sujetada por unas de las aletas. Y el cobre del cable debe estar apretado por las aletas internas haciendo contacto con el terminal.







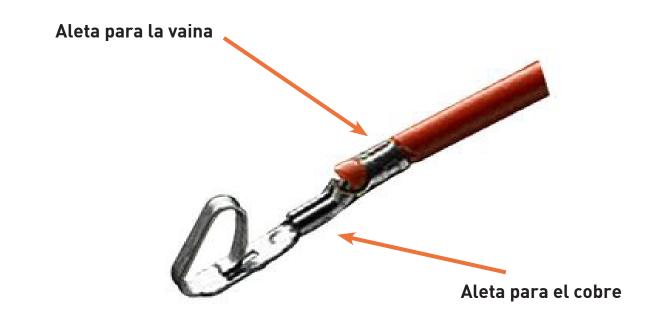




## PASO Nº 4.1.3 Armado de cables.

#### Crimpeado de cables

El cable debe quedar apretado en el terminal firmemente. Y el cobre del metal en contacto con el terminal. Intentar que el cobre del cable no sobresalga por la parte trasera del terminal y que los hilos de cobre entren en el tubo del terminal.











## PASO Nº 4.2.1 Armado de conectores

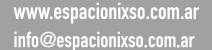


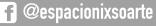


#### Conexión de Cables

El próximo paso es colocar los cables con los terminales en el conector de 2 vias. Cada cable debe ir en una posición determinada.

ATENCIÓN: Cuidado!!!, si no se conecta adecuadamente es probable que se rompa la electrónica.





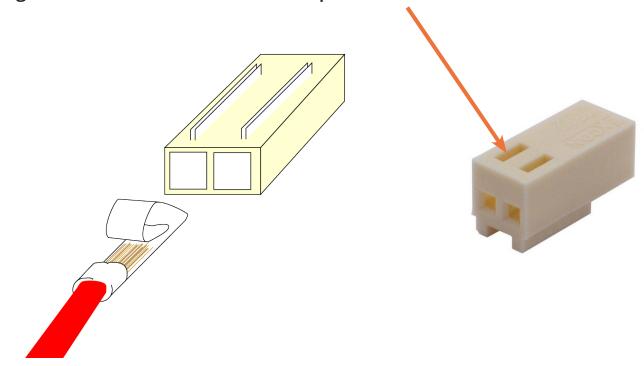


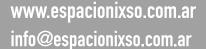


## PASO Nº 4.2.2 Armado de conectores

#### Conector/Ficha Molex 2 vías.

Colocar un cable en cada vía del conector. La parte plana del terminal va hacia debajo. La parte plana se encuentra con la perforación del conector. El terminal entra bien profundamente hasta que una **pequeña lengua del terminal** se encastre en la **perforación del conector**.









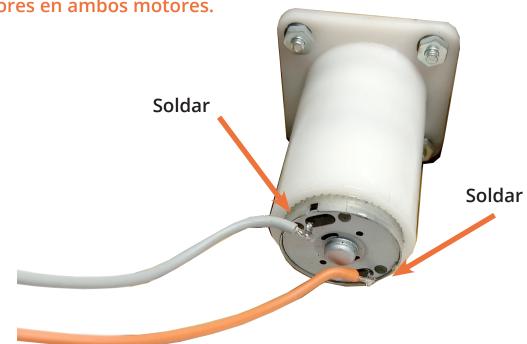


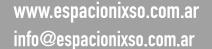
## PASO N° 4.3 Terminal del motor

#### Conectar los cables al motor

Ya con la ficha molex armada, vamos a soldar los extremos de los dos cables a los terminales del motor.

Atención!. Soldar los cables de forma que concuerde el orden de los colores en ambos motores.









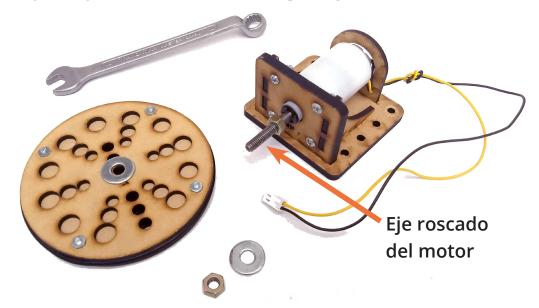


## PASO Nº 4.4 Vinculación Rueda

#### Vincular motor a rueda

Una de las ventajas de utilizar motor con eje roscado, es que nos permite utilar cualquier objeto circular como rueda. Podemos usar un disco de madera o un plato. También podemos descentrar la rueda con lo que lograremos modificar la calidad del desplazamiento del objeto.

Atención!. Al ajustar la tuerca al eje, tener cuidado de no forzar el motor, para que no se dañen los engranajes internos.









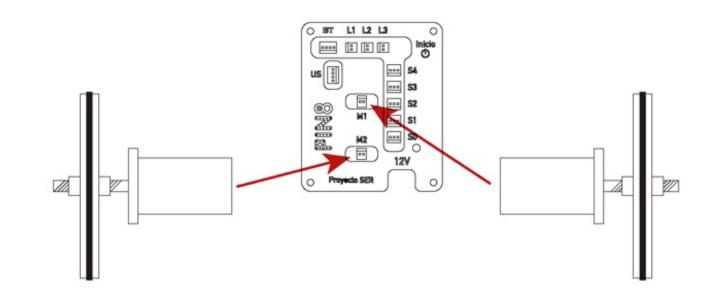


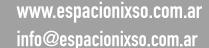
## PASO Nº 4.5 Conexión a la placa

#### Conectar los motores a la placa de Pin8

Las fichas de conexión para los motores en la placa, están indicadas con las siglas M1 y M2.

Atención!. Las conexiones M1 y M2 entregan aprox. 12Volt. Si van a conectar un motor de 5Volt tendrán que modificar la fuente de energía.









# PASO N° 5 Conexiones del Sensor Ultrasónico

(necesario solamente para la utilización de este sensor)





#### Conexiones del Sensor Ultrasónico.

Este sensor permite medir distancias, en centímetros, entre el robot y un obstáculo. En este sistema permite medir hasta un metro de distancia en una dirección recta al sensor. Mide dentro de un rango e 30 grados, es decir que el sensor es bástate dirección. Atención: el sensor reconoce obstáculos grandes ( por ejemplo no reconoce patas de sillas, si una pared ). Un típico error es que en sensor apunte al piso, colocarlo de forma que apunte hacia delante y levemente hacia arriba. Este paso es complejo por ello lo realizaremos de a pequeñas acciones.









## PASO Nº 5.1.1 Armado de cables.

#### Crimpeado de cables

En este paso, usaremos unas pinzas de punta plana para crimpear los cables al Terminal metálico Molex tipo HUXX. Debemos colocar el terminal en ambos extremos de los 4 cables.



La medida de los cables debe ser de unos 25 cm aproximadamente y en preferencia de distintos colores para identificar facilmente las conexiones.





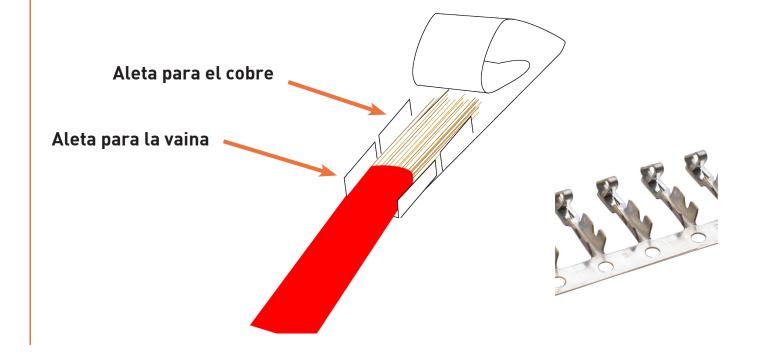




#### Crimpeado de cables

La vaina del cable debe estar sujetada por unas de las aletas. Y el cobre del cable debe estar apretado por las aletas internas haciendo contacto con el terminal.

## PASO Nº 5.1.2 Armado de cables.

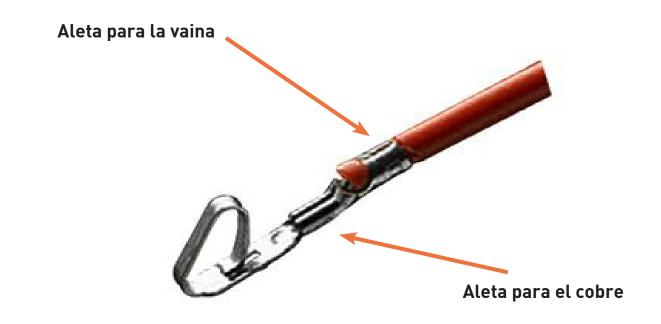




## PASO Nº 5.1.3 Armado de cables.

#### Crimpeado de cables

El cable debe quedar apretado en el terminal firmemente. Y el cobre del metal en contacto con el terminal. Intentar que el cobre del cable no sobresalga por la parte trasera del terminal y que los hilos de cobre entren en el tubo del terminal.











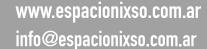
## **PASO Nº 5.2** Armado de conectores

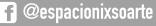


#### Conexión de Cables

El próximo paso es colocar los cables con los terminales en el conector. Cada cable debe ir en una posición determinada.

ATENCIÓN: Cuidado!!!, si no se conecta adecuadamente es probable que se rompa el sensor o el Arduino.





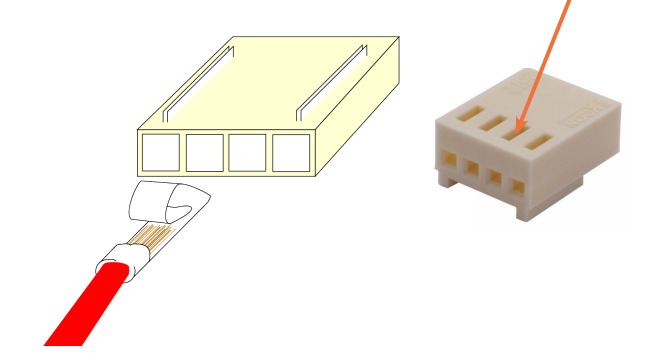




## PASO N° 5.2.1 Armado de conectores

#### Conector Molex 4 vías

Colocar un cable en cada vía del conector. La parte plana del terminal va hacia debajo. La parte plana del terminal se encuentra con la perforación del conector. El terminal entra bien profundamente hasta que una pequeña lengua del terminal se encastre en la perforación del conector.









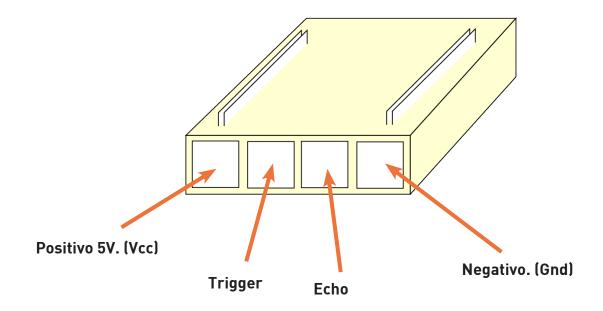




#### Conector Molex 4 vías

El orden de los cables para el sensor HC-sr04 es el que se muestra en la imagen. Lo que importa es recordar el color que se utilizo para luego hacer la conexión correcta del otro extremo del cable.

## PASO N° 5.2.2 Armado de conectores









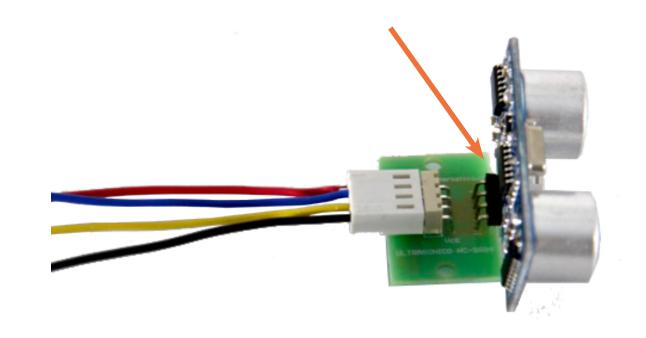


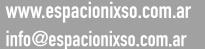


Conectar el cable al sensor ultrasónico.

Cuidado!!. Poner el conector de forma que concuerden los cables designados con la inscripción que posee el sensor en cada pin.

## PASO Nº 5.2.3 Armado de conectores









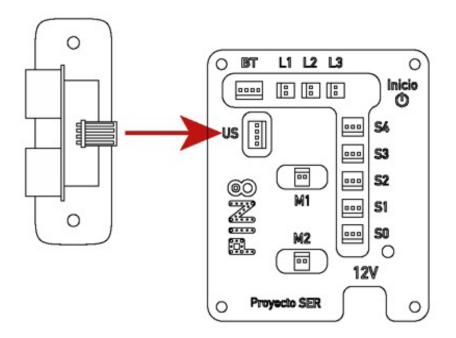


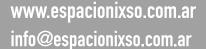


## PASO Nº 5.3 Conexión a la placa

#### Conectar el conector de 4 vías a la placa Pin8

Los conectores molex tienen una única posición correcta, donde encaja perfectamente la ficha hembra del cable con el conector macho de la placa y del sensor.



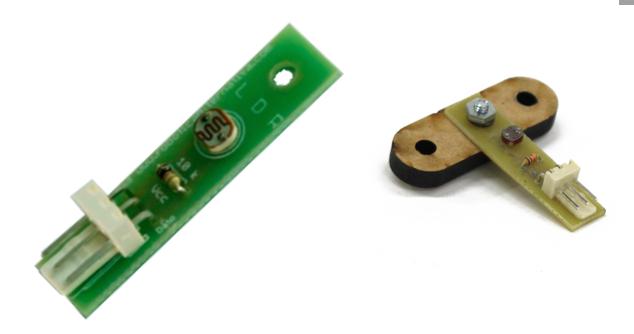






# PASO Nº 6 Conexiones del Sensor LDR

(necesario solamente para la utilización de este sensor)



#### Conexiones del Sensor LDR.

Este sensor permite medir cantidad de luz, es una resistencia que varia con la llegada de fotones. A mayor luz será mayor la medida, que esta en un rango de 0 a 1023. Cuidado: la conexión puede variar dependiendo el sensor, ya que existen distintos modelos, pero la lógica de conexión es la misma.





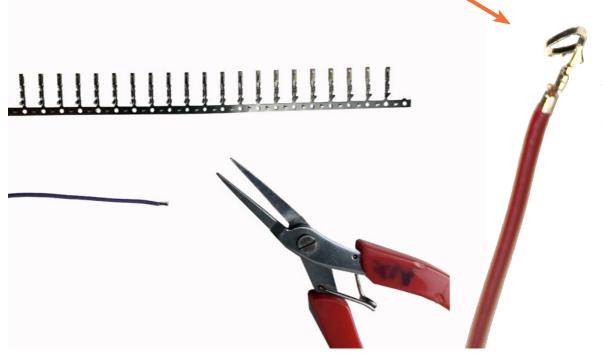




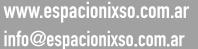
## PASO Nº 6.1.1 Armado de cables.

#### Crimpeado de cables

En este paso, usaremos unas pinzas de punta plana para crimpear los cables al **Terminal metálico Molex tipo HUXX**. Debemos colocar el terminal en ambos extremos de los 3 cables.



La medida de los cables debe ser de unos 25 cm aproximadamente y en preferencia de distintos colores para identificar facilmente las conexiones.





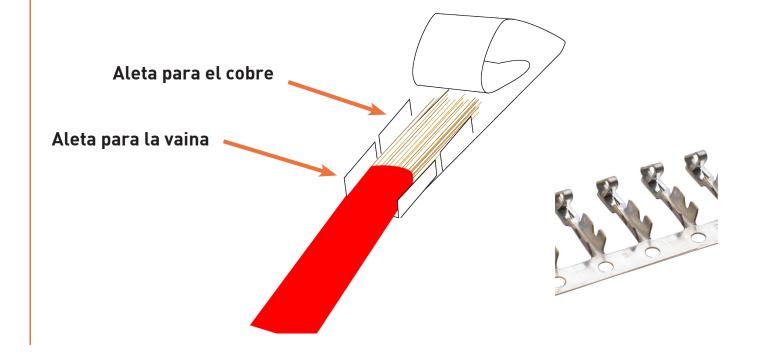




#### Crimpeado de cables

La vaina del cable debe estar sujetada por unas de las aletas. Y el cobre del cable debe estar apretado por las aletas internas haciendo contacto con el terminal.

## PASO Nº 6.1.2 Armado de cables.







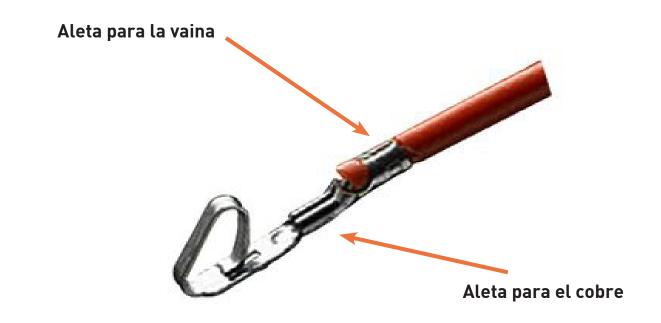


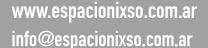


# PASO Nº 6.1.3 Armado de cables.

#### Crimpeado de cables

El cable debe quedar apretado en el terminal firmemente. Y el cobre del metal en contacto con el terminal. Intentar que el cobre del cable no sobresalga por la parte trasera del terminal y que los hilos de cobre entren en el tubo del terminal.



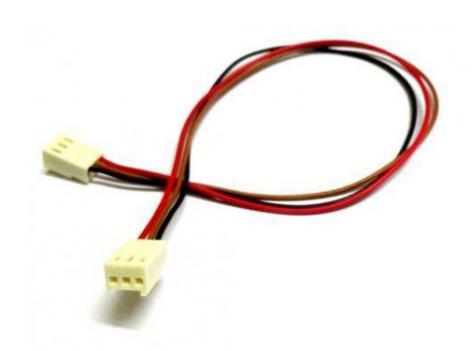








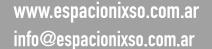
# PASO Nº 6.2 Armado de conectores

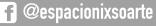


#### Conexión de Cables

El próximo paso es colocar los cables con los terminales en el conector. Cada cable debe ir en una posición determinada.

ATENCIÓN: Cuidado!!!, si no se conecta adecuadamente es probable que se rompa el sensor o el Arduino.





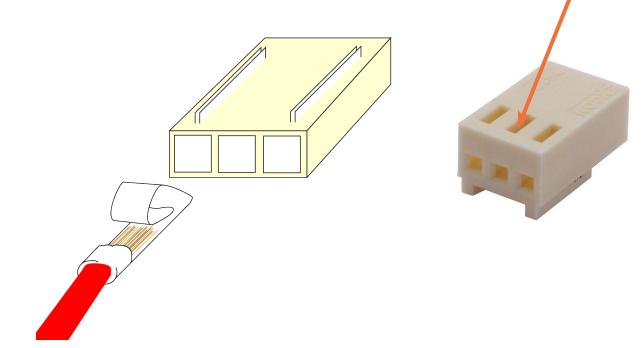




# PASO Nº 6.2.1 Armado de conectores

#### Conector Molex 3 vías

Colocar un cable en cada vía del conector. La parte plana del terminal va hacia debajo. La parte plana del terminal se encuentra con la perforación del conector. El terminal entra bien profundamente hasta que una pequeña lengua del terminal se encastre en la perforación del conector.









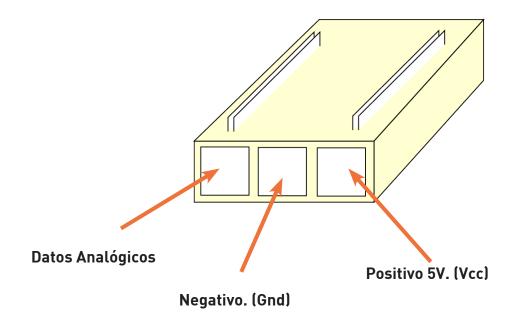




Tira de pines simples.

El orden de los cables para el sensor LDR de tres vías es el que se muestra en la imágen. Lo que importa es recordar el color que se utilizó para luego hacer la conexión correcta del otro extremo del cable.

# PASO Nº 6.2.2 Armado de conectores











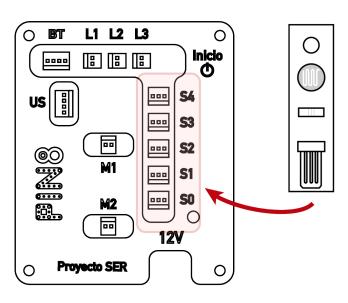


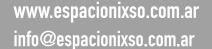
# PASO Nº 6.3 Conexión a la placa

#### Conectar el conector de 3 vías a la placa Pin8

Los conectores molex tienen una única posición correcta, donde encaja perfectamente la ficha hembra del cable con el conector macho de la placa y del sensor.











# PASO Nº 7 Conexiones del Sensor Infrarrojo

(necesario solamente para la utilización de este sensor)



#### Conexiones del Sensor Infrarrojo.

la misma.

Este sensor posee un emisor de luz infrarroja y un receptor de luz infrarroja.

Es módulo que permite medir cuanta luz infrarroja rebota, de su propio emisor, sobre una super-

ficie. A mayor luz de rebote, mayor será la medición, que esta en un rango de 0 a 1023. Por lo general, si la superficie es clara o brillante, rebota mayor cantidad de luz que en una superficie oscura o rugosa. ATENCIÓN: Cuidado! la conexión puede variar dependiendo el modelo de sensor, ya que existen varios, pero la lógica de conexión es











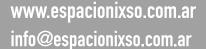
# PASO Nº 7.1.1 Armado de cables.

#### Crimpeado de cables

En este paso, usaremos unas pinzas de punta plana para crimpear los cables al **Terminal metálico Molex tipo HUXX.** Debemos colocar el terminal en ambos extremos de los 3 cables.



La medida de los cables debe ser de unos 25 cm aproximadamente y en preferencia de distintos colores para identificar facilmente las conexiones.





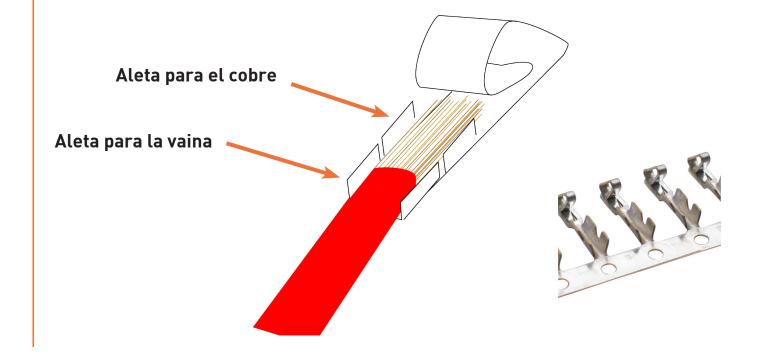


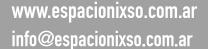


# PASO Nº 7.1.2 Armado de cables.

#### Crimpeado de cables

La vaina del cable debe estar sujetada por unas de las aletas. Y el cobre del cable debe estar apretado por las aletas internas haciendo contacto con el terminal.







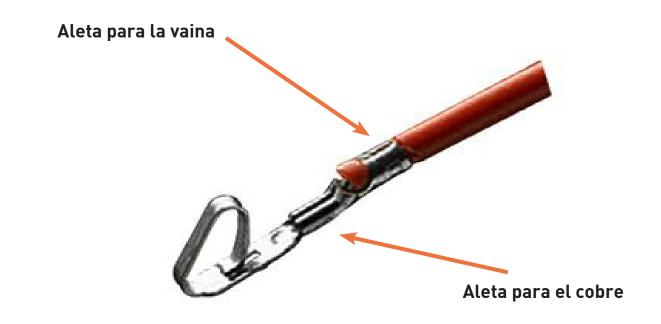




# PASO Nº 7.1.3 Armado de cables.

#### Crimpeado de cables

El cable debe quedar apretado en el terminal firmemente. Y el cobre del metal en contacto con el terminal. Intentar que el cobre del cable no sobresalga por la parte trasera del terminal y que los hilos de cobre entren en el tubo del terminal.



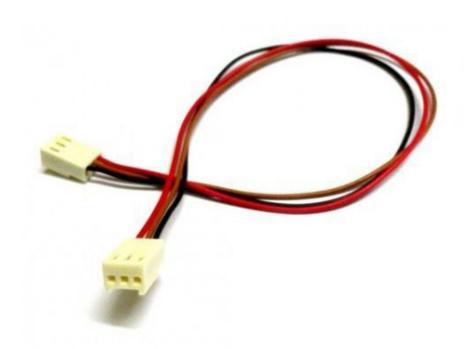








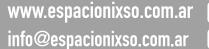
# **PASO Nº 7.2** Armado de conectores

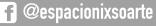


#### Conexión de Cables

El próximo paso es colocar los cables con los terminales en el conector. Cada cable debe ir en una posición determinada.

ATENCIÓN: Cuidado!!!, si no se conecta adecuadamente es probable que se rompa el sensor o el Arduino.





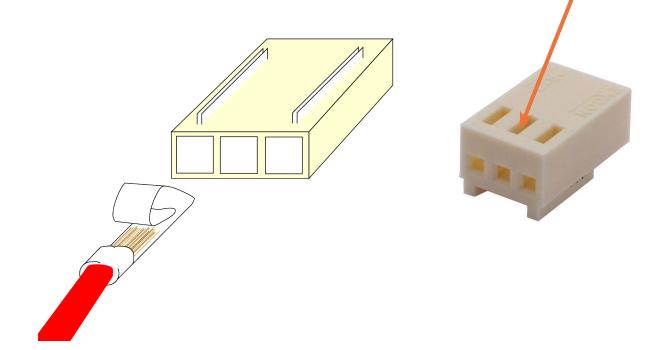




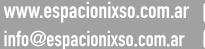
# PASO Nº 7.2.1 Armado de conectores

#### Conector Molex 3 vías

Colocar un cable en cada vía del conector. La parte plana del terminal va hacia debajo. La parte plana del terminal se encuentra con la perforación del conector. El terminal entra bien profundamente hasta que una pequeña lengua del terminal se encastre en la perforación del conector.









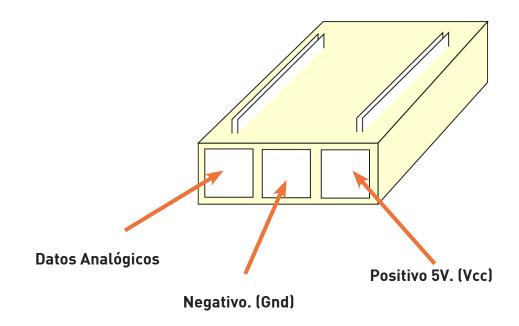




# PASO Nº 7.2.2 Armado de conectores

#### Tira de pines simples.

El orden de los cables para el sensor LDR de tres vías es el que se muestra en la imágen. Lo que importa es recordar el color que se utilizó para luego hacer la conexión correcta del otro extremo del cable.











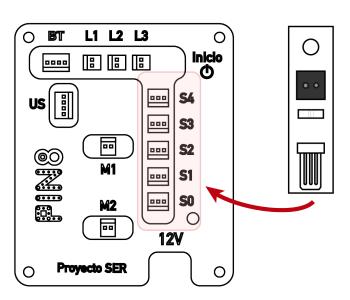


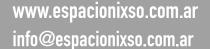
# PASO Nº 7.3 Conexión a la placa

#### Conectar el conector de 3 vías a la placa Pin8

Los conectores molex tienen una única posición correcta, donde encaja perfectamente la ficha hembra del cable con el conector macho de la placa y del sensor.











# PASO Nº 8 CONEXIÓN DE BATERÍAS





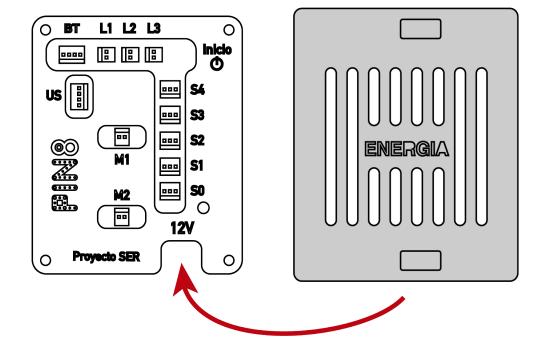
Conexiones de las Batería.

Este paso es muy importante, no hay que confundir la conexión porque hay peligro de ruptura. La conexión se realiza en las borneras correctas sin confundir el positivo y el negativo. El negativo está marcado con la sigla GND: Los voltajes que soporta la placa es de 5 a 19 volt. Nuestra propuesta es trabajar con 8 baterías de 1.5. Un total de 12 volt máximo.

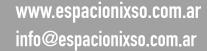




PASO Nº 8.1 Conexiones de la batería Conexiones de las Baterías.









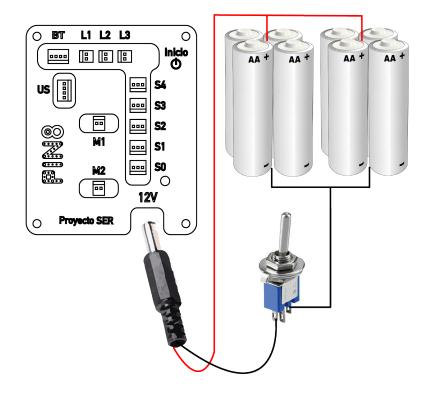




# PASO Nº 8.2 Conexiones de la batería

#### Conexiones de las Baterías.

Se recomienda colocar un interruptor en el conexionado de la energía entre la placa y las baterías. Asi se puede prender y apagar el robot desde allí.









# ¿Cómo programar Pin8?











El robot Pin8 posee una placa Arduino Uno como controlador de sus sistemas. El programa que realizaremos para controlar el robot debe alojarse en la placa, a esa acción la llamaremos cargar el programa.

El programa debe escribirse en lenguaje **Arduino** desde su propio **Ide** ( entorno de desarrollo integrado ). El **Ide** es un procesador de texto dedicado al lenguaje, que con algunas herramientas nos ayuda a programar y cargar el programa al robot.

El lenguaje Arduino puede ser complejo para los recién iniciados en la programación, por ello Pin8 posee su propio Ide que permite acercarse a la programación de forma visual y más intuitiva. El Ide de Pin8 generará un código Arduino que debe abrirse con el Ide de Arduino para ser cargado en el robot. Este paso es importante, el objetivo es que el programador de a poco se vaya familiarizando con el lenguaje Arduino, que en el fondo es realmente la forma de programar eficientemente.

Resumiendo, para poder programar Pin8 hay que descargar de internet el Ide de Arduino y el





#### Ide de Pin8.

El programa lo realizaremos en el Ide de Pin8, lo guardaremos en el disco rígido, luego lo abriremos con el Ide de Arduino para poder cargárselo al robot. Este procedimiento parece a primera vista engorroso pero al utilizarlo no es nada incómodo.

La guía de instalación del Ide de Arduino se encuentra en un anexo al final de este tutorial.

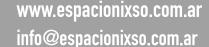
# Ide de ComPacto Descarga e instalación

El Ide ( entorno de desarrollo integrado ) de Pin8 se encuentre las descargas de:

http://www.roboticaalternativa.com/comPacto.html

Se descargara el archivo pin8.zip que hay que descoprimir. Dentro de los archivos descomprimidos, se encontrara un carpeta con el nombre "softpin8".

Dentro de esta se encontrara el archivo "pin8.html". Este archivo es el lde de "pin8".









#### Ide de Pin8

El Ide (entorno de desarrollo integrado) de "Pin8" es un archivo HTML, en otras palabras, es una pagina web. Puede utilizarse con cualquier navegador web, pero se recomienda utilizar <u>Google Chrome</u>.



El Ide consta de un área de trabajo blanca punteada, un menú horizontal superior y un menú vertical a la izquierda.

En este menú se encuentran todos los bloques que podemos utilizar agrupados por su función. El menú horizontal superior contiene las opciones de traducción del programa y guardado





#### Menú horizontal

El menú horizontal contiene la opciones de traducción de lenguajes. El programa se realizará en la opción "Bloques" y al tocar cualquiera de las otras pestaña, este programa de bloques se traducirá a la opción seleccionada.









#### Menú horizontal Arduino

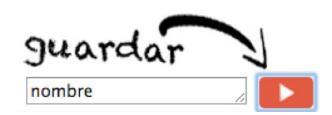
Al seleccionar esta pestaña, el programa realizado con bloques pre-diseñados, se traducirá instantaneamente a **código Arduino**. Si se modifica un bloque al volver a esta pestaña se cambiara el código automáticamente.

Para trasladar este código al Ide de Arduino y finalmente cargárselo al robot, se puede tomar dos caminos.

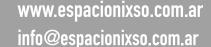
**#1** El primero es seleccionar el texto de código con el Mouse y copiarlo con las teclas "control+c" y pegarlo en el **Ide de Arduino** con las teclas "control+v".

**#2** El otro camino es descargar el código presionando el botón **"guardar"** luego de colocarle el nombre.

Se guardara un archivo .ino ( .arduino) con el nombre seleccionado más los datos del día y horario actual. Esto es para que no se superpongan versiones del mismo proyecto. Luego se podrá abrir con el Ide Arduino.















# Menú horizontal Arduino con Librería

Al seleccionar esta pestaña, el programa realizado con bloques ya pre-diseñados, se traducirá instantaneamente a **código Arduino con librería**. Es decir que los comandos serán funciones que ocultan parte del lenguaje **Arduino** pero es más sencillo de leer. Si se modifica un bloque al volver a esta pestaña se cambiará el código automáticamente.

Para la utilización del código con librería, previamente hay que instalar la librería.

La librería se encuentra entre los documentos descargados, se trata de la carpeta que lleva el nombre de LibreriaPin8. Para instalar la librería solamente hay que copiar esta carpeta a misDocumentos/Arduino/libraries de tu computadora (previamente hay que instalar el Ide de Arduino y ejecutarlo al menos una vez).

Una vez copiado, hay que cerrar el lde de Arduino y volver a abrirlo (al iniciar Arduino nuevamente se reconocerá sus librerías nuevas).







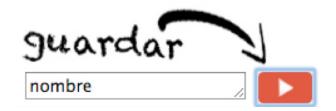
# Menú horizontal Arduino con Librería

Para trasladar este código al Ide de Arduino y finalmente cargárselo al robot, se puede tomar dos caminos.

#1 El primero es seleccionar el texto de código con el Mouse y copiarlo con las teclas "control+c" y pegarlo en el Ide de Arduino con las teclas "control+v".

#2 El otro camino es descargar el código presionando el botón "guardar" luego de colocarle el nombre.

Se guardara un archivo .ino ( .arduino) con el nombre seleccionado más los datos del día y horario actual. Esto es para que no se superpongan versiones del mismo proyecto. Luego se podrá abrir con el Ide Arduino.













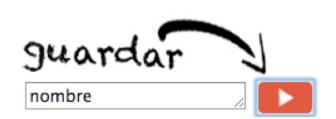


# Menú horizontal XML

Al seleccionar esta pestaña, el programa realizado con bloques pre-diseñados, se traducirá instantaneamente a código XML. Si se modifica un bloque al volver a esta pestaña se cambiará el código automaticamente.

Esta pestaña permite guardar el proyecto en forma de bloques, para poder recuperar el programa y seguir trabajando sobre él. Para guardar hay que presionar el botón "guardar" luego de colocarle el nombre.

Se guardará un archivo .xml con el nombre seleccionado más los datos del día y horario actual. Esto es para que no se superpongan versiones del mismo proyecto. Luego se podrá abrir con el Ide Arduino.









# Menú horizontal XML

Para recuperar un programa ya guardado en xml hay que presionar el botón "cargar xml".

Luego en la ventana emergente, seleccionar el programa deseado. Este proceso recuperará el programa de bloques guardado.





Bloques	Arduino	Libreria	XML	Manú horizontal Bloques				
Pin 8 Movimientos Motor Serial	Menú horizontal Bloques  En este menú se presentan todos los bloques que posee "Pin8" para poder programar el robot.							
Lógica Secuencias Matemáticas	Todos los bloque se encuentran agrupados en sub menús según su función en el menú vertical izquierdo.  Para programar solamente hay que buscar el bloque deseado en los menús.							
Variables	Una vez encontrado hay que arrastrarlo con el mouse sobre el lienzo blanco punteado de trabajo.							
			•	ienzo de trabajo, hay que rastrarlo al tacho de <mark>basura</mark>				











Bloques	Arduino	Libreria	XML	Menú horizontal Bloques				
Pin 8 Movimientos Motor Serial	En menú "Pin8" se encuentran los bloques con funciones del robot que sean movimientos. Por ejemplo las funciones de sensado, de espera y de no repeción de ciclos.  En el menú "Movimientos" están las funciones de movimientos del robot diferenciar los motores. Por ejemplo el avanzar o retroceder los dos motores simitáneamente.							
Lógica Secuencias Matemáticas								
Variables	El menú "Motor" podemos encender y apagar los motores en forma ir diente entre ellos.							











Bloq	ues
------	-----

Arduino

Libreria XI

**XML** 

# Menú horizontal Bloques

Pin 8
Movimientos
Motor

Serial

Lógica Secuencias Matemáticas

**Variables** 

El menú "Serial" es más avanzado, tiene las función de comunicación entre el robot y la computadora en tiempos de ejecución. Por ejemplo, puedo comunicar al robot con la computadora para ver los valores de los sensores en tiempo real durante la ejecución de un programa.

El menú de "Lógica" contiene las herramientas necesarias para la toma de decisiones de un programa. Por ejemplo si un robot descubre a través de sus sensores que se encuentra un obtáculo al frente, este robot puede decidir retroceder y girar para evitarlo.

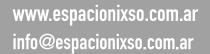
En "Secuencias" están los bloques de loops. Estos sirven para realizar tareas repetitivas. Por ejemplo, si deseo que mi robot repita 10 veces el mismo movimiento, entre estos bloques encontraré las herramientas para lograrlo, sin tener que colocar 10 veces los mismos bloques.







Bloques	Arduino	Libreria	XML	Menú horizontal Bloques				
Pin 8 Movimientos Motor								
Serial  Lógica Secuencias Matemáticas	Dentro del menú "Matemáticas" encontraré una serie de bloques que me permiten realizar cálculos matemáticos. Algunos simples y otro más complejos. Sumas, restas y hasta generación de números aleatorios entre otros.							
Variables	de variable el program	es. Son espa	cios de m valores. P	'ariables" encontrarán la posibilidad de la creación emorias que se van a utilizar cuando se quiere que or ejemplo, registrar los valores de un sensor para del mismo.				





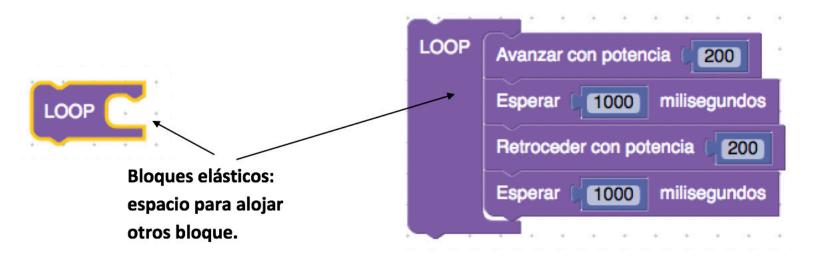




### **Tipos de Bloques**

Existen distintos tipos de bloques. Algunos bloques contienen un espacio para alojar otros bloques para su funcionamiento, los que se denominan parámetros.

Y hay otros bloques que los llamaremos elásticos, a parte de los parámetros de funcionamiento, pueden estirarse para alojar más bloques.



Los bloques son intuitivos y a simple vista puede verse en su morfología estas diferencias.

Entre bloques, se encastran con unas anclas imantadas en forma lineal. Uno debajo del otro.



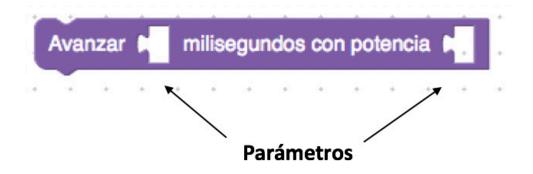


### Ejemplo de anclaje entre bloques.





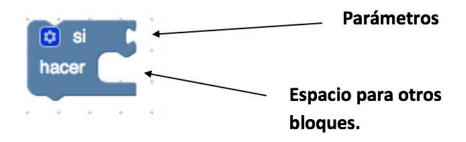
### Ejemplo de Parámetros de los bloques.

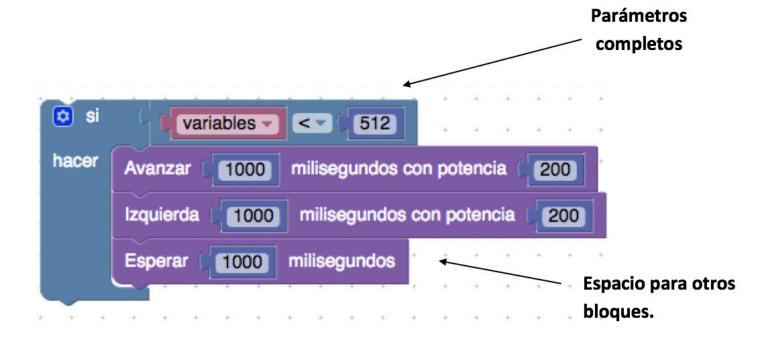






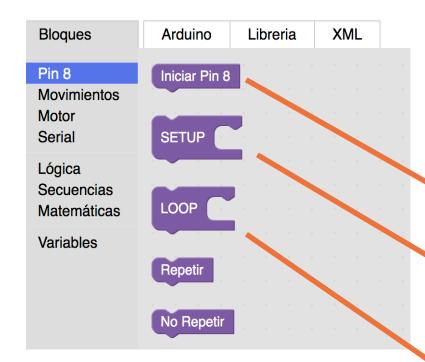
### Ejemplo de parámetros de los bloques y bloques elásticos.











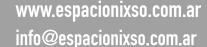
# **Bloques Fundamentales**

Existen tres bloques fundamentales que deben estar presentes en todos los programas. Los tres se encuentran en el menú de "Bloques" / "Pin8":

El primero es el bloque "iniciar Pin8". Es un bloque sin parámetros.

El segundo es el bloque "Setup" que es un bloque elástico. Dentro de él colocaremos todos bloques que se ejecutarán solo una vez y al inicio del programa.

El tercero es el bloque "Loop" que es un bloque elástico. Dentro del él colocaremos las acciones que realizará el robot. Una vez que culmine con la ultima acción, volverá al inicio para repetir indefinidamente todos los bloques.







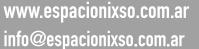


#### **Bloques Fundamentales**

Los tres bloques fundamentales deben estar siempre en todos los programas. Unidos entre ellos uno debajo del otro y en orden.

El siguiente ejemplo es un programa funcional, correcto pero vacío.





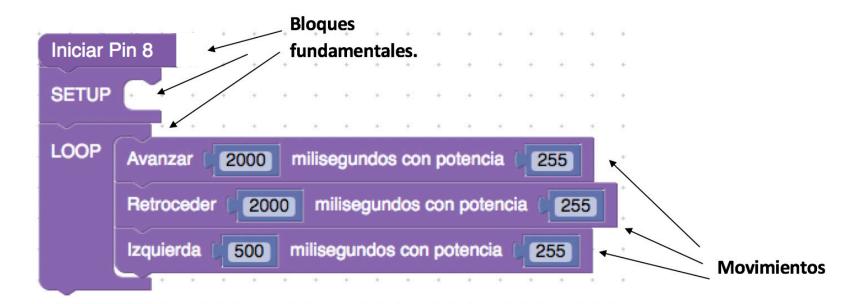




#### **Bloques Fundamentales**

Realizaremos un programa sencillo como ejemplo.

El objetivo es que nuestro robot avance unos segundos, luego retroceda al mismo lugar y por último realice un giro a la izquierda.

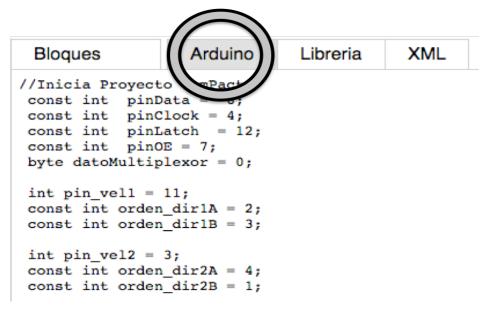




# ¿Cómo pasar el programa de bloques a

**Arduino?** 

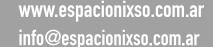
#### Conversión de bloques a Arduino



Una vez colocados los bloques se debe seleccionar del menú horizontal superior la opción "Arduino". En esta pestaña, encontraremos el mismo programa que realizamos con bloques pero en lenguaje por código.

Este es el código que debo transportar al Ide de "Arduino".











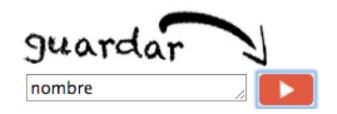
#### Conversión de bloques a Arduino.

Como se ha explicado anteriormente existen dos formas de transportar el código al Ide de "Arduino".

1ra opción. Descargar el código en un archivo: pulsar el botón de guardar y se descargará un archivo que debo abrir con el Ide de "Arduino". Podemos agregar un nombre al archivo utilizando solamente letras sin espacios. El nombre se completará con el día, la hora minuto y segundo del instante en que se descarga el archivo.

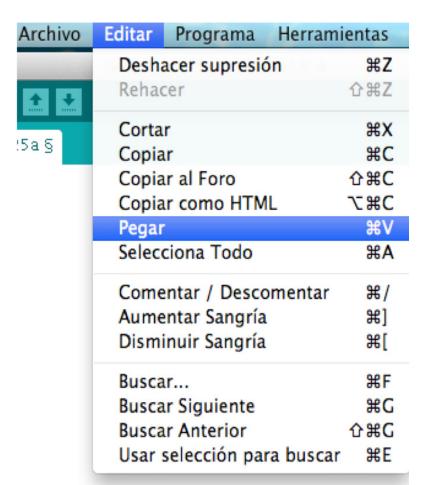
El archivo se descarga en "Descargas" de la computadora. Al hacer doble click sobre el archivo se abrirá automáticamente con el programa Ide de "Arduino". También se puede abrir seleccionando el archivo descargado desde la barra de menú del Ide de "Arduino": Archivo / abrir.

Al abrir el Ide me mostrará un cartel pidiendo que se debe crear una carpeta con nombre: aceptamos esa necesidad.









#### Conversión de bloques a Arduino.

2da opción. Copiar el código completo generado en el Ide de "Pin8" y pegar en el Ide de "Arduino" (asegúrense de copiar todo el código).

Para copiar el código: posicione el mouse en la parte superior del código presione el botón izquierdo y arrastre el mouse hasta el final sin soltar el botón. Una vez que pinten todo el código suelten el botón del mouse. Posteriormente sin despintar el código, presionen sobre el botón derecho del mouse y seleccionen del menú emergente la opción "copiar".

Una vez copiado, diríjanse al Ide de "Arduino". Tomen un nueva hoja de trabajo y borren todo el texto que contenga. Seleccionen del menú desplegable la opción pegar.



# ¿Cómo cargar un

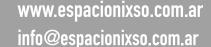
programa al robot?

PASO Nº 1.1 Código en el Ide de Arduino

#### Cargando programa al Robot.

```
//Inicia Proyecto ComPacto
const int pinData = 8;
const int pinClock = 4;
const int pinLatch = 12;
const int pin0E = 7;
byte datoMultiplexor = 0;
int pin_vel1 = 11;
const int orden_dir1A = 2;
const int orden_dir1B = 3;
int pin_vel2 = 3;
const int orden_dir2A = 4;
const int orden_dir2B = 1;
int pin_led0 = 6;
const int orden_led0 = 5;
int pin_led1 = 5;
Suardado.
```

El primer paso es tener con seguridad el programa deseado en cuerpo del Ide de "Arduino".





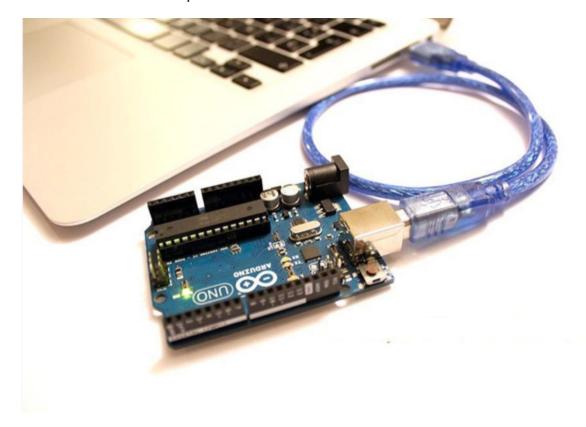




# **PASO Nº 1.2 Conexión USB**

#### Cargando programa al Robot.

El segundo paso es conectar el cable USB desde el controlador "Arduino" a la computadora.









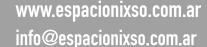


# **PASO Nº 1.3** Selección de Placas









tro de la opción "Placas" buscamos la nuestra.





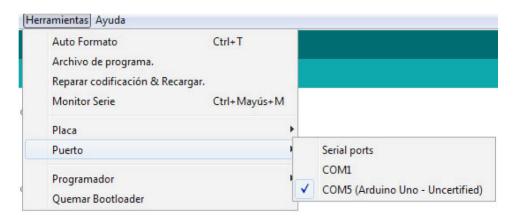


### PASO Nº 1.4 Selección de Puerto

#### Cargando programa al Robot.

El próximo paso es seleccionar en que puerto de la computadora hemos conectado la placa "Arduino". Esto se debe a que podemos tener conectados distintos periféricos o distintos "Arduino" simultáneamente, por ello debemos indicar cuál vamos a utilizar. También es probable que este paso lo tengamos que realizar solamente una vez o cada tanto.

Vamos a la barra de menú horizontal, seleccionamos la opción "Herramientas" y luego dentro de la opción "Puertos" buscamos allí nuestra placa conectada. Si no aparece el nombre, tenemos que probar una por una hasta dar con la correcta. Los números de puerto lo asigna el sistema operativo de cada computadora.









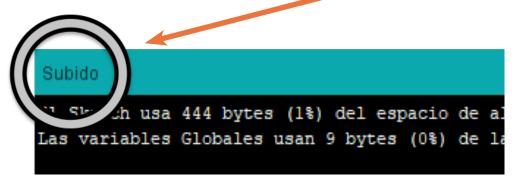
### PASO N° 1.5 Puerto

#### Cargando programa al Robot.

El último paso es cargar el programa. Solamente seleccionamos el botón "Subir".



El proceso tardará unos segundos. Y finalmente en la barra superior de la terminal nos dará un mensaje de programa "Subido". El robot ejecutará el programa en unos instantes.





# ¿Cómo programar Sensores?









#### ¿Qué es un sensor ultrasónico?

Un sensor ultrasónico es un detector de proximidad. Permiten calcular que distancia existe entre el sensor y el objeto más próximo frente a este.

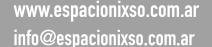


El sensor consta de un emisor sonoro y un receptor. El sensor emite un sonido y mide el tiempo que tarda el sonido en golpear con un objeto y regresar. Ese tiempo permite calcular la distancia entre él y el objeto donde se reflejó el sonido. Estos objetos tiene un distancia máxima de uso, en el caso de nuestro programa posee una máximo valor de 100 cm.

Hay que tener presente que el sensor solo trabaja en forma lineal con un ángulo pequeño de trabajo, en otras palabras, el sensor ve solamente hacia delante y no hacia los costados. También solo detecta objetos grandes, como una pared, una caja, etc. Es menos probable que detecte objetos pequeños como la pata de una silla o de una mesa. Otro posible problema es si el sensor esta inclinado hacia el piso, es probable que entienda que el piso sea un objeto delante de él.

# Sensor Ultrasónico











### Sensor Ultrasónico

#### Bloque Ultrasónico

La conexión del sensor ya se ha explicado en este mismo tutorial.

El bloque que utilizaremos para el uso del sensor se llama "Establecer Elemento con el valor de sensor Ultrasónico". Este bloque se encuentra en el menú "Pin8". El bloque sensor tiene un solo parámetro que es una variable. Por defecto se llama "elemento" pero puede cambiarse el nombre.

Establecer elemento con el valor del sensor Ultrasonico

Una variable es en un espacio de la memoria donde se guardará el valor que el bloque devuelve:

El resultado de la medición de la distancia de 1 a 100 cm.

A ese espacio de memoria se le llama variable y puede etiquetarse:

Podemos llamarla con un nombre que nosotros inventemos. Por lo general, se le pone un nombre que describa el contenido del espacio. Puede pensarse como una caja donde guardo cosas y le pongo una etiqueta por fuera describiendo el contenido.





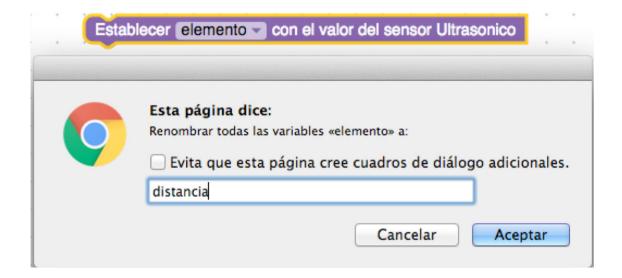




#### Sensor Ultrasónico

#### Nombre de una variable

El bloque posee el nombre "elemento", pero puede crearse uno nuevo, por ejemplo: "distancia". Uno de los caminos para generar una nueva variable es haciendo "click" en el menú desplegable al lado del nombre "elemento" dentro del bloque. En este mismo, seleccionaremos la opción "Renombrar". Por último asignaremos el nuevo nombre de la variable. El nombre puede estar formado por letras y números, pero no puede empezar su nombre con un número.









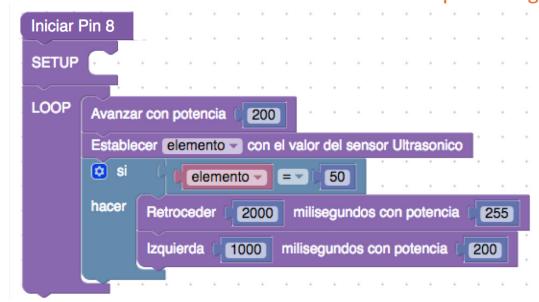




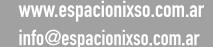
# Sensor Ultrasónico

En resumen, el bloque "Establecer elemento con el valor de sensor Ultrasónico" recoge el valor que entrega el sensor ultrasónico y lo guarda en una variable. Por default lo guarda en una variable llamada "elemento" pero el bloque permite generar una nueva variable con el nombre que deseamos.

Lo importante es que con el valor de la medición realizada por el sensor, podemos tomar decisiones de comportamiento luego de realizar un análisis. Esa toma de decisiones se realiza con los bloques de lógica.









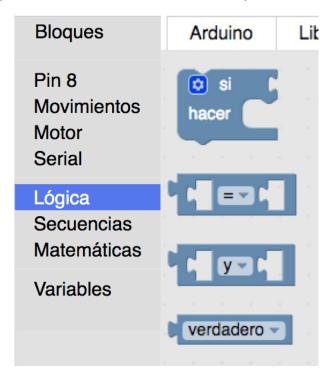




#### Sensor Ultrasónico

#### Bloque de lógica

En el menú vertical de **bloques "lógica"**, encontraremos los bloques necesarios para que el robot pueda tomar decisiones por sí mismo. Hasta ahora todos los ejemplos que hemos realizado han establecido los movimientos de nuestro robot linealmente: primero uno, luego otro y otro para finalmente volver a empezar.



Ahora podemos generar distintas opciones, por ejemplo utilizando el valor del sensor ultrasónico como disparador de distintos movimientos.

Por ejemplo: podemos hacer un programa para que el robot avance pero si y solo si el robot detecta un objeto delante de él, retroceda y gire a la izquierda. Con esta lógica no podríamos decir cuanto tiempo el robot avanzará, pero si sabemos que retrocederá y girará cuando detecte algún obstáculo.







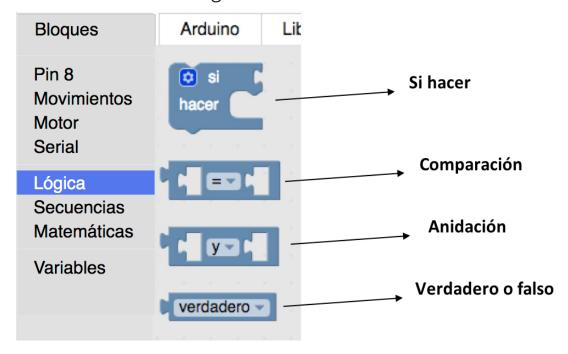


#### Sensor Ultrasónico

#### Bloque de lógica

En el menú vertical de **bloques "lógica"**, encontraremos los bloques necesarios para que el robot pueda tomar decisiones por si mismo.

La utilización de cada bloque puede verse en las referencias de la web: <a href="http://www.roboticaalternativa.com/referenciasPin8.html">http://www.roboticaalternativa.com/referenciasPin8.html</a>
o en el material de descarga.





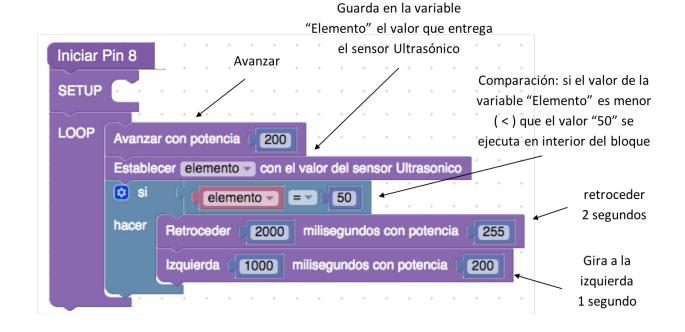




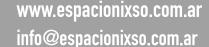
# Sensor Ultrasónico

#### Ejemplo Sensor Ultrasónico

El robot avanza siempre, pero si y solo si el robot detecta un objeto delante de él: retrocede y gira a la izquierda.











#### Sensores analógicos

Todos los sensores analógicos poseen la misma lógica de funcionamiento. Técnicamente, por lo general y a grandes rasgos, entregan un valor variables de 0 a 5 volt.

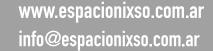
"Pin8" admite aquellos sensores que respondan a esta características. Este proyecto propone la utilización de sensores LDR y infrarrojos. El rango de medición de un sensor analógico en "Arduino" se encuentra entre un valor de 0 a 1023.





Sensores Analógicos LDR Infrarrojo otros











#### ¿Qué es un Sensor LDR?

Este sensor permite medir cantidad de luz, es una resistencia que varia con la llegada de fotones. A mayor luz será mayor la medida, que esta en un rango de 0 a 1023. Cuidado: la conexión puede variar dependiendo el sensor, ya que existen distintos modelos, pero la lógica de conexión es la misma.

# **Sensores Analógicos**

#### ¿Qué es un Sensor Infrarrojo?

Este sensor posee un emisor de luz infrarroja y un receptor de luz infrarroja. Es modulo permite medir cuanta luz infrarroja revota, de su propio emisor, sobre una superficie. A mayor luz de rebote, mayor será la medida, que esta en un rango de 0 a 1023. Por lo general, si la superficie es clara o bríllate rebota mayor cantidad de luz que una superficie oscura o rugosa. Cuidado: la conexión puede variar dependiendo el sensor, ya que existen distintos modelos, pero la lógica de conexión es la misma.









## Sensores Analógicos

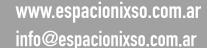
#### Bloque analógico

Los dos tipos de sensores propuestos utilizan los mismos bloques y lógica. Así que solo explicaremos como ejemplo de ambos uno: usaremos como ejemplo los LDR. El bloque que utilizaremos para el uso del sensor se llama "Establecer Elemento con el valor de sensor". Este bloque se encuentra en el menú "Pin8". El bloque sensor tiene dos parámetros: una variable que se llama "elemento" pero puede cambiarse el nombre. El otro parámetro indica en que pines hemos conectado el sensor en la placa shield de "Arduino".

Establecer elemento con el valor del sensor 0

Una variable es un espacio de la memoria donde se guardará el valor que el bloque devuelve. A ese espacio de memoria se le llama variable y puede etiquetarse:

Podemos llamarla con un nombre que nosotros inventemos. Por lo general, se le pone un nombre que describa el contenido del espacio. Puede pensarse como una caja donde guardo cosas y le pongo una etiqueta por fuera describiendo el contenido.









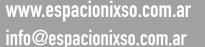
Sensores Analógicos

En resumen, el bloque "Establecer elemento con el valor de sensor" recoge el valor que entrega el sensor analógico y lo guarda en una variable. Por default lo guarda en una variable llamada "elemento" pero el bloque permite generar una nueva variable con el nombre que deseamos.

```
Iniciar Pin 8
SETUP
LOOP
        Establecer sensor1 con el valor del sensor
         Establecer sensor2 con el valor del sensor
        🔯 si
                                        512
                       sensor1 -
                Avanzar motor 1 con potencia
                                              200
        sino
                Avanzar motor 1 con potencia
        🔯 si
                      sensor2 - > -
                                         512
        hacer
                                              200
                Avanzar motor 2 con potencia
        sino
                Avanzar motor 2 con potencia
```

Lo importante es que con el valor de la medición realizada por el sensor, podemos tomar decisiones de comportamiento luego de realizar un análisis. Esa toma de decisiones se realiza con los bloques de lógica.







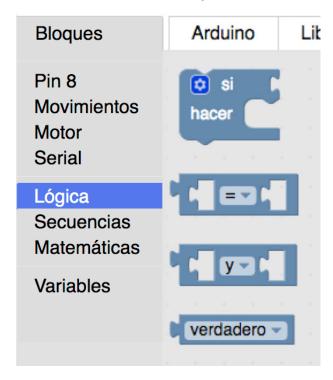




# Sensores Analógicos

#### Bloque de lógica

En el menú vertical de **bloques "lógica"**, encontraremos los bloques necesarios para que el robot pueda tomar decisiones por sí mismo. Hasta ahora todos los ejemplos que hemos realizamos han establecido los movimiento de nuestro robot linealmente: primero uno, luego otro y otro para finalmente volver a empezar.



Ahora podemos generar distintas opciones, por ejemplo utilizando el valor del sensor LDR como disparador de distintos movimientos.

Por ejemplo: podemos hacer un robot que escape de la luz. Lo primero es armar un Robot con dos ruedas y colocar un sensor LDR cerca de cada rueda. Si logramos hace un programa que cuando un sensor detecta luz encienda una rueda, el robot escapara de la luz.







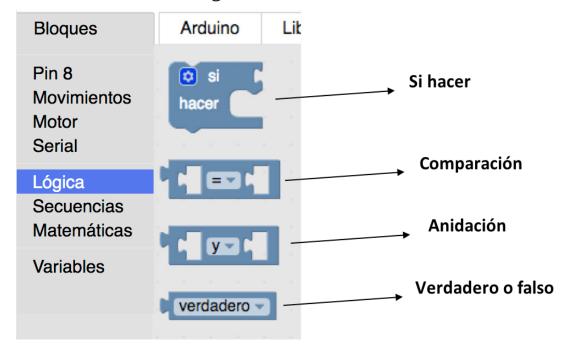


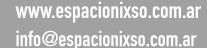
# **Sensores Analógicos**

#### Bloque de lógica

En el menú vertical de bloques "lógica", encontraremos los bloques necesarios para que el robot pueda tomar decisiones por si mismo.

La utilización de cada bloque puede verse en las referencias de la web: <a href="http://www.roboticaalternativa.com/referenciasPin8.html">http://www.roboticaalternativa.com/referenciasPin8.html</a>
o en el material de descarga.







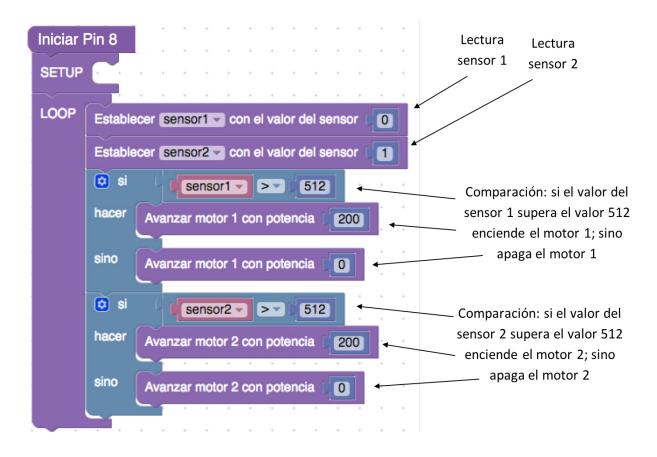


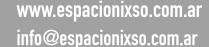


# Sensores Analógicos

#### **Ejemplo Sensor LDR**

Hacer un Robot que escape de la luz.









# **ANEXO: Instalación de ARDUINO**











El Ide de "Arduino" deberá descargarse de la página web de Arduino. En el menú superior, la pestaña "SOFTWARE", hay que seleccionar la opción "DESCARGAS".





Posteriormente, se abrirá una nueva página. Deslizar el cursor hacia abajo y seleccionar el Ide a descargar que corresponda al sistema operativo de tu computadora.



#### ARDUINO 1.8.11

El software Arduino de código abierto (IDE) hace que sea fácil escribir código y subirlo a la placa. Se ejecuta en Windows, Mac OS X y Linux. El entorno está escrito en Java y se basa en el procesamiento y otro software de código abierto.

Este software se puede usar con cualquier placa Arduino. Consulte la página de Inicio para obtener instrucciones de instalación.

Windows Installer, para Windows XP y archivos ZIP de

Windows para instalación no administrativa

La aplicación de Windows requiere Win 8.1 o 10 Get #

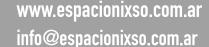
Mac OS X 10.8 Mountain Lion o más reciente

Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits









La nueva página que se abrirá, visualiza la opción para contribuir monetariamente a los desarrollos de la plataforma Arduino. La contribución es opcional y el software a descargar seguirá siendo el mismo, independientemente de la decisión que tomes.







Si descargas el software para Windows, tiene dos opciones. Un archivo .exe ejecutable o un archivo comprimido .zip. Si seleccionas el .exe ejecutable solo debes hacer "doble click" con el mouse para que se inicie el programa. Si seleccionas el .zip deberás descomprimir el archivo. Se generará un directorio dentro del cual se encontrará el software de Arduino.

Se recomienda utilizar el .exe, ya que al instalarse, también se instalarán los controladores (drivers). Esta es la guía oficial para Windows de Arduino: <a href="https://www.arduino.cc/en/Guide/Windows">https://www.arduino.cc/en/Guide/Windows</a>

Para descargar el software para MAC OSX, encontrarás un archivo comprimido en .zip. Descarga, descomprime y por último, arrastra el archivo del software a la carpeta de aplicaciones a modo de darle un orden. Esta es la guía oficial: <a href="https://www.arduino.cc/en/Guide/MacOSX">https://www.arduino.cc/en/Guide/MacOSX</a>





Una vez instalado el software, hacer doble click sobre el programa. Si este fue instalado correctamente, se abrirá un ventana similar a la siguiente imagen:

```
sketch
void setup() {
  // put your setup code here, to run once:
void loop() {
 // put your main code here, to run repeatedly:
```

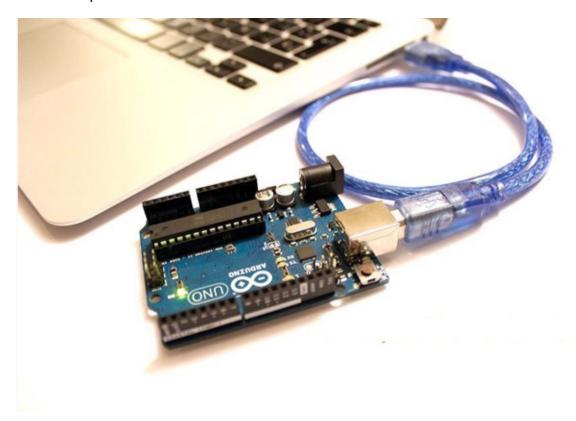




Test de Carga PASO Nº 1 Conexión USB

#### Test de Carga de un programa al "Arduino".

El primer paso es conectar el cable USB desde el controlador "Arduino" a la computadora.



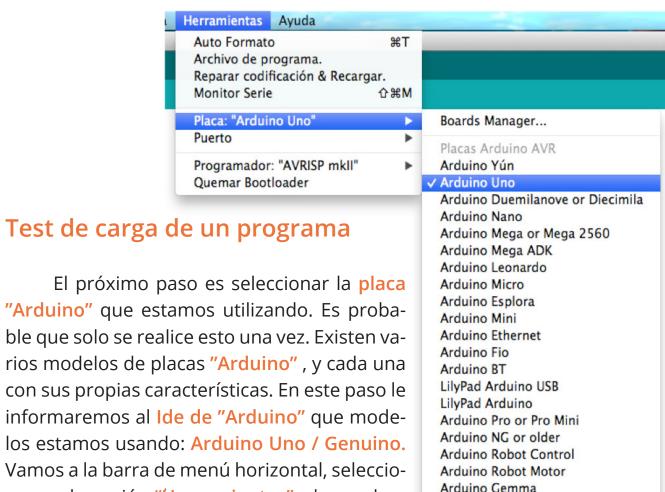


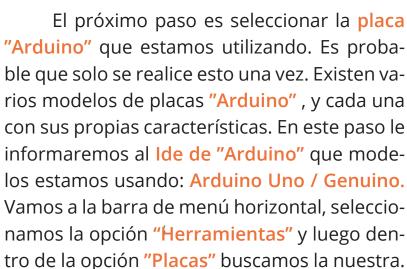


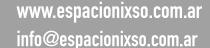




# Test de Carga PASO Nº 2 Selección de Placas











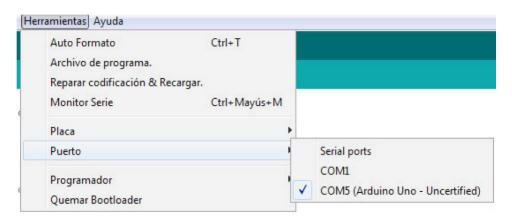


# Test de Carga PASO Nº 3 Selección de Puerto

#### Test de carga de un programa

El próximo paso es seleccionar en que puerto de la computadora hemos conectado la placa "Arduino". Esto se debe a que podemos tener conectados distintos periféricos o distintos "Arduino" simultáneamente, por ello debemos indicar cuál vamos a utilizar. También es probable que este paso lo tengamos que realizar solamente una vez o cada tanto.

Vamos a la barra de menú horizontal, seleccionamos la opción "Herramientas" y luego dentro de la opción "Puertos" buscamos allí nuestra placa conectada. Si no aparase el nombre, tenemos que probar una por una hasta dar con la correcta. Los números de puerto lo asigna el sistema operátivo de cada computadora.









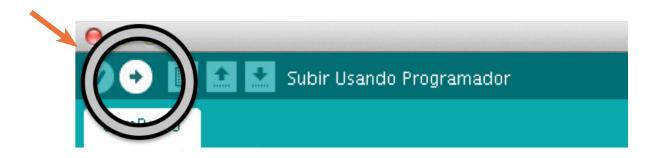




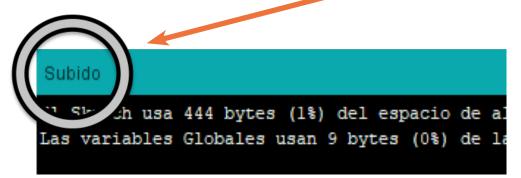
# PASO N° 4 Puerto

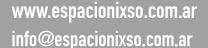
#### Cargando programa al Arduino

El ultimo paso es cargar el programa. Solamente seleccionamos el botón "Subir".



El proceso tardara unos segundos. Y finalmente en la barra superior de la terminal nos dará un mensaje de programa "Subido". El robot ejecutará el programa en unos instantes.









# arte

www.espacionixso.com.ar

info@espacionixso.com.ar

@espacionixsoarte

© @espacionixso





