



Arduino desde Physical Etoys

Debido a la popularidad y a las diversas características que tiene el Arduino, en este tutorial se verá cómo acceder a algunas de ellas desde Physical Etoys para hacer programitas sencillos y tener una agradable introducción a la electrónica.

Herramientas necesarias:

1. Una plaqueta Arduino (nosotros probamos con una Duemilanove).
2. Un cable USB.
3. El software Physical Etoys.

Herramientas optativas:

1. Un led, un pulsador, un protoboard, tres resistencias de 10 KOhms y un servomotor (para demostrar que funciona la plaqueta).

Instalación de drivers:

Lo primero que tenemos que hacer es conectar la plaqueta a la computadora vía USB. Si es la primera vez que conectamos un Arduino, Windows debería inicializar el proceso de instalación de drivers.

Con Windows Vista, el driver se baja y se instala automáticamente.

Con Windows XP, se abre el asistente para hardware nuevo encontrado:

- Cuando pregunta si desea conectarse a Windows Update para buscar el controlador debemos elegir: “no por esta vez” y clicar en siguiente.
- Elegimos “Instalar desde una lista o ubicación específica (avanzado)” y clickeamos en siguiente.
- La opción “Buscar el controlador más adecuado en estas ubicaciones” debe estar marcada y la opción “Buscar en medios extraíbles” debe estar desmarcada. Habilitamos la opción “Incluir esta ubicación en la búsqueda”, clickeamos en “Examinar” y buscamos la carpeta “Arduino\drivers\FTDI USB Drivers” de la distribución de Physical Physical Etoys. La última versión de los drivers puede encontrarse en [FTDI website](#). Clickeamos en siguiente.
- El asistente buscará el controlador y nos dirá que encontró un “USB Serial Converter”. Clickeamos en “Finalizar”.
- El asistente para hardware nuevo aparecerá nuevamente. Realizamos los mismos pasos y elegimos las mismas ubicaciones para buscar. Esta vez encontrará un “USB Serial Port”.

Conexión de Arduino a Physical Physical Etoys:



El primer paso, una vez dentro de Physical Physical Etoys, es conseguir un “Arduino board”. Éste es un objeto gráfico que representa en la computadora a la plaqueta Arduino real.

Abrimos la solapa de “Supplies”.



La ventana de supplies contiene los objetos más utilizados, conforme vayamos usando el sistema vamos a ir aprendiendo cada vez más sobre ellos. El que nos interesa ahora es el “Object catalog”, o catálogo de objetos.

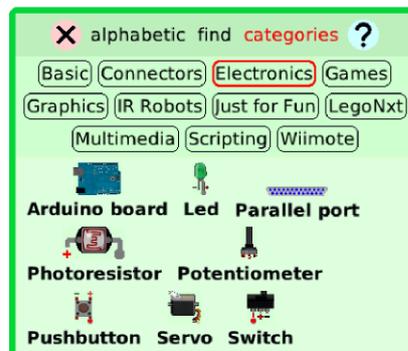


Arrastramos afuera un catálogo de objetos.



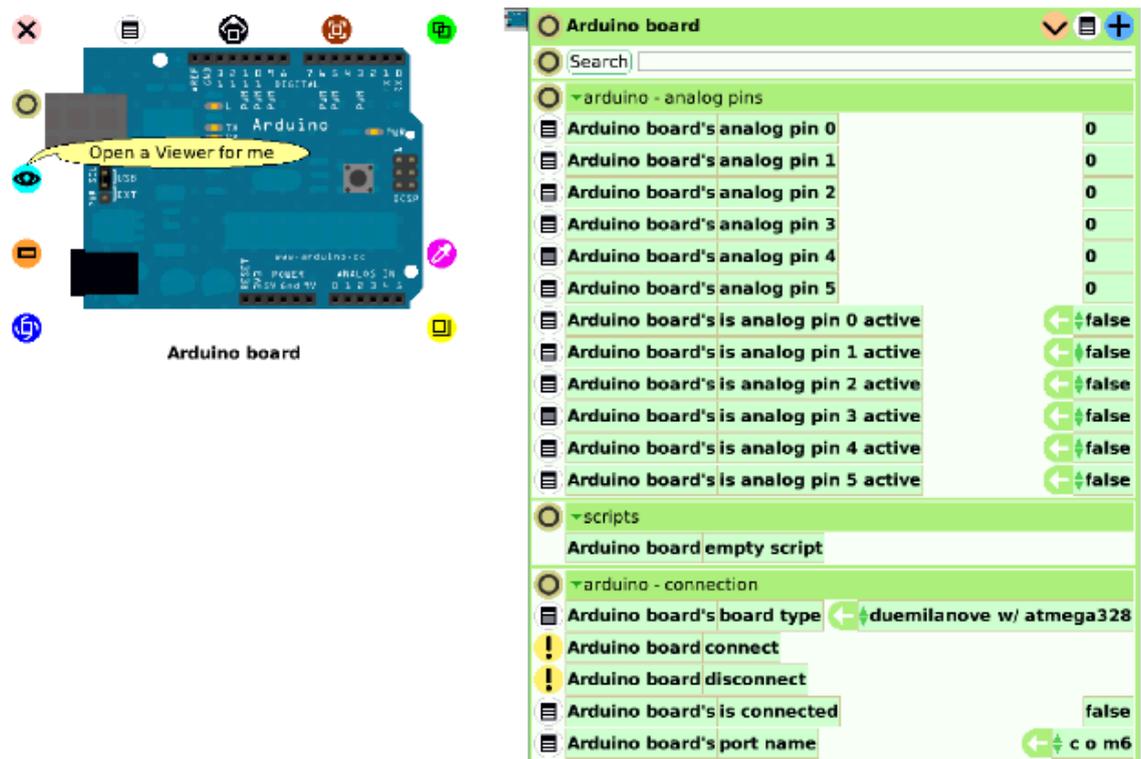
El catálogo de objetos es como una caja que contiene todos los objetos que podemos utilizar. Está ordenado por categorías para que sea más fácil de usar, pero también podemos ordenarlo alfabéticamente o buscar por un objeto particular.

Seleccionamos la categoría “Electronics” y arrastramos afuera un “Arduino board”. Como vemos, el “Arduino board” se parece bastante a la plaqueta real.

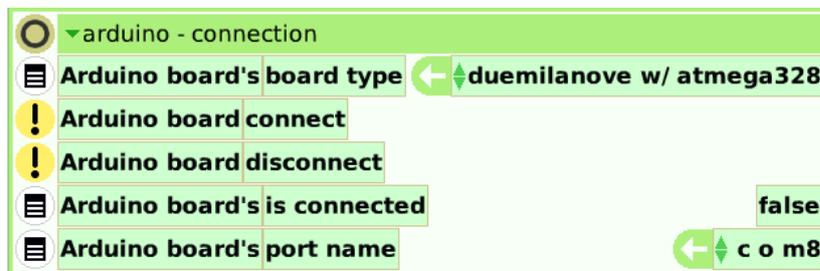


Una vez que tenemos el “Arduino board” debemos conectarlo. Para ello debemos hacer click derecho en el “Arduino board” y de esta manera se abrirá el halo. El halo es un conjunto de botones que rodean al objeto y nos permiten modificarlo, moverlo, eliminarlo, agrandarlo, etc.

Luego hacemos click en el botón celeste del halo para abrir un visor (viewer). El “Viewer” es una solapa donde podemos ver y modificar las propiedades de los objetos en pantalla así como crear guiones para que los objetos ejecuten acciones (tales como moverse por la pantalla, etc.).

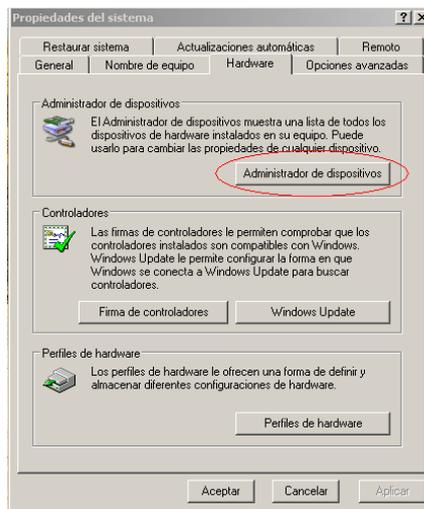


Dentro de la categoría de conexión, aparecen distintos bloques:

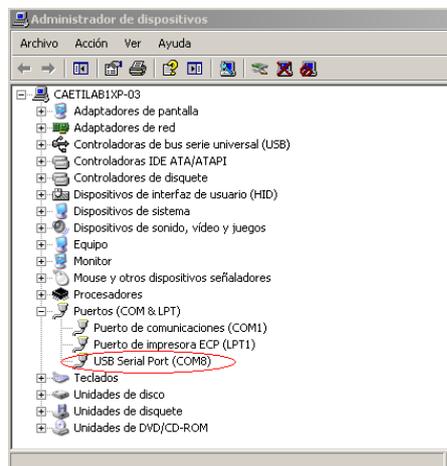




Primero hay que elegir qué tipo de placa se utilizará. En este caso elegimos la Duemilanove con el microcontrolador ATmega328. Después hay que definir el puerto COM al cual está conectada la placa. Para saber en dónde está conectada hay que hacer clic derecho en Mi PC y luego en Propiedades. Luego, dentro de la solapa de Hardware se clickea en “Administrador de dispositivos”.



Una vez dentro se busca la parte de “Puertos” y finalmente a la derecha de donde dice “USB Serial Port” estará el número del puerto. En este caso es el 8.



Por lo tanto, el bloque correspondiente al puerto quedará en COM8.

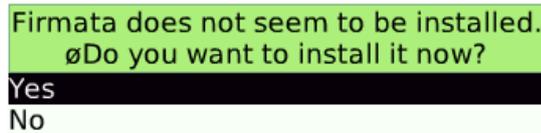




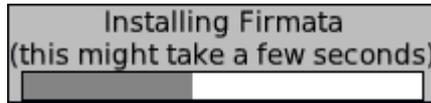
La instrucción que nos importa ahora es “connect”. Si ejecutamos la instrucción (haciendo clic en el botón amarillo de la izquierda con un signo de exclamación) conectaremos el Arduino a Physical Physical Etoys.



Si es la primera vez que se conecta la placa, aparecerá un cartel que preguntará si desea instalar Firmata. Firmata es un programa que contiene el protocolo de comunicaciones para que la placa pueda recibir órdenes. Ponemos que sí.



Una barra indicará el progreso de la instalación:

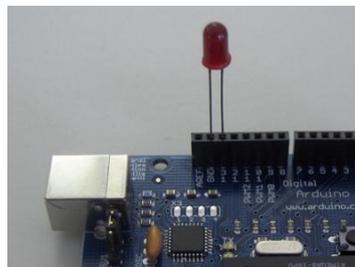


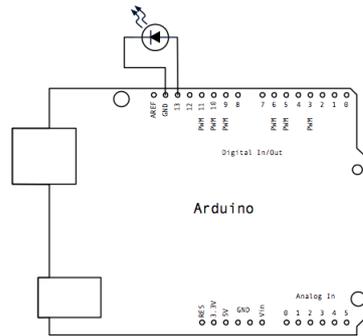
Si todo funcionó correctamente nos mostrará una leyenda diciendo que el proceso terminó correctamente. Intentamos conectar nuevamente (clickeando en el bloque “connect”) y el valor del bloque que indica si está conectado será verdadero (true):



El “Hola Mundo” de la electrónica (prender y apagar un led)

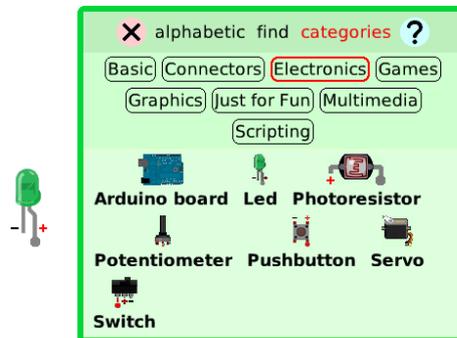
Antes de empezar a trabajar con Physical Physical Etoys debemos conectar físicamente el led a la placa. La patita más larga debe ir conectada al **pin 13** de la placa y la más corta al “ground”.



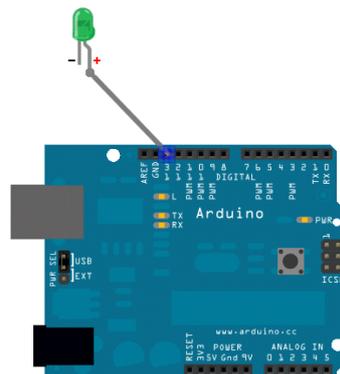


Ya podemos arrancar con Physical Etoys.

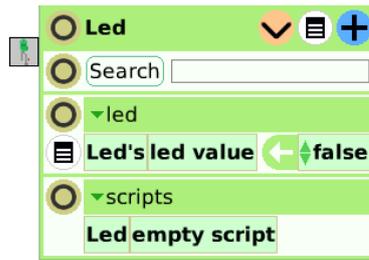
Dentro de la categoría “Electronics” del catálogo de objetos, arrastraremos un Led.



Una vez situado, clickeamos y arrastramos desde su pata positiva hacia uno de los pines de entrada digitales de la placa (en este caso el número 13). Aparecerá un cable que los conectará.



Luego entramos al visor (viewer) del led y aparecerán estos bloques:

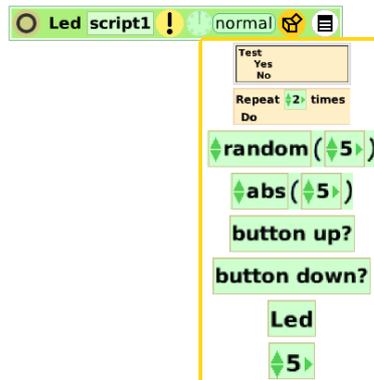


Ahora crearemos un gui3n para hacer que el led parpadee. Para crear un nuevo gui3n arrastramos el bloque “empty script” que se encuentra en la categor3a scripts. Una vez arrastrado se ver3 esto:

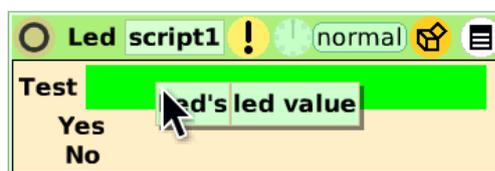


Luego clickeamos en el bot3n que tiene una caja y elegimos un bloque “test yes/no”.

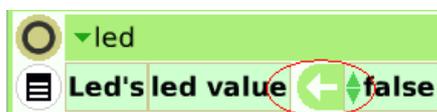
Este bloque significa que el gui3n har3 algo dependiendo de una condici3n (en este caso ser3 seg3n el valor del Led). Los que tengan un poco de experiencia en programaci3n se dar3n cuenta de que es igual a una estructura IF.

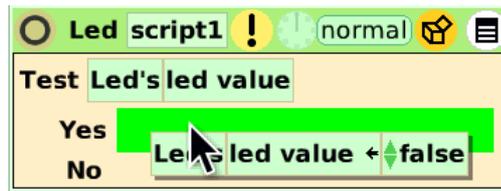


Para establecer la condici3n arrastraremos (no desde la flecha porque no es una asignaci3n) el bloque “led value” hacia la derecha de “test”:



Ahora hay qu3 establecer qu3 har3 el gui3n si el led est3 prendido. L3gicamente lo apagar3. Entonces vamos a asignarle a “led value” el valor Falso. Para ello arrastraremos desde la flecha porque es una asignaci3n.

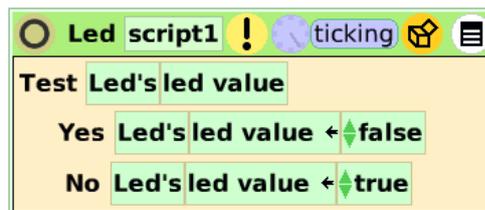




Ahora haremos lo mismo pero con una sutil diferencia: cuando el Led esté apagado, que lo prenda. Arrastramos desde la flecha el bloque “led value” hacia la parte del No pero cambiando su valor a verdadero (true):



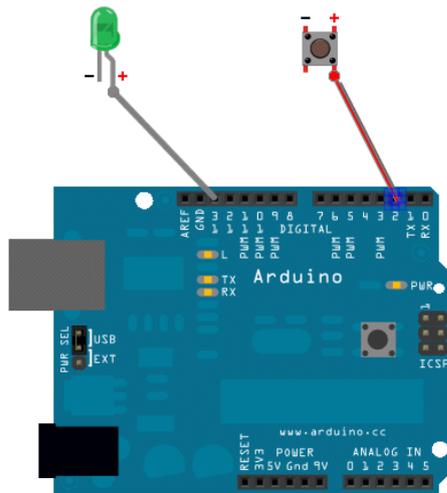
Ahora ejecutaremos el guión clickeando en el ícono con forma de reloj:



Si todo salió bien, nuestro led real debería empezar a parpadear.

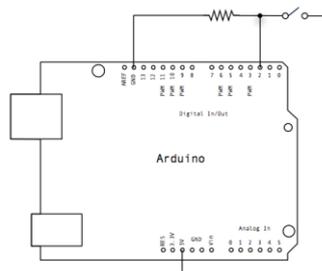
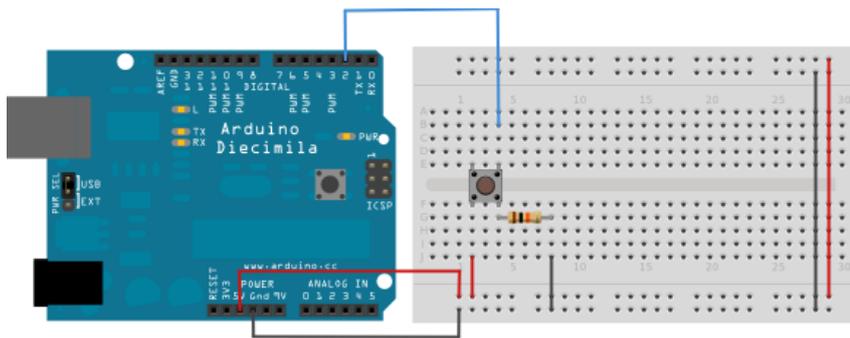
Prender un Led pulsando un botón

Dentro de la categoría Electronics arrastramos un “push button” y lo conectamos al pin 2 de la “Arduino Board”:

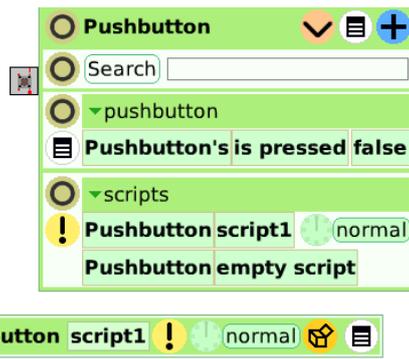


Acordarse de no desconectar el led del pin 13 y de parar su script correspondiente. Físicamente la conexión del pulsador (con la resistencia y el protoboard) debería verse así:

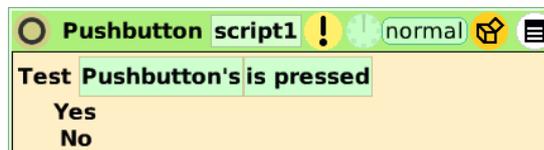
así:



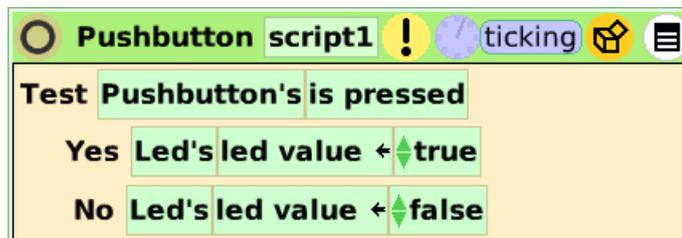
Ahora entramos al visor del Pushbutton y creamos un nuevo script:



Agregamos un bloque "Test yes/no" para saber cuándo está presionado el pulsador. Como no es una asignación el bloque "isPressed" lo arrastraremos desde el nombre en lugar de la flechita.



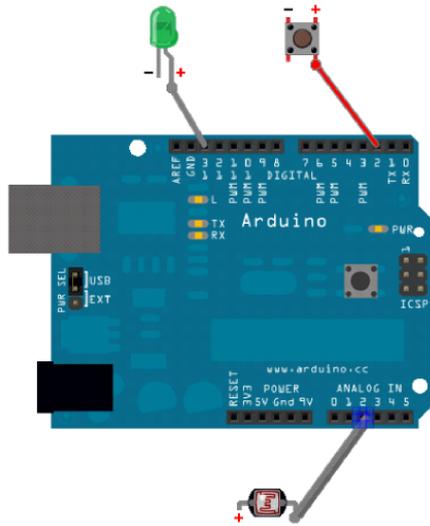
En caso que esté presionado el pulsador le asignaremos un valor verdadero al Led y si no uno falso. Acordarse de arrastrar los bloques (que se encuentran en el visor del led) desde la flechita porque representan asignaciones. Una vez que esté realizado, clickeamos en el reloj para que se ejecute el guión.



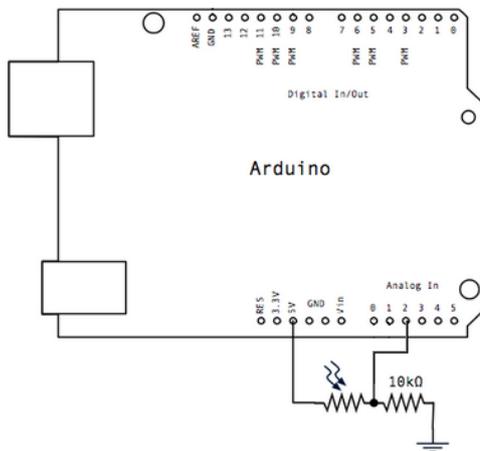
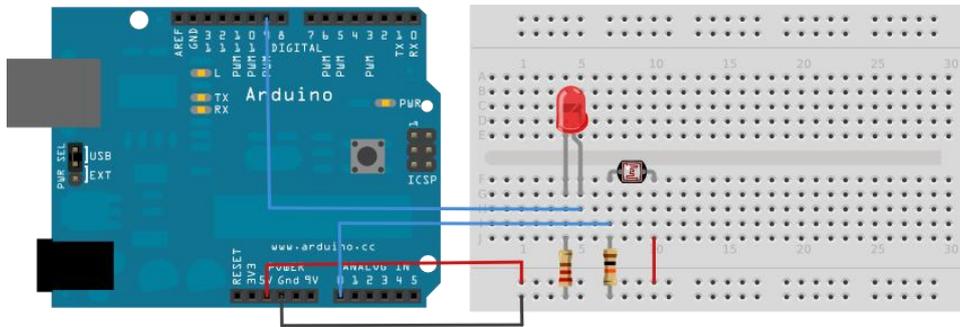
Si todo fue realizado correctamente, cuando pulsemos el botón, el led se prenderá.

Leyendo los valores de un Fotorresistor

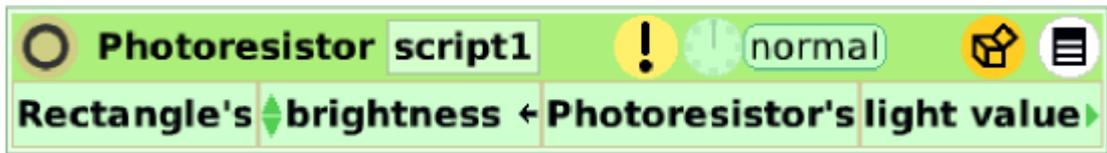
Como ya tenemos el "Arduino Board" en pantalla conectado, le añadiremos un Fotorresistor a la segunda entrada analógica.



La conexión real debería ser así:



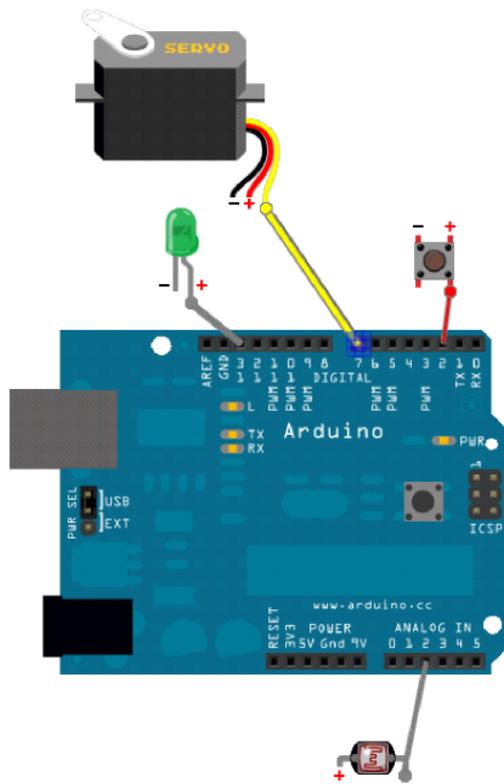
Abrimos el viewer del fotoresistor y generamos un nuevo guión. Ahora tenemos que ir al catálogo de objetos y extraer de la categoría basic un "Rectangle". Abrimos el viewer del "Rectangle" y vamos a la categoría "color" para extraer (desde la flechita) el bloque de brightness hacia el guión del Fotorresistor. Ahora abrimos el visor del Fotorresistor y sustituimos el valor del brightness del "rectangle" (100) por el tile "Photoresistor's lightvalue" (no desde la flechita porque sólo queremos leer el valor).



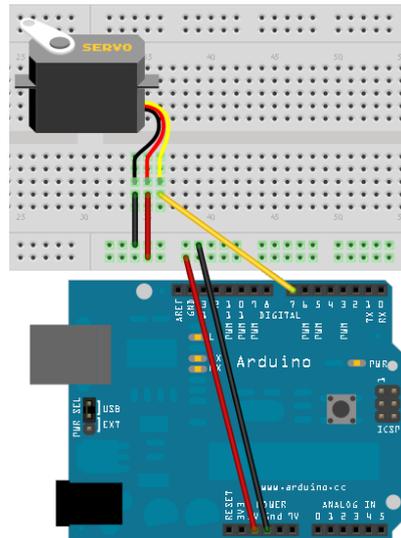
Ejecutamos el script y veremos como el “rectangle” cambia su brillo según la luz ambiente.

Rotar un Servo con un rectángulo

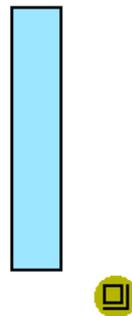
Como ya tenemos el “Arduino Board” en pantalla conectado por haber realizado los ejercicios anteriores conectaremos un motor Servo (que se encuentra en la categoría electronics) al pin 7 del “Arduino Board”:



Físicamente la conexión es así:



Luego para ahorrar tiempo podemos utilizar el rectángulo creado en el ejercicio anterior. Abrimos su halo y arrastramos el botón amarillo para cambiar su tamaño.



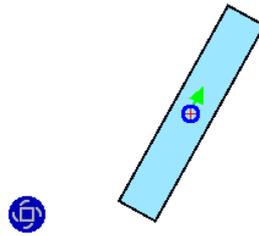
Ahora abrimos el visor del servo y generamos un nuevo guión con el bloque de los grados (degrees) del servo. Recordar que es una asignación por lo tanto se lo arrastrará desde la flechita.



Luego abrimos el visor del "Rectangle" y dentro de la categoría basic arrastramos (no desde la flechita porque nos interesa consultar el valor en vez de cambiarlo) el bloque "heading". Una vez hecho este paso, ejecutamos el guión.



Con el guión corriendo, abrimos el halo del "Rectangle" y arrastramos el botón azul para girarlo.



Si todo se realizó correctamente el motor girará según la orientación del rectángulo.

Conclusión

Bueno, eso es básicamente todo lo que necesitamos saber para empezar a usar la placa Arduino. Las posibilidades de interacción entre la computadora, Arduino y los distintos componentes electrónicos que provee Physical Etoys son demasiadas para cubrirlas todas en este pequeño tutorial. Lo que queda por delante para poder explotar al máximo esas posibilidades es explorar el ambiente: probar, jugar, tocar y romper si es necesario.

¡Que se diviertan!