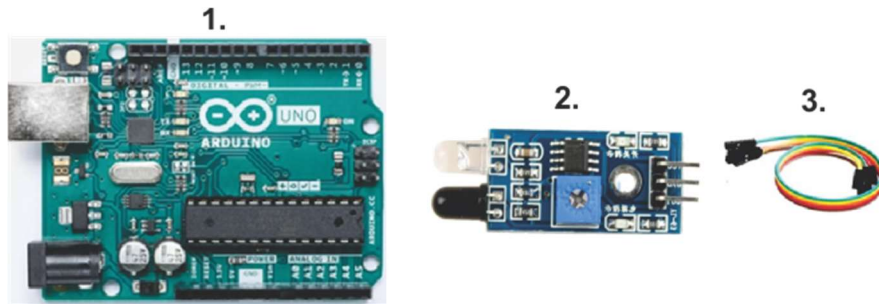


## Práctica 10: Sensor de presencia de objetos o personas

Para esta práctica se necesita:

1. Placa Arduino UNO
2. Módulo sensor de obstáculos FC-51
3. Cables para realizar las conexiones



### Introducción

En esta práctica utilizaremos un sensor de obstáculos basado en un módulo IR FC-51. Este módulo contiene un emisor y un receptor infrarrojo, un LED indicador y un potenciómetro (resistencia ajustable) para definir la distancia a la que se activa el sensor. El objetivo es controlar el encendido y apagado del LED que trae la placa Arduino UNO.

El módulo se puede utilizar en variedad de proyectos como detectar el estado de una puerta (abierta o cerrada), detector de obstáculos, seguidores de línea o incluso robots mini sumos y es compatible con Arduino y microcontroladores que proporcionen una alimentación de 5 volts.

Los sensores infrarrojos se componen de un LED emisor infrarrojo (IR) y un receptor que detecta la luz reflejada por un objeto. De esta manera se puede determinar la presencia de un objeto cercano ya que la luz infrarroja reflejada es capturada por el LED receptor.

### Montaje

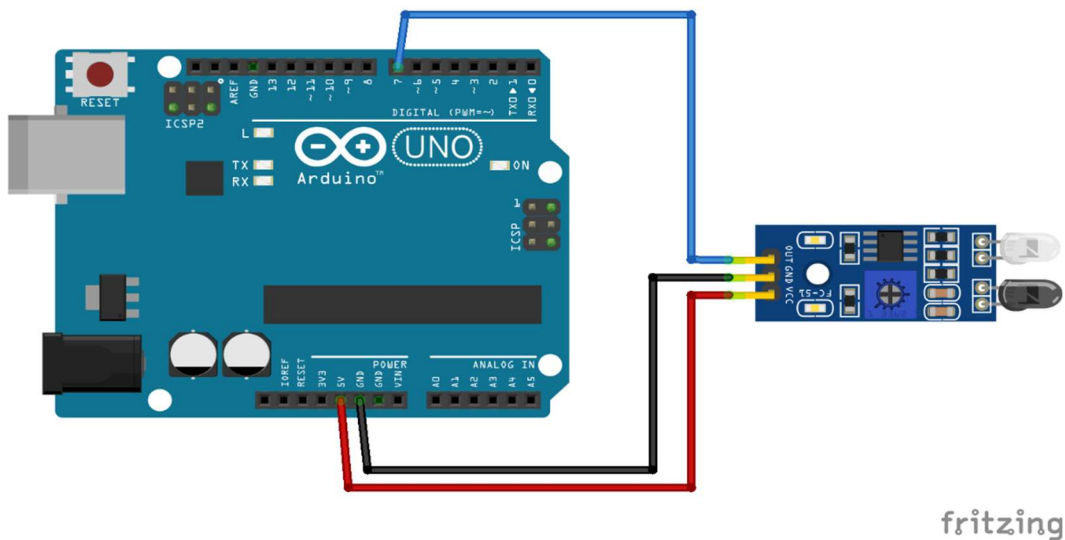
Las partes principales del módulo se puede ver en la figura 1.



Fig. 1 Sensor IR FC-51

Realizar la conexión del sensor IR FC-51 cableando el terminal **GND** al pin **GND**, el terminal **"Vcc"** a **+5V**, el terminal **"+OUT"** al pin **5** de la placa Arduino.

Sensor de obstáculos	Pin Arduino
<b>GND</b>	<b>GND</b>
<b>Vcc</b>	<b>+5V</b>
<b>+OUT</b>	<b>PIN 7</b>



## Programación

Para llevar a cabo nuestro programa plantearemos primero la lógica que debe tener. Recordemos que el objetivo es controlar el encendido y apagado del LED que trae incorporado la placa Arduino UNO. Cada vez que se detecte un objeto dentro de la zona de alcance del sensor se cambia el estado de la salida 13 (LED de la placa Arduino UNO) a encendido (ON), de lo contrario el LED de permanecer en apagado (OFF).

Un dato importante para tener en cuenta es que la salida del módulo (PIN **+OUT**) va a estar en **5V (HIGH)** si no detecta un objeto, y cambia a **0V (LOW)** si se acerca un objeto.

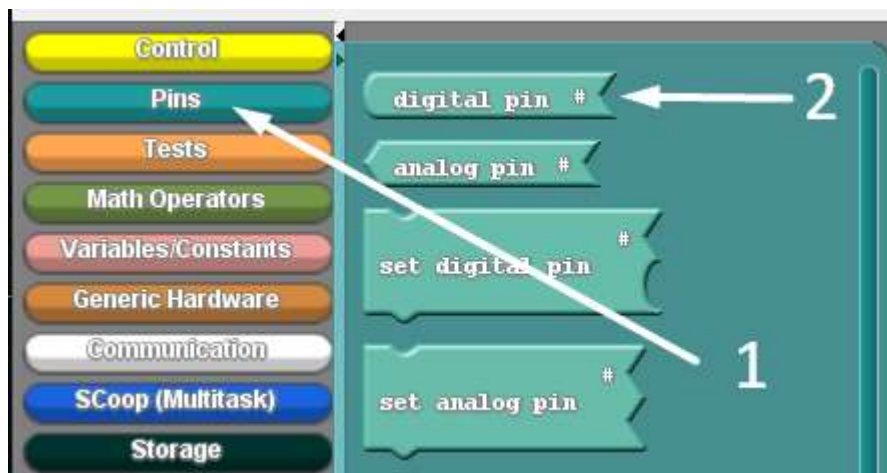
Con esta claridad podemos iniciar la construcción de la secuencia del programa. El planteamiento sería el siguiente:

1. Leer la entrada de Arduino al que va conectado el PIN **+OUT**.

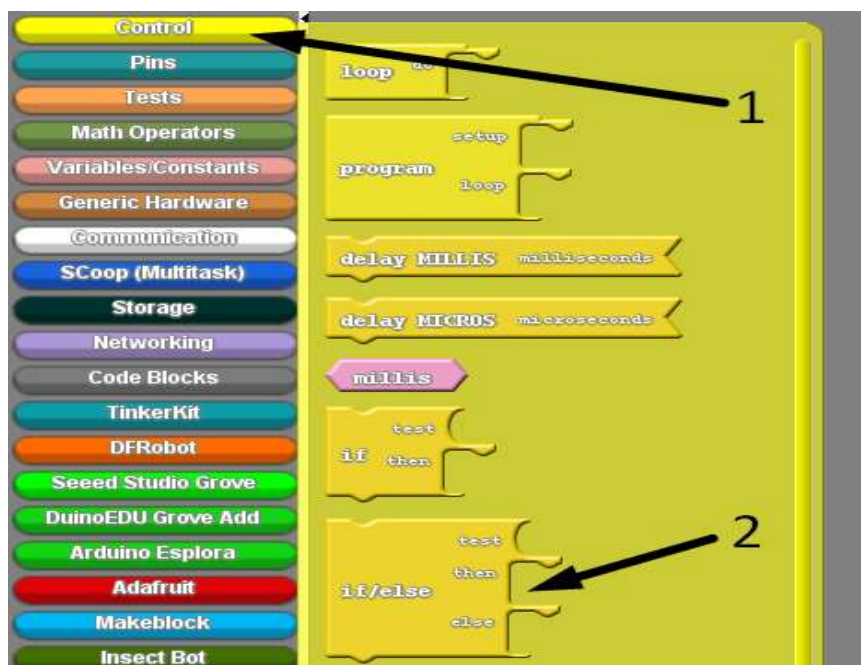
2. Comparar que estado tiene (HIGH= No hay objeto, LOW=Hay objeto)
3. Si entrada es LOW encender el LED de la tarjeta Arduino (pin 13), sino apagar el LED.
4. Adicionalmente colocaremos un tiempo de espera antes de realizar la siguiente comprobación

Con la secuencia del proceso pasamos a construir el código en Ardublock.

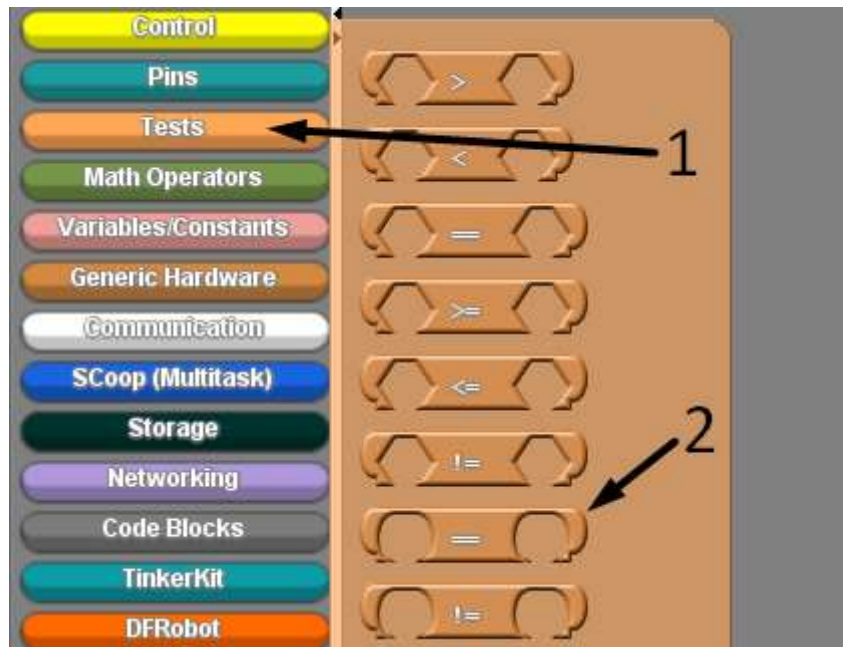
1. Seleccionar el bloque para leer un pin digital 7:  
Ir al botón **"Pins"** del panel izquierdo y seleccionar el bloque **digital pin**.



2. Colocar el bloque condicional **if/then** (si/entonces)



3. Colocar el operador de comparación “==” para evaluar que dos valores booleanos son iguales.



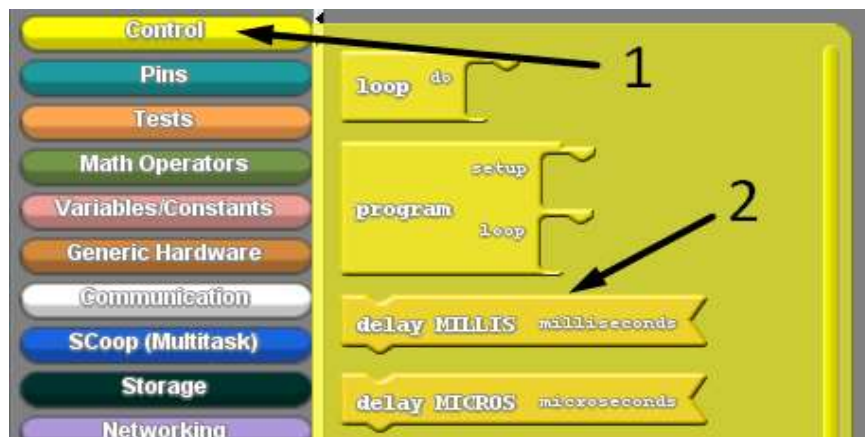
4. Colocar un valor booleano de estado **LOW** para comparar con la lectura del pin 7



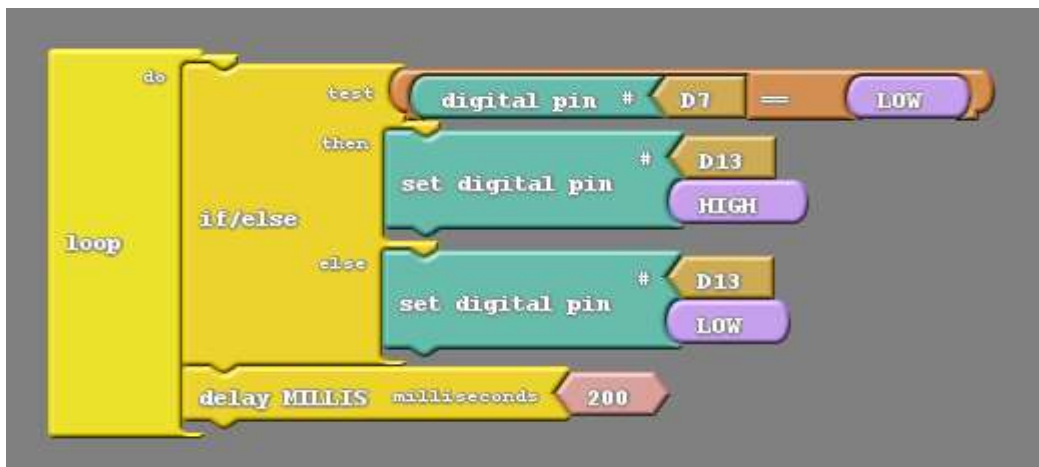
5. Colocar dos bloques para activar/desactivar una salida digital. Un bloque para el encendido (ON) y el otro para el apagado (OFF).



6. Colocamos un bloque para un retardo de 200 milisegundos.



Finalmente, nuestros bloques de programación nos deben quedar de la siguiente manera:



## **Ampliación**

¿Qué crees que pasaría si se aumenta el tiempo del retardo a 2000 o más milisegundos?

¿Qué bloques se deben añadir al programa para encender una alarma sonora (Buzzer=zumbador) en lugar de LED incorporado en la placa?