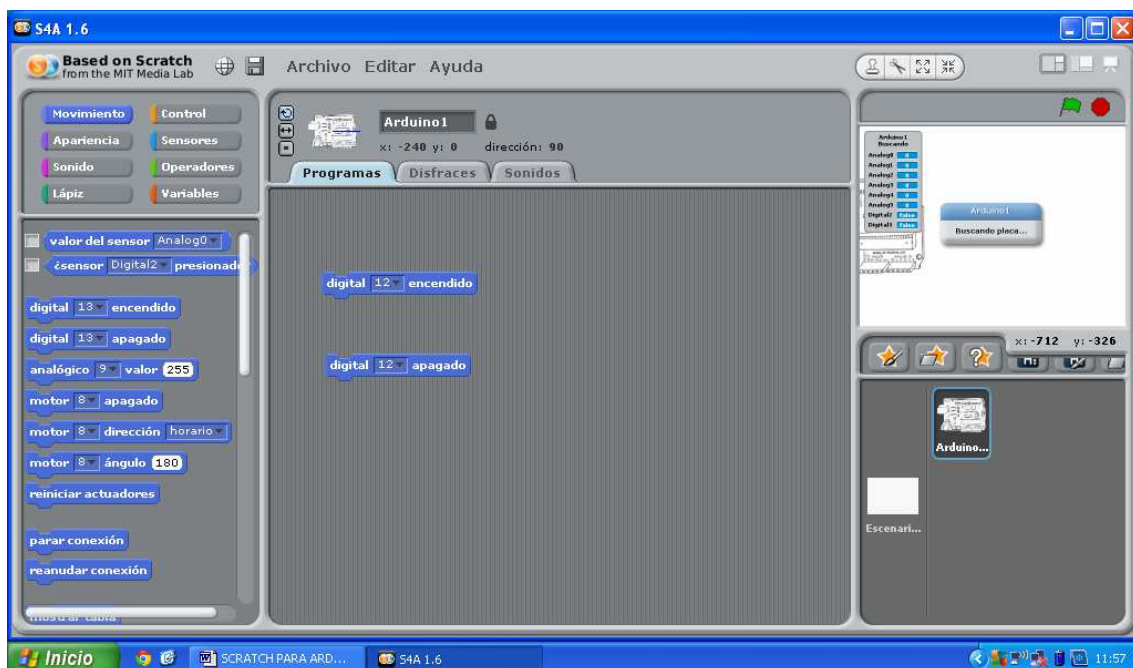
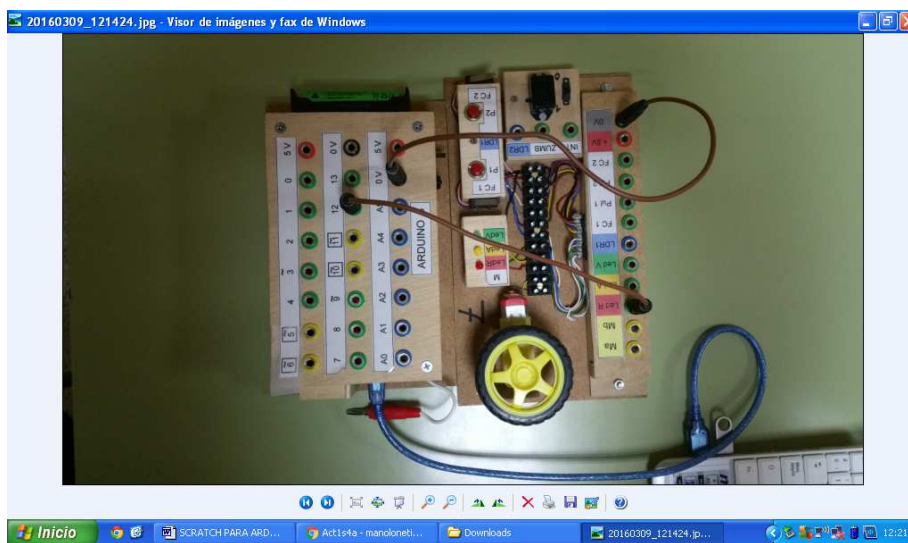


## ACTIVIDADES DE SCRATCH PARA ARDUINO (S4A) 3º ESO TECNOLOGÍAS

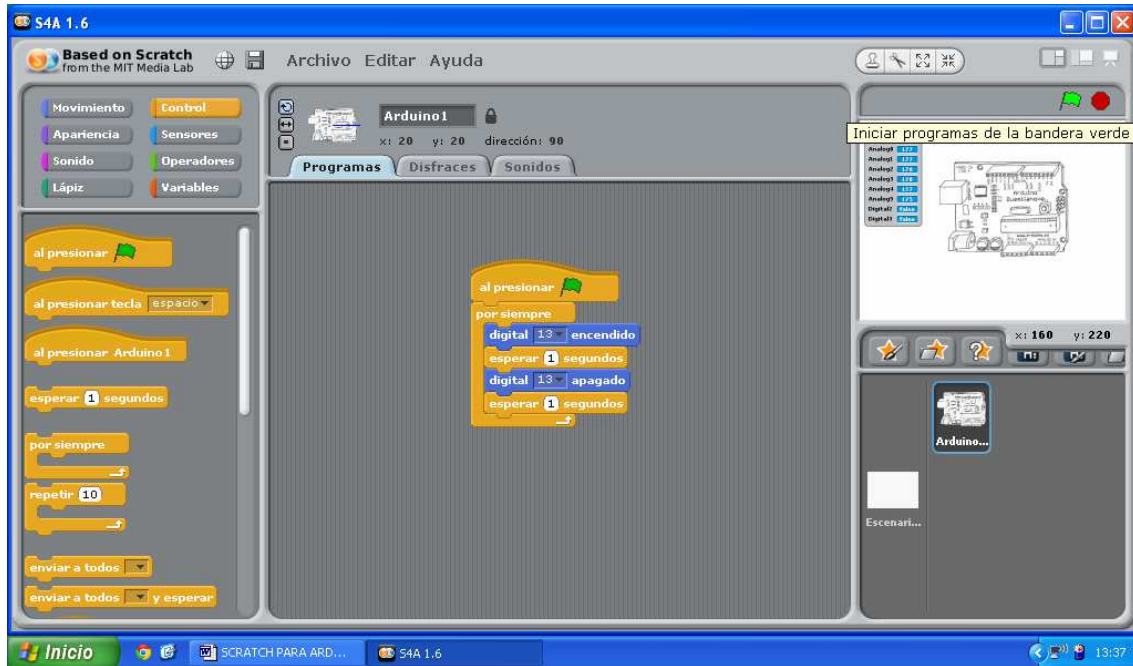
### A1: ENCENDER UN LED Y APAGARLO CON S4A

Conecta la placa arduino al ordenador y arranca S4A. Recuerda que la placa arduino tiene que tener instalado el firmware para S4A. Cuando haya detectado la placa, en la pestaña Programas del objeto Arduino1, mete las órdenes digital 12 encendido y digital 12 apagado. Cuando hagas doble clic sobre una de estas órdenes, el programa mandará la orden a la placa arduino y pondrá a nivel alto o bajo la salida 12, donde habremos conectado el led de la placa de pruebas.



## A2: HACEMOS TITILAR UN LED

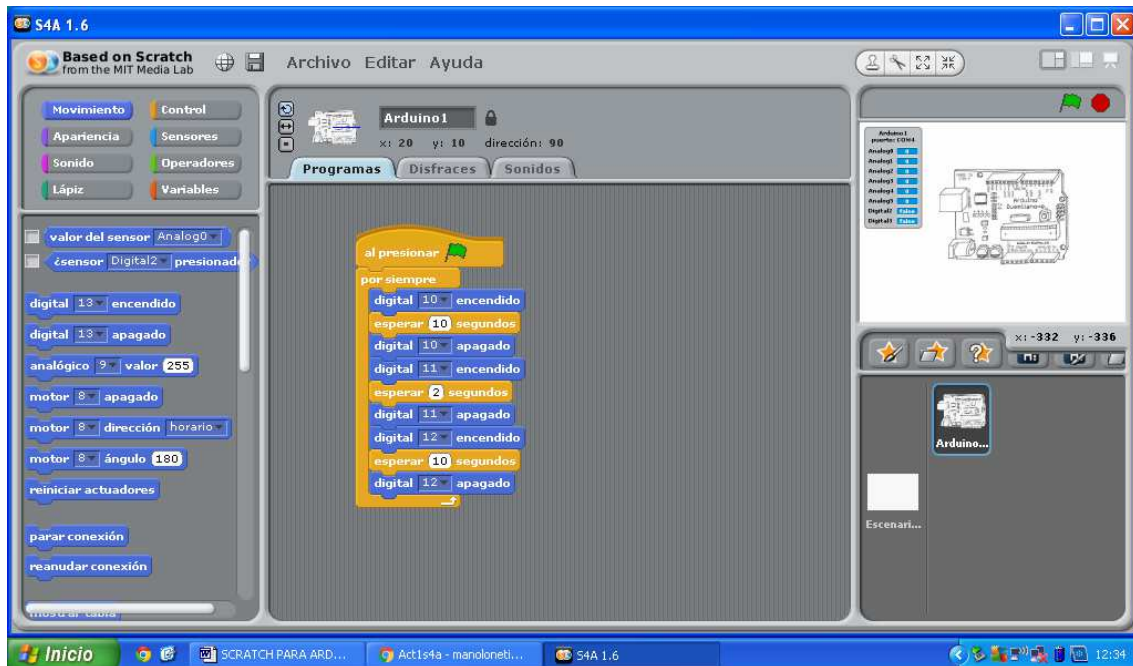
Conectaremos al pin 12 de arduino un led rojo. Luego realizaremos un programa en S4A tal que el presionar la bandera verde, el led se encienda y se apague con una frecuencia de 1 segundo.



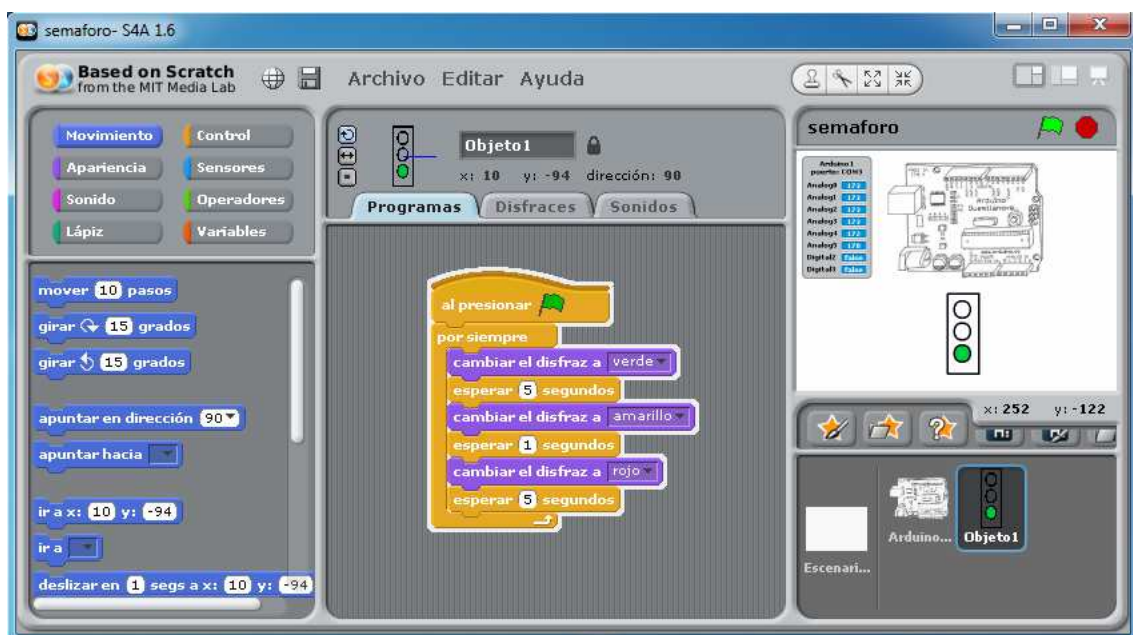
Cuando acabemos esta actividad, podemos hacer pruebas variando la frecuencia de encendido y apagado y observando cual es la frecuencia máxima que el ojo humano puede percibir.

### A3: HACEMOS UN SEMÁFORO CON LEDS

En esta actividad vamos a conectar la placa arduino a tres leds (verde, amarillo y rojo) en las salidas digitales 10, 11 y 12 y realizaremos en S4A un programa que realice el comportamiento de un semáforo.



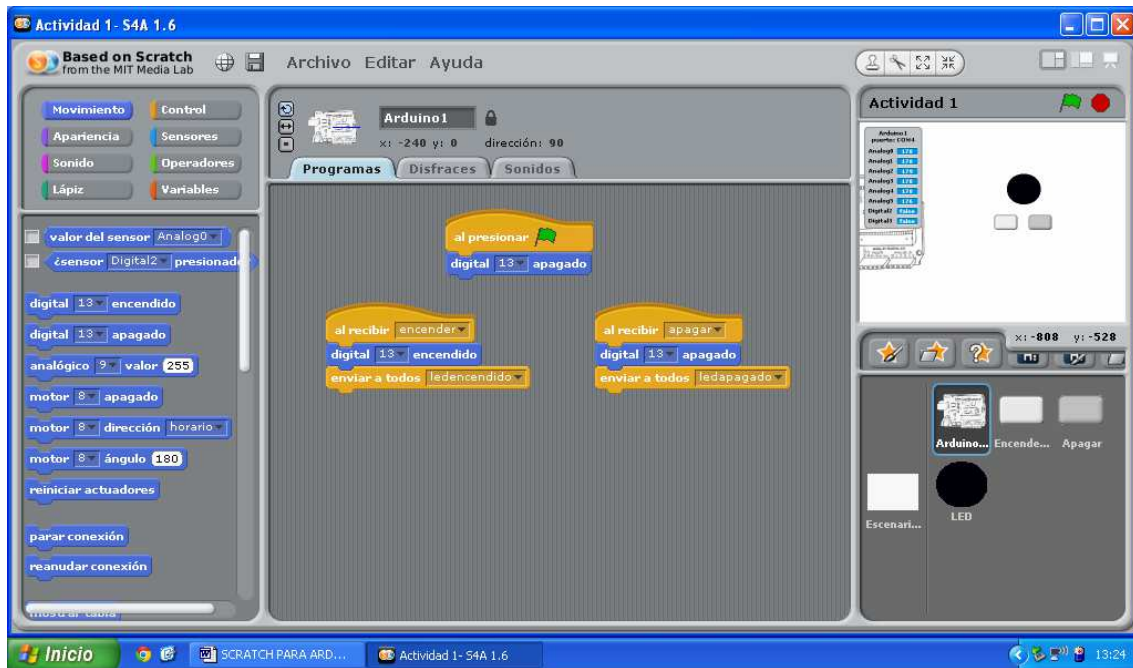
Ahora realizaremos el mismo semáforo sólo que de forma virtual con la interfaz de Scratch. Para ello habrá que crear un nuevo objeto, en el que dibujaremos un semáforo con tres disfraces diferentes; uno para el verde, otro para el amarillo y otro para el rojo. El programa debería tener el siguiente aspecto:



## A4: ENCENDER Y APAGAR UN LED CON EL ORDENADOR

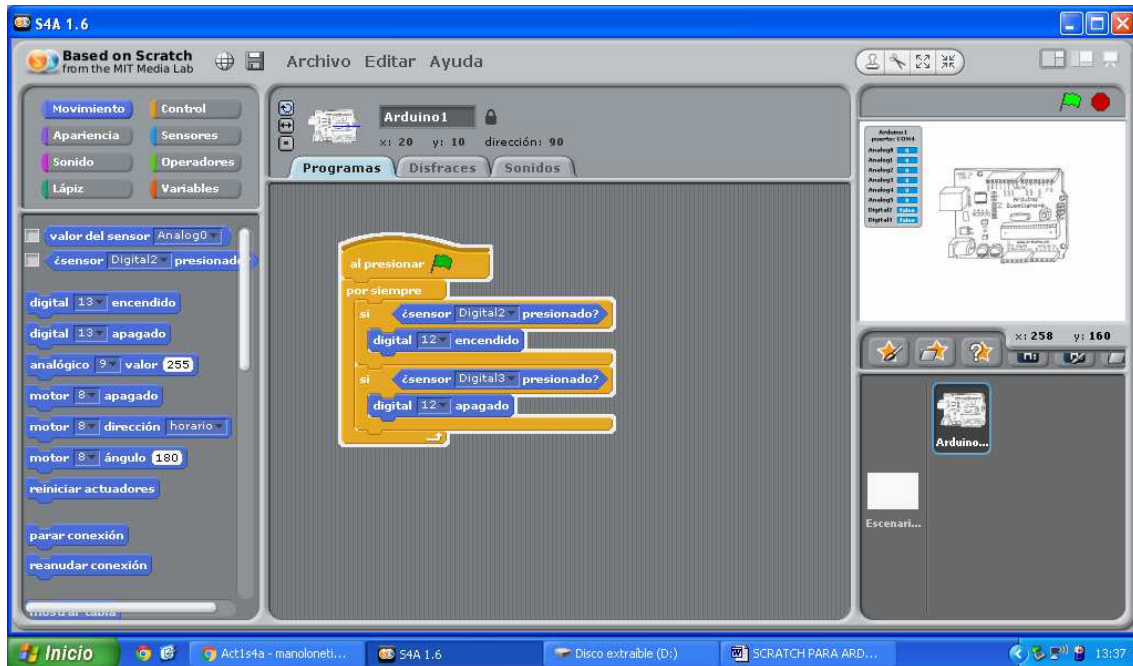
En esta actividad vamos a conectar la placa de Arduino al ordenador y después de meterle el software necesario para que sea reconocida en Scratch para Arduino (S4A), iniciaremos dicho programa.

A continuación crearemos un objeto llamado led que tendrá dos disfraces, uno para el led encendido y otro para el led apagado. Luego crearemos dos objetos en S4A, uno que al pulsarlo hará que el led se encienda y otro que al pulsarlo hará que el led se apague.



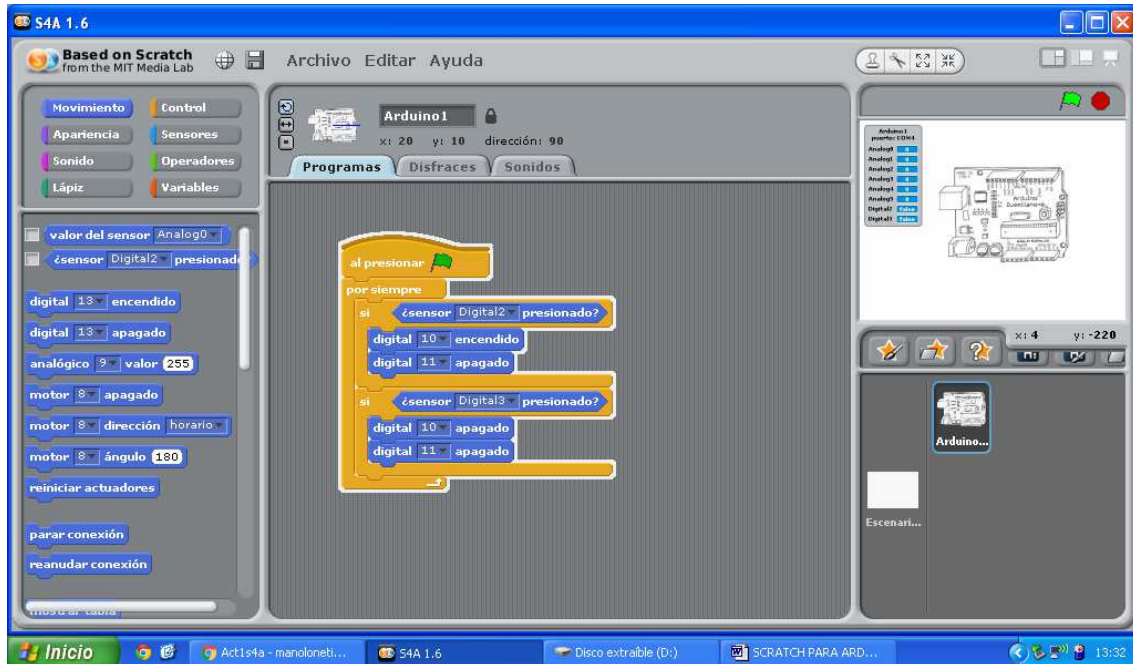
## A5: ENCENDER UN LED PULSANDO UN FINAL DE CARRERA Y APAGARLO PRESIONANDO EL OTRO FINAL DE CARRERA.

Conecta el led rojo a la salida 12 de arduino, el final de carrera 1 a la entrada digital 2 y el final de carrera 2 a la entrada digital 3. No te olvides de conectar a 5V y a 0V arduino con la placa de pruebas encendida.



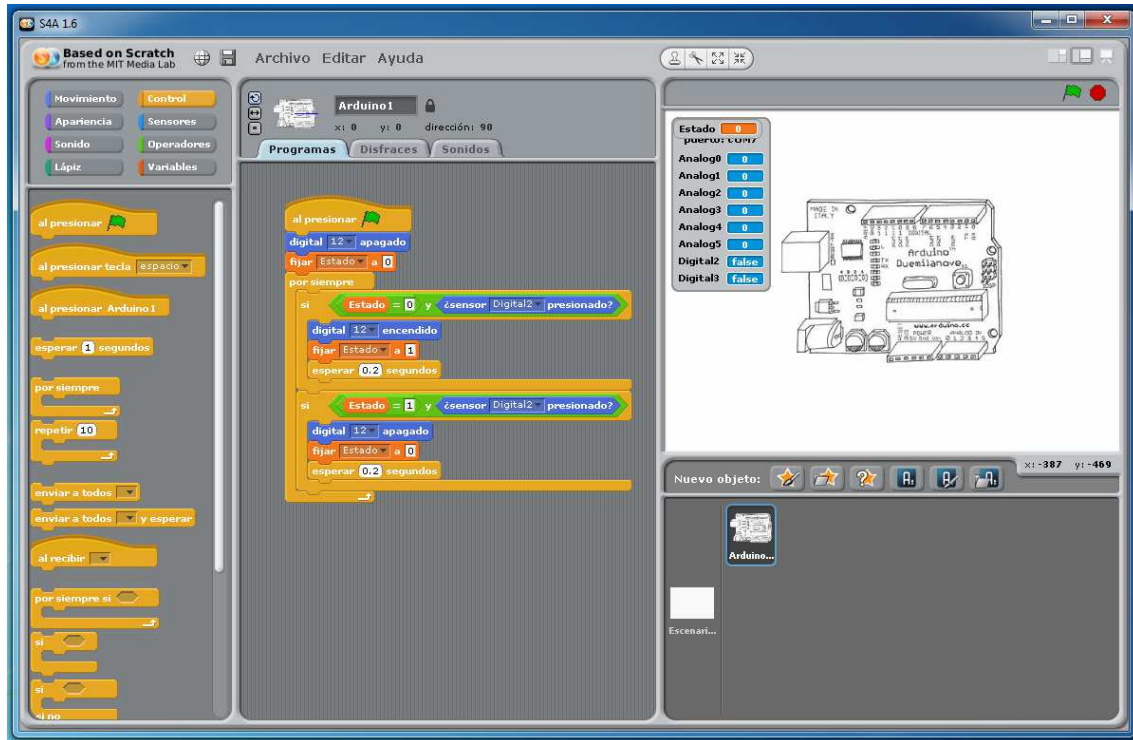
**A6: ACTIVAR O DETENER UN MOTOR UTILIZANDO DOS PULSADORES.**

Conecta el pulsador 1 a la entrada digital 2 y el pulsador 2 a la entrada digital 3. Luego conecta el motor a las salidas digitales 10 y 11. No te olvides de proporcionar 5 V y 0V a la placa de pruebas.



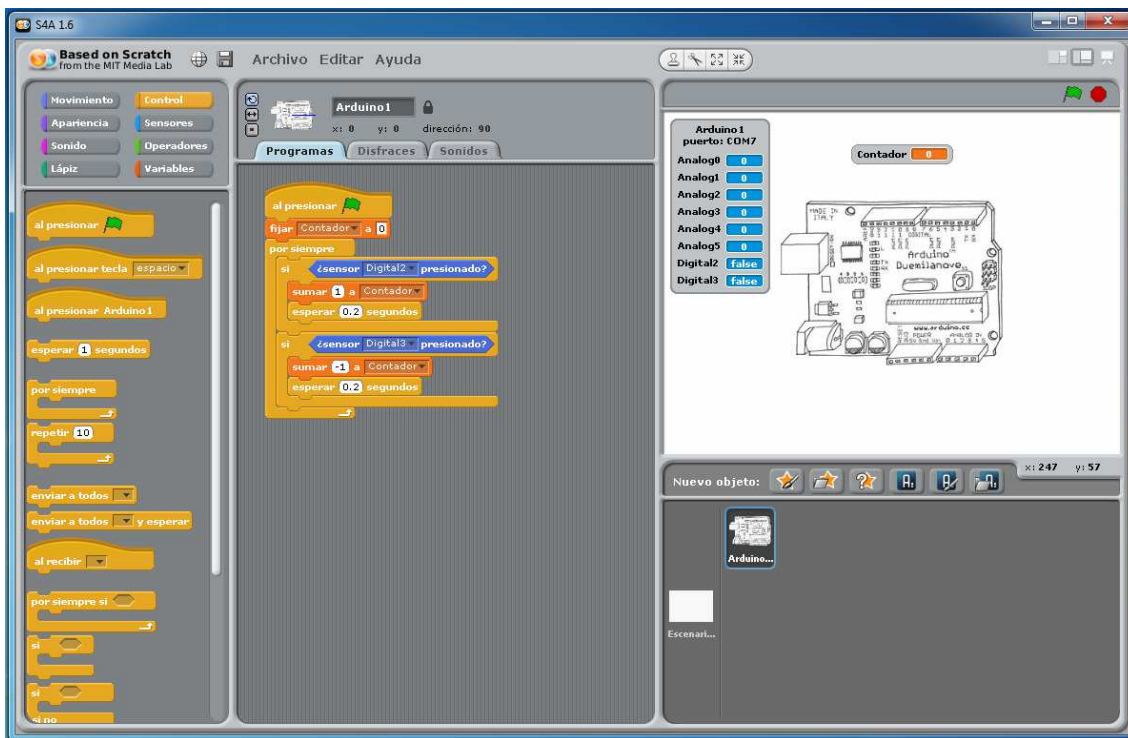
**A7: ENCENDER O APAGAR UN LED UTILIZANDO EL MISMO PULSADOR**

Conectaremos la salida 12 de arduino a un led y luego conectaremos un pulsador a la entrada 2. El programa en S4A funcionará de forma que al presionar la bandera, cuando pulsemos el pulsador, se encienda el led, al volver a pulsarlo, el led deberá apagarse y así sucesivamente. Para lograrlo deberás crear una variable llamada Estado que estará a 0 si el led estaba apagado y a 1 si el led estaba encendido. Esta es la manera de tener memoria sobre las pulsaciones.

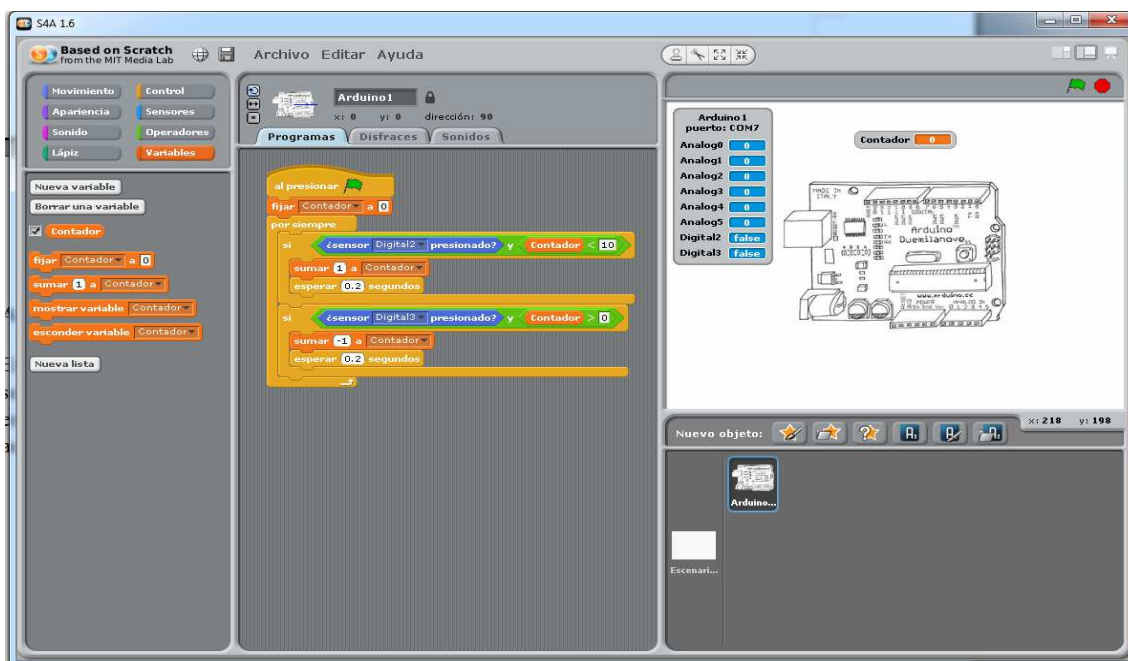


### A8: PARKING CON PULSADORES

Queremos hacer el contador de entrada y salida de un parking mediante 2 pulsadores. Cuando pulsemos uno de ellos, una variable mostrada por pantalla se incrementará en una unidad, y cuando pulsemos el otro pulsador, la variable se decrementará.



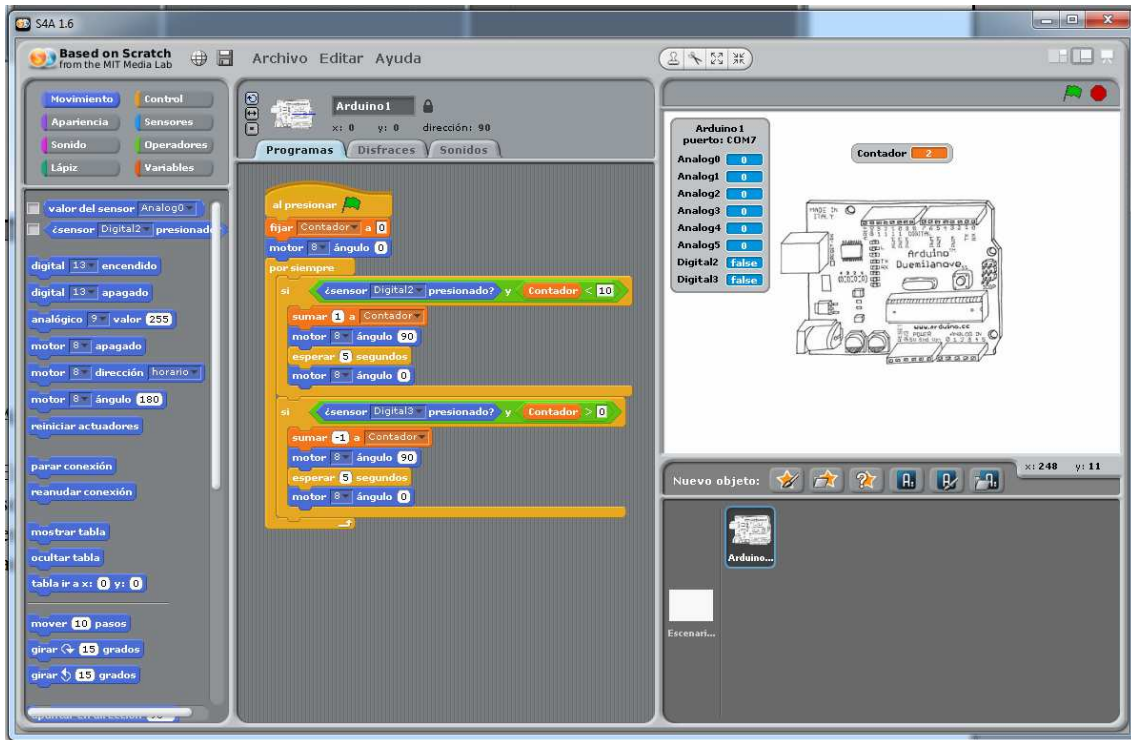
Una vez conseguido, puedes mejorarlo de manera que el contador nunca pueda ser menor que 0 y tampoco pueda ser mayor que 10 (el aforo máximo del parking).





**A9: SERVOMOTOR DE POSICIÓN. AÑADIENDO BARRERA AL PARKING.**

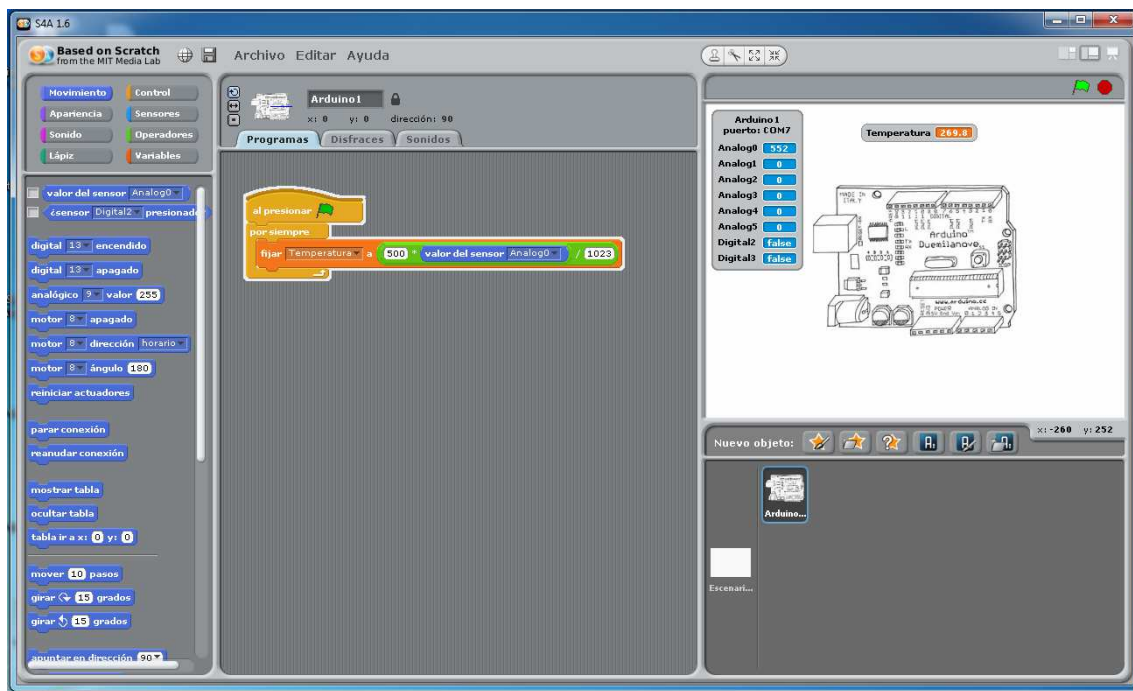
En la actividad 8, creamos un parking con pulsadores. Ahora vamos a añadirle un servomotor que sea capaz de levantar la barrera cada vez que pulsemos el pulsador de entrada y siempre que el parking no esté lleno. Al cabo de 5 segundos la barrera, bajará automáticamente. Lo mismo ocurrirá al pulsar el pulsador de salida.



## A10: MIDIENDO LA TEMPERATURA

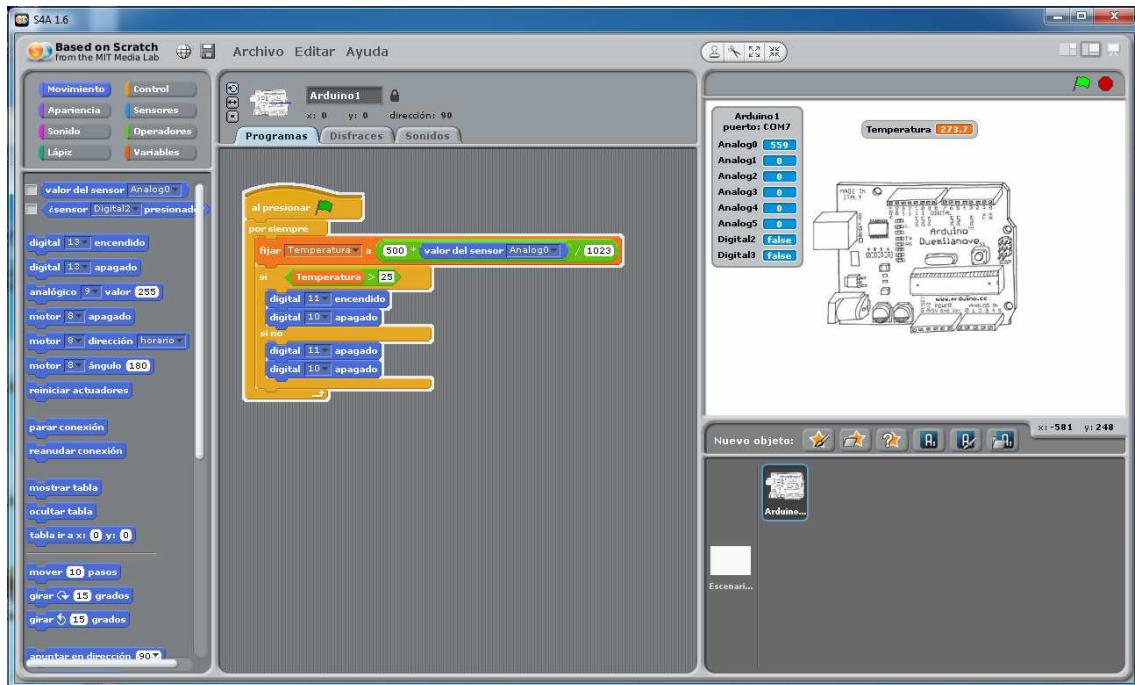
Vamos a realizar un programa en S4A que tenga una variable que nos muestre la temperatura de la sala. Para ello utilizaremos un sensor LM35. Este sensor tiene una patilla de salida cuyo valor oscila entre 0 y 5V. Cada 0,01 V, aumenta 1° C la temperatura, por lo que  $\text{Temp} = \text{Vout} \times 100$ . Sabiendo que la entrada analógica A0 de arduino puede oscilar entre 0 (para 0 V) y 1023 (para 5V), podemos concluir que:  $\text{Temp (}^\circ\text{C)} = 500 \cdot \text{Lectura Analógica} / 1023$ .

Así quedaría el programa:



### A11: CONTROL DE TEMPERATURA

Realizar un sistema en S4A que controle la temperatura de forma automática, de forma que cuando la temperatura suba de 25°, se ponga en marcha un ventilador de PC que enfriará el sensor LM35. Al bajar de 25°, se apagará el ventilador.



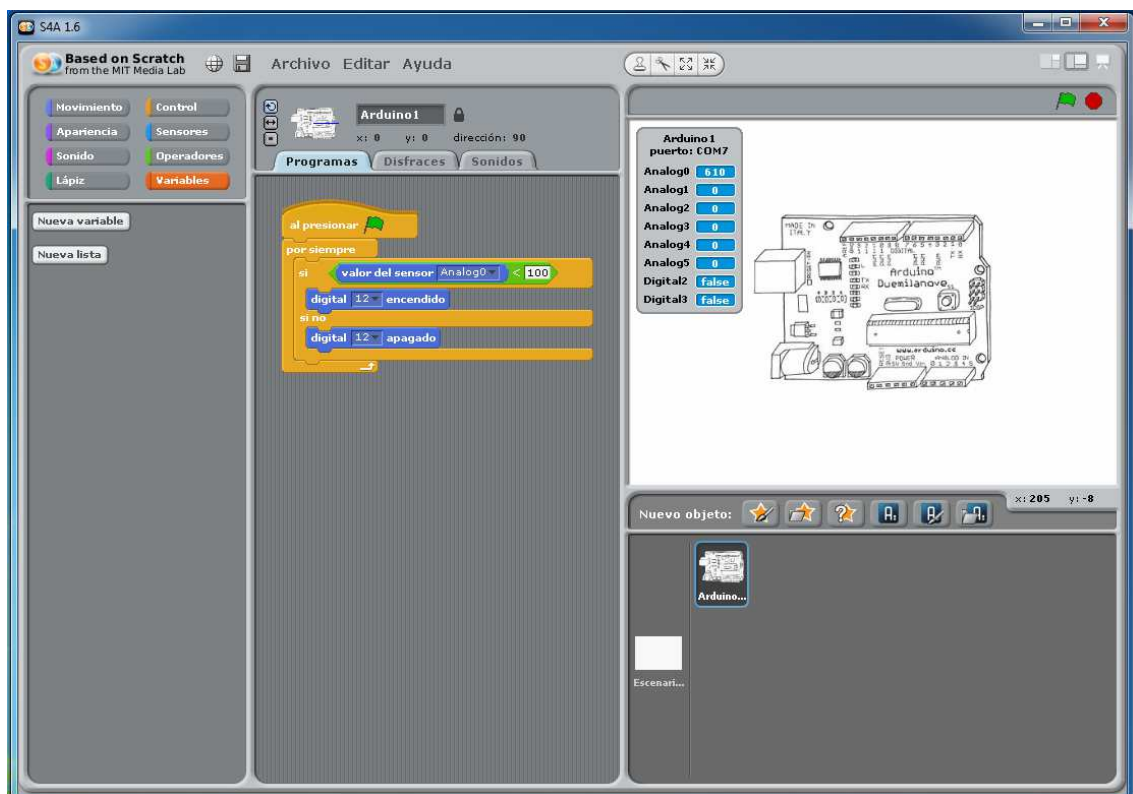
## A12: EL SENSOR DE LUZ. LA LDR

Vamos a realizar un programa que encienda un led cuando la LDR apenas reciba luz y que lo apague cuando la LDR sí reciba luz.

La LDR es una resistencia que depende de la luminosidad que le llega. Cuanto más luz le llega, menor es su resistencia y viceversa.

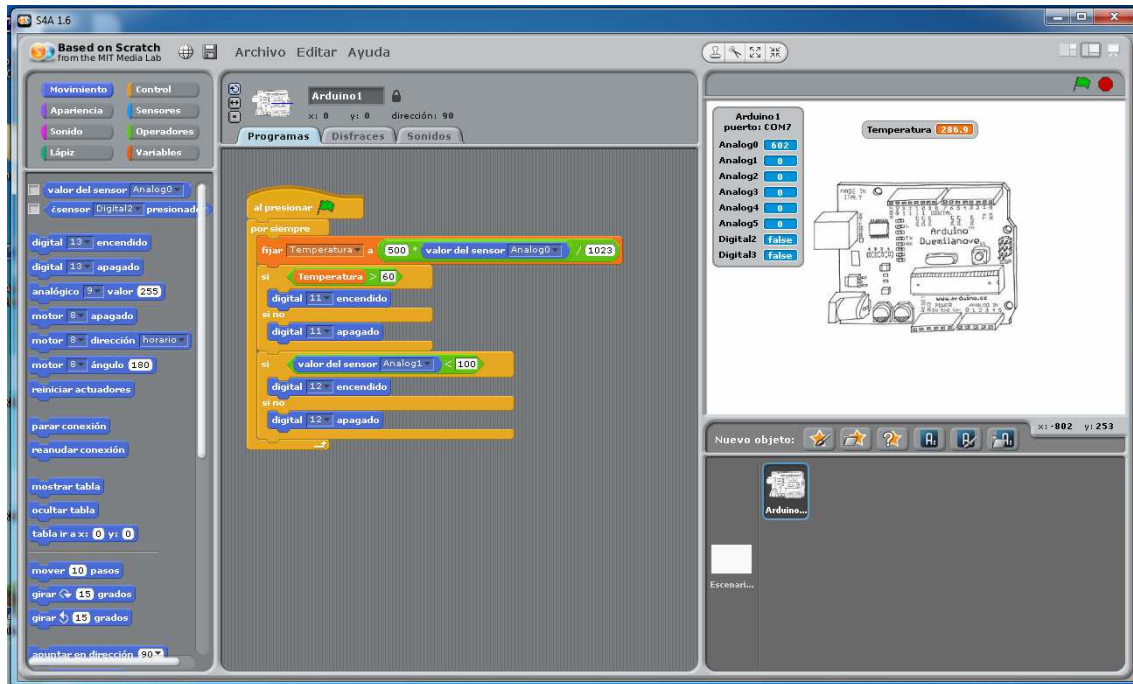
Para medir los valores que da una LDR necesitamos una entrada analógica de S4A. Podemos utilizar A0 (Analog 0). Su valor puede estar comprendido entre 0 y 1024. Cuando el valor se acerca a 0, está registrando poca entrada de luz y cuando se acerca a 1024, hay mucha entrada de luz.

El programa en S4A quedaría de la siguiente manera:



### A13: CASA INTELIGENTE

Queremos hacer una casa inteligente en la que la luz del porche se encienda automáticamente al hacerse de noche y tenga además una alarma de incendios. Para el control automático de la luz, emplearemos una LDR, mientras para la alarma de incendios, usaremos un sensor de temperatura LM35.



**A14: CONTROLA CON S4A EL PROYECTO QUE REALIZASTE EN LA SEGUNDA EVALUACIÓN.**

En esta última actividad, harás un programa en S4A que permitirá controlar el proyecto que realizaste en la segunda evaluación. Se valorará la recreación de un entorno virtual en S4A.