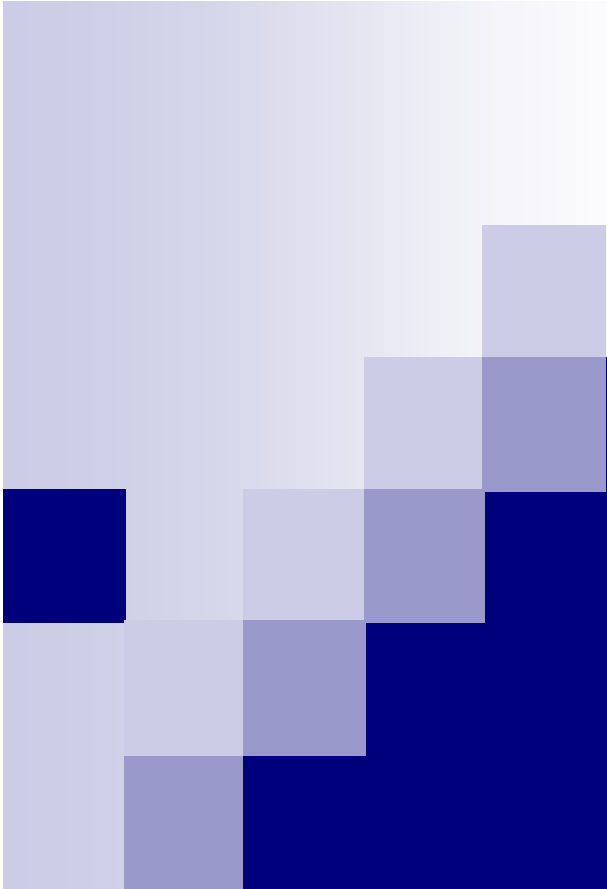




Robótica Sistema PICAXE

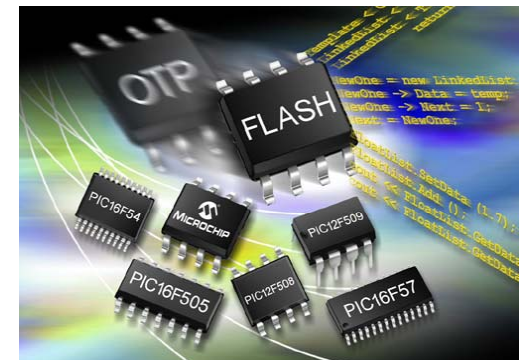
Curso:

**La gestión del Aula-Taller de
Tecnología en la ESO**

Introducción

No exageramos demasiado si decimos que los microcontroladores forman parte de nuestra vida, especialmente en lo referente al "confort". Hoy en día, es casi imposible pensar en un dispositivo electrónico que no contenga, por lo menos, un chip de proceso.

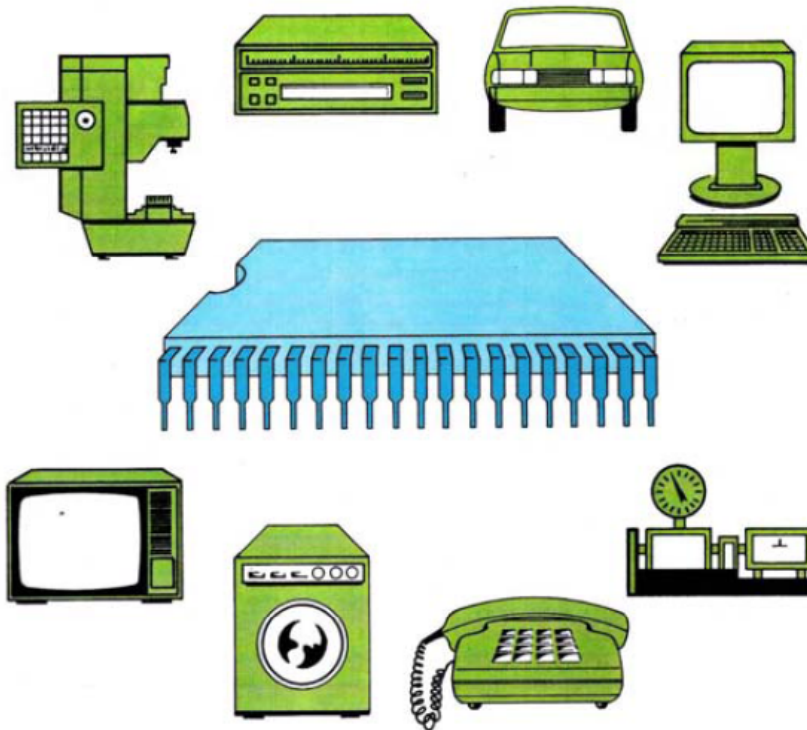
El sistema **Picaxe** nos permite trabajar con estos dispositivos de forma sencilla y con un sistema de programación muy amigable.



Introducción



Introducción



Instrumentación: equipos de medida
Robótica

Aparatos electrodomésticos:

microondas, refrigeradores, hornos, TV, reproductores y grabadores de CD y DVD, equipos de sonido, teléfonos,...

Equipo de Cómputo: impresoras, módems, unidades de disco, ratones, teclados,...

Automóviles: mando de sistemas del automóvil (ABS, inyección, encendido, climatizador..)

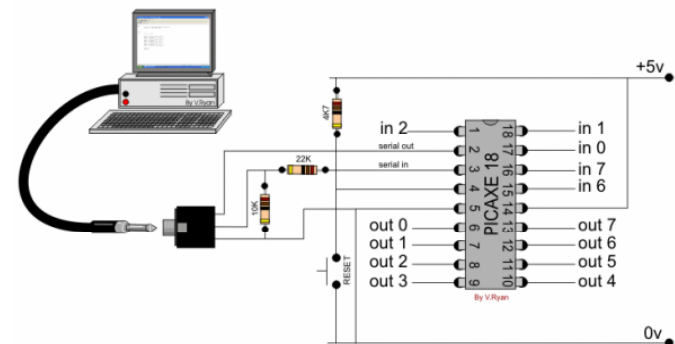
Domótica: sistemas antirrobo, climatizadores, .. **los sistemas de supervisión, vigilancia y alarma en los edificios utilizan estos chips para optimizar el rendimiento de ascensores, calefacción, alarmas de incendio, robo, etc..**

El sistema PICAXE

El sistema de desarrollo PICAXE hace las cosas aún mas sencillas para el programador, ya que cuenta con dos opciones de diseñar una aplicación:

- una por medio de diagramas de flujo o flujogramas
- otra por medio de "BASIC"

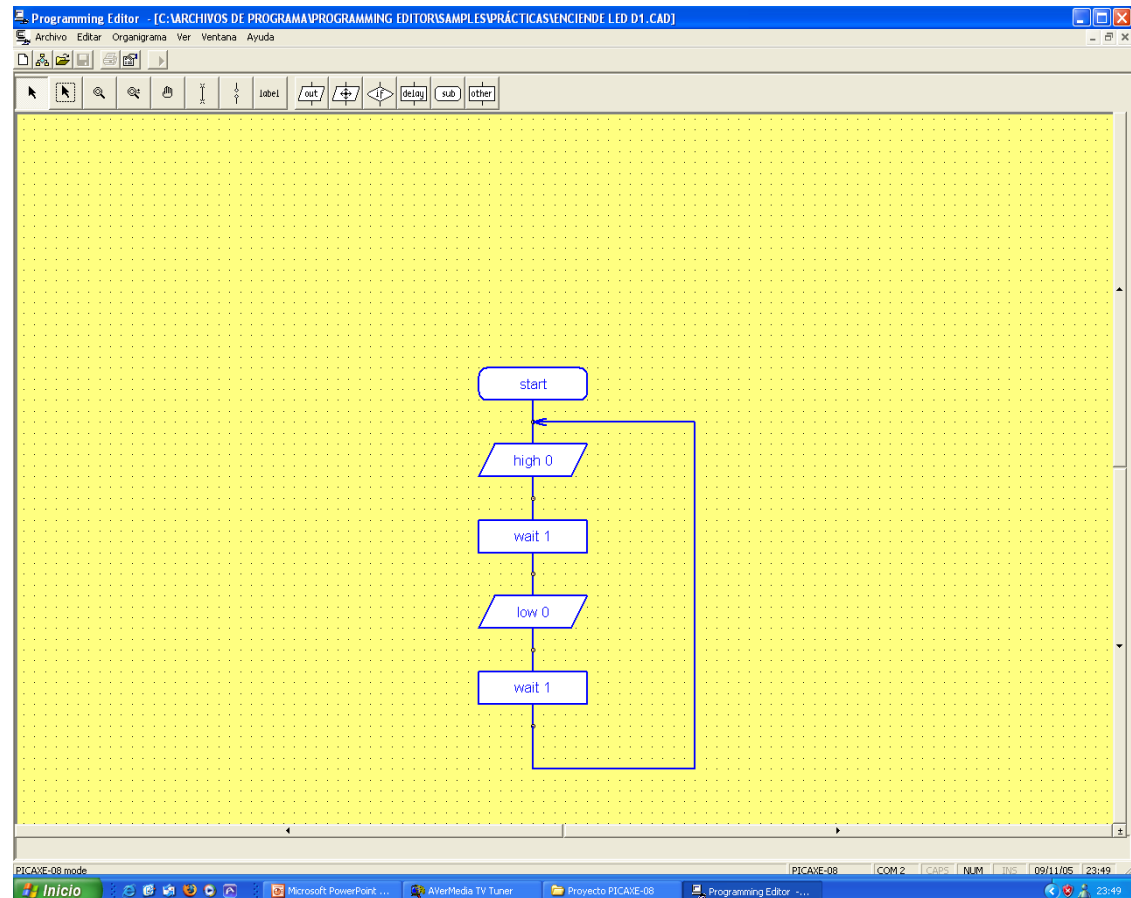
aunque esto no es ninguna novedad, (ya que estas herramientas existían con anterioridad), lo ventajoso de PICAXE radica en el hecho de que se trata de un microcontrolador PIC que, en un segmento de memoria ROM interna le fue grabado desde su fabricación, un firmware a modo de BIOS que simplifica la forma de programarlo.



El sistema PICAXE

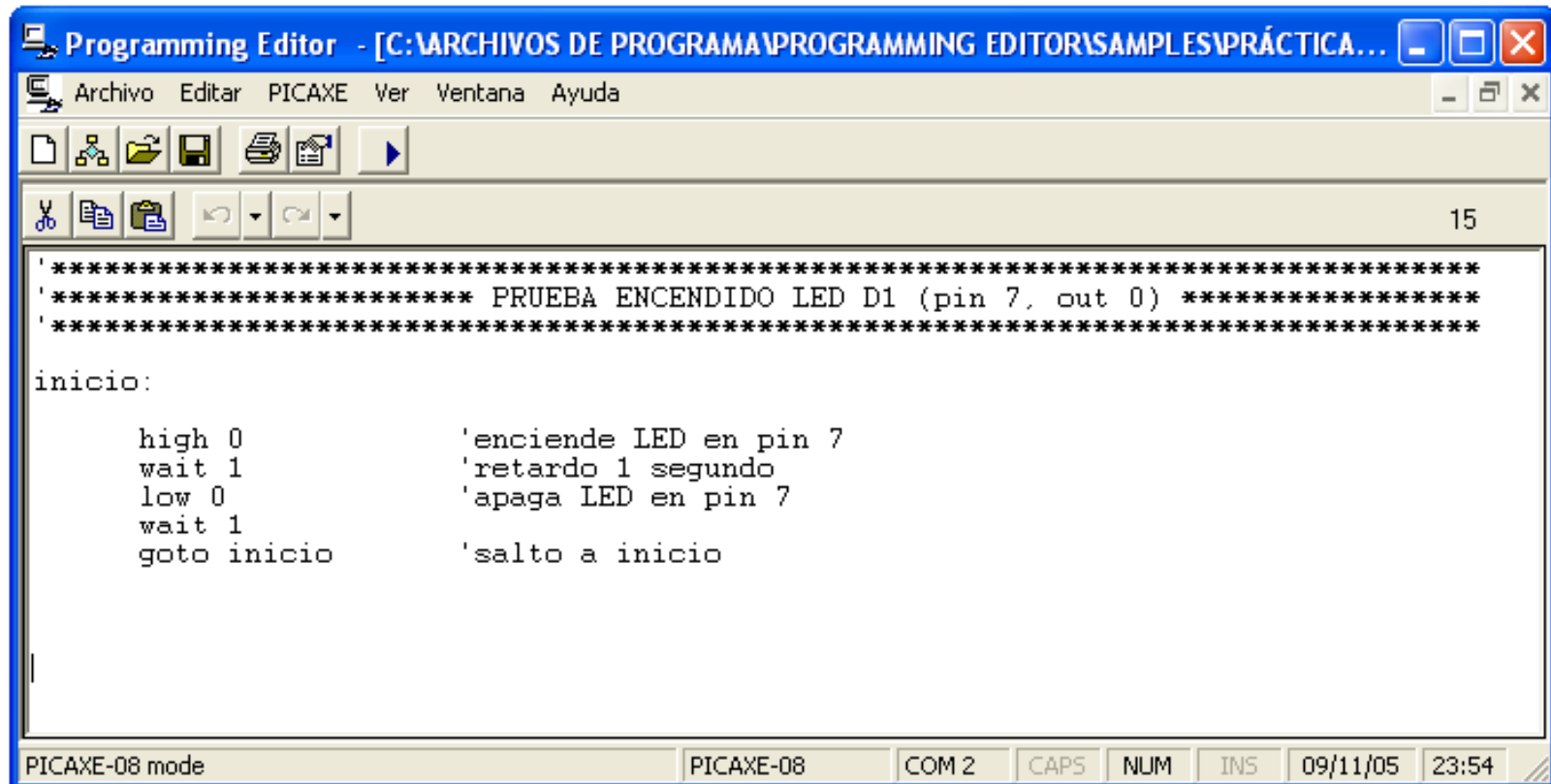
El sistema PICAXE hace mas accesible la programación de microcontroladores utilizando dos métodos de programación:

- Mediante Diagramas de flujo:



El sistema PICAXE

- O mediante el lenguaje BASIC:



The screenshot shows the 'Programming Editor' window for PICAXE. The title bar indicates the file path: [C:\ARCHIVOS DE PROGRAMA\PROGRAMMING EDITOR\SAMPLES\PRÁCTICA...]. The menu bar includes 'Archivo', 'Editar', 'PICAXE', 'Ver', 'Ventana', and 'Ayuda'. The toolbar contains icons for file operations and execution. The main text area displays a BASIC program with the following code:

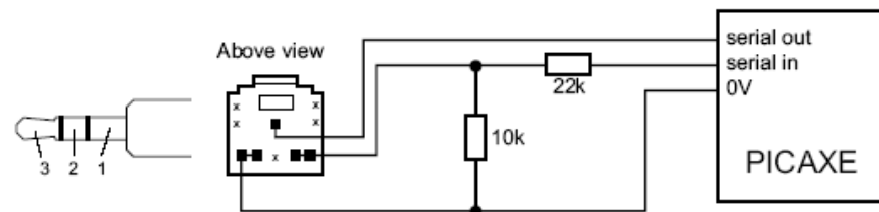
```
*****  
***** PRUEBA ENCENDIDO LED D1 (pin 7, out 0) *****  
*****  
  
inicio:  
  
    high 0          'enciende LED en pin 7  
    wait 1          'retardo 1 segundo  
    low 0           'apaga LED en pin 7  
    wait 1  
    goto inicio     'salto a inicio
```

The status bar at the bottom shows 'PICAXE-08 mode', 'PICAXE-08', 'COM 2', 'CAPS', 'NUM', 'INS', '09/11/05', and '23:54'.

El sistema PICAXE

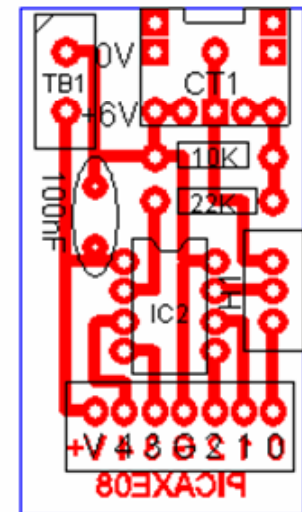
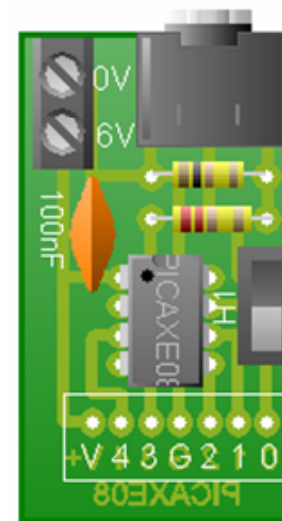
El sistema integra todas las fases de diseño de un sistema con microcontrolador:

- ❑ **Edición:** escritura del código del programa y la revisión de la sintaxis bien utilizando flujogramas o Basic.
- ❑ **Compilación:** conversión del formato del lenguaje de alto nivel a nivel de interpretación del micro en Hex.
- ❑ **Grabación:** mediante el software de comunicación integrado en el sistema y el **loader** cargado en el PIC es capaz de grabar los datos en su memoria con tan solo 3 hilos de conexión.



El sistema PICAXE

Al igual que en todos los sistemas de desarrollo, existen ya predefinidas toda una serie de tarjetas de prácticas sobre las cuales podemos emular las aplicaciones que diseñemos, pero gracias al firmware que poseen los microcontroladores PICAXE se puede construir la aplicación completa incluyendo al “microcontrolador”, y sobre la aplicación programarlo sin necesidad del sistema de desarrollo, ni del circuito programador de microcontroladores.





VENTAJAS DEL SISTEMA PICAXE

- Es un sistema de microcontroladores muy fácil de programar (nivel 4º ESO), dado que utiliza lenguaje BASIC o diagramas de flujo.
- No necesita programador de microcontroladores, utiliza el mismo software de programación y dos hilos de conexión con el PICAXE.
- Múltiples programaciones al utilizar memoria FLASH.
- Sistemas muy económicos.



VANTAJAS DEL SISTEMA PICAXE

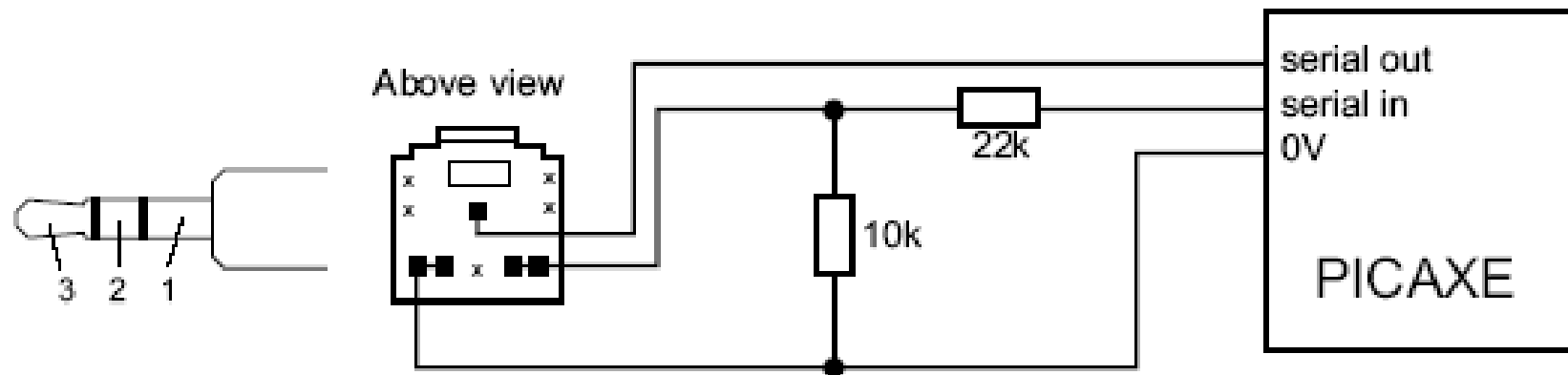
Está disponible en tres versiones de:

- ☐ 8 terminales (PICAXE-08)
- ☐ 18 terminales (PICAXE-18)
- ☐ 28 terminales (PICAXE-28)

En estos microcontroladores ya se tienen definidos los terminales que tienen la función de entrada y salida de datos, además de los terminales que sirven para programar el PICAXE, utilizando el software ***Programming Editor con interface*** en múltiples idiomas, de uso libre, para fines educativos.

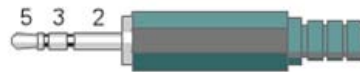
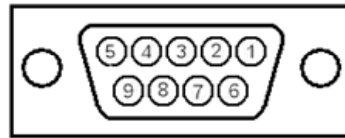
Características PICAXE

Fácil sistema de programación:



Características PICAXE

Cable programación PICAXE (RS232 → Jack):





Características PICAXE

Cable programación PICAXE (USB → Jack):

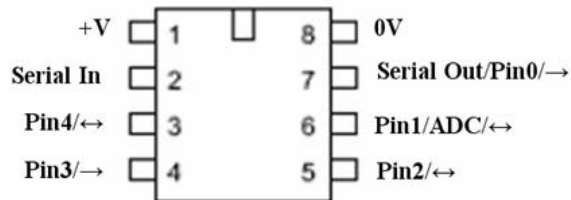


SISTEMA PICAXE-08

Características PICAXE08:

Tipo de PICAXE	Nº Pines	Memoria (lineas)	Pines	Salidas	Entradas	A/D (b-baja)	Memoria Datos	Interrupciones
PICAXE-08	8	40	5	1-4	1-4	1b	128-prog	-

NOTA:



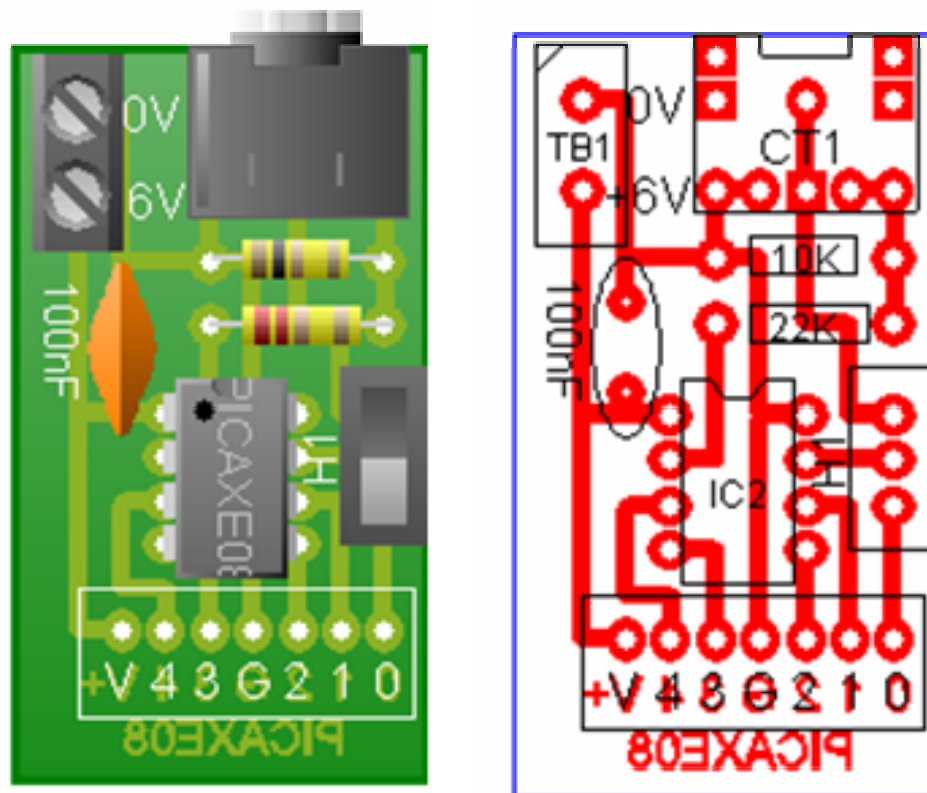
El número de líneas de memoria de programa es solo aproximado, ya que diferentes comandos requieren diferentes cantidades de memoria. El pin3 es solo de entrada y el pin0 es solo de salida. La alimentación de 3V a 5V. Se puede alimentar con 6V (4x1,5).

El conector H1 tiene un jumper que permite la programación del PICAXE-08 (jumper en parte superior) o bien poner el pin 0 como salida (jumper en parte inferior, tal y como figura en la imagen).

128 bytes de memoria Datos
40 líneas de código
1 Conversor A/D

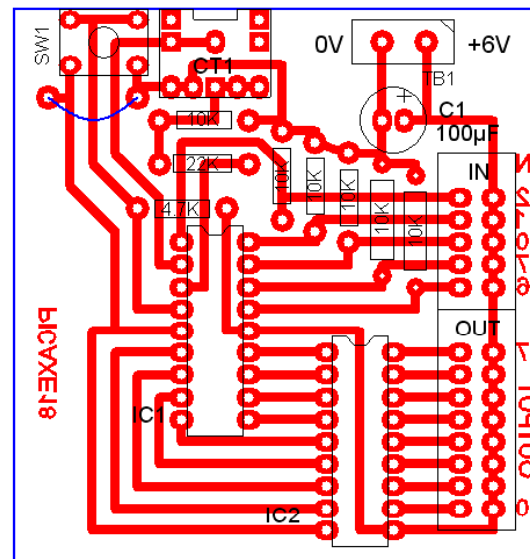
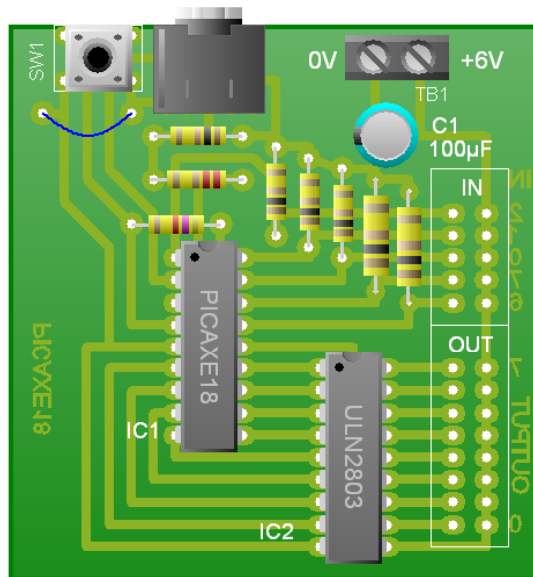
SISTEMA PICAXE-08

- Tarjeta de control con PICAXE08:



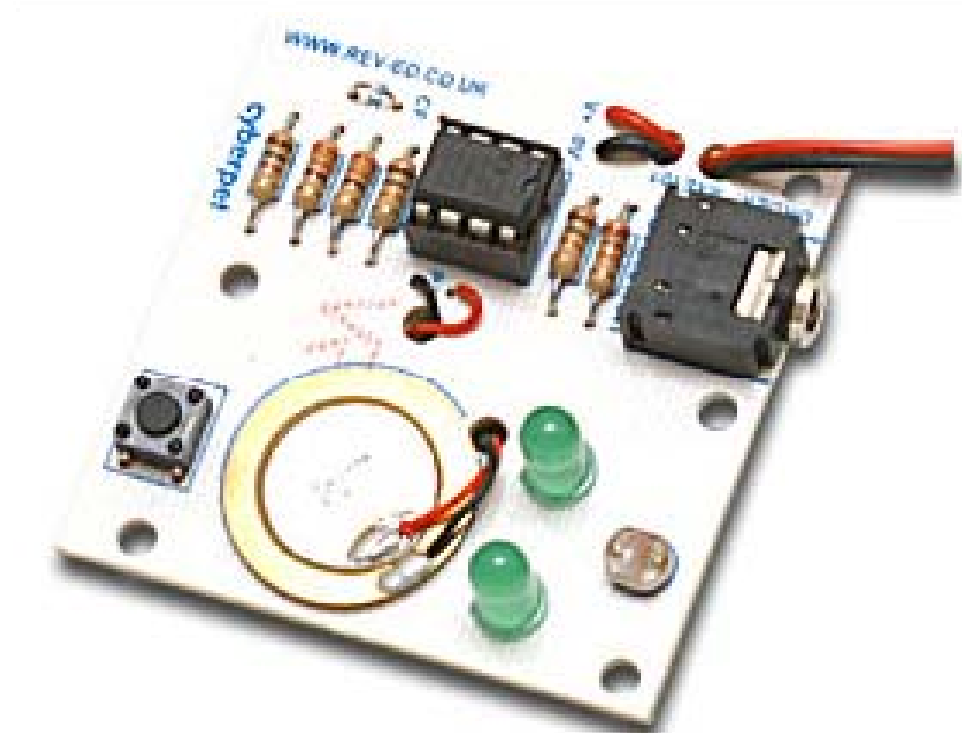
SISTEMA PICAXE-18

- Tarjeta de control con PICAXE18:



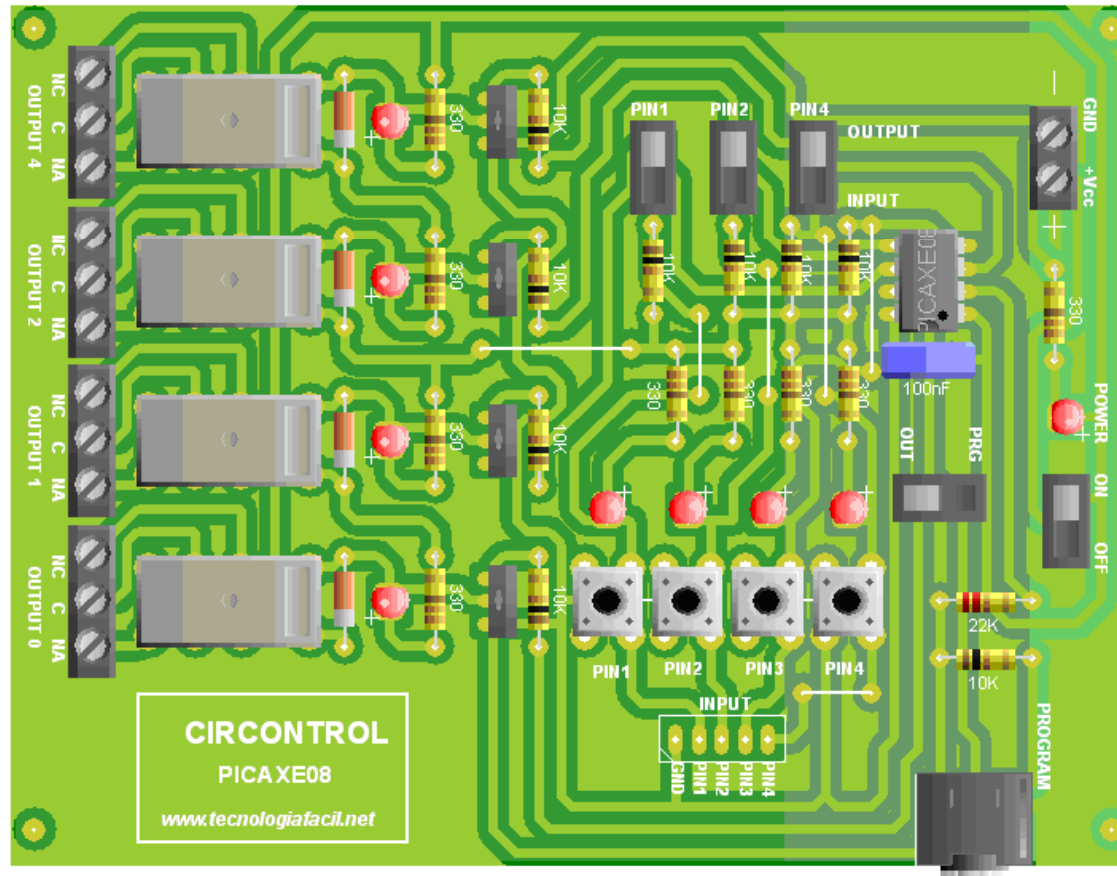
SISTEMA PICAXE-08

■ Cyberpet:



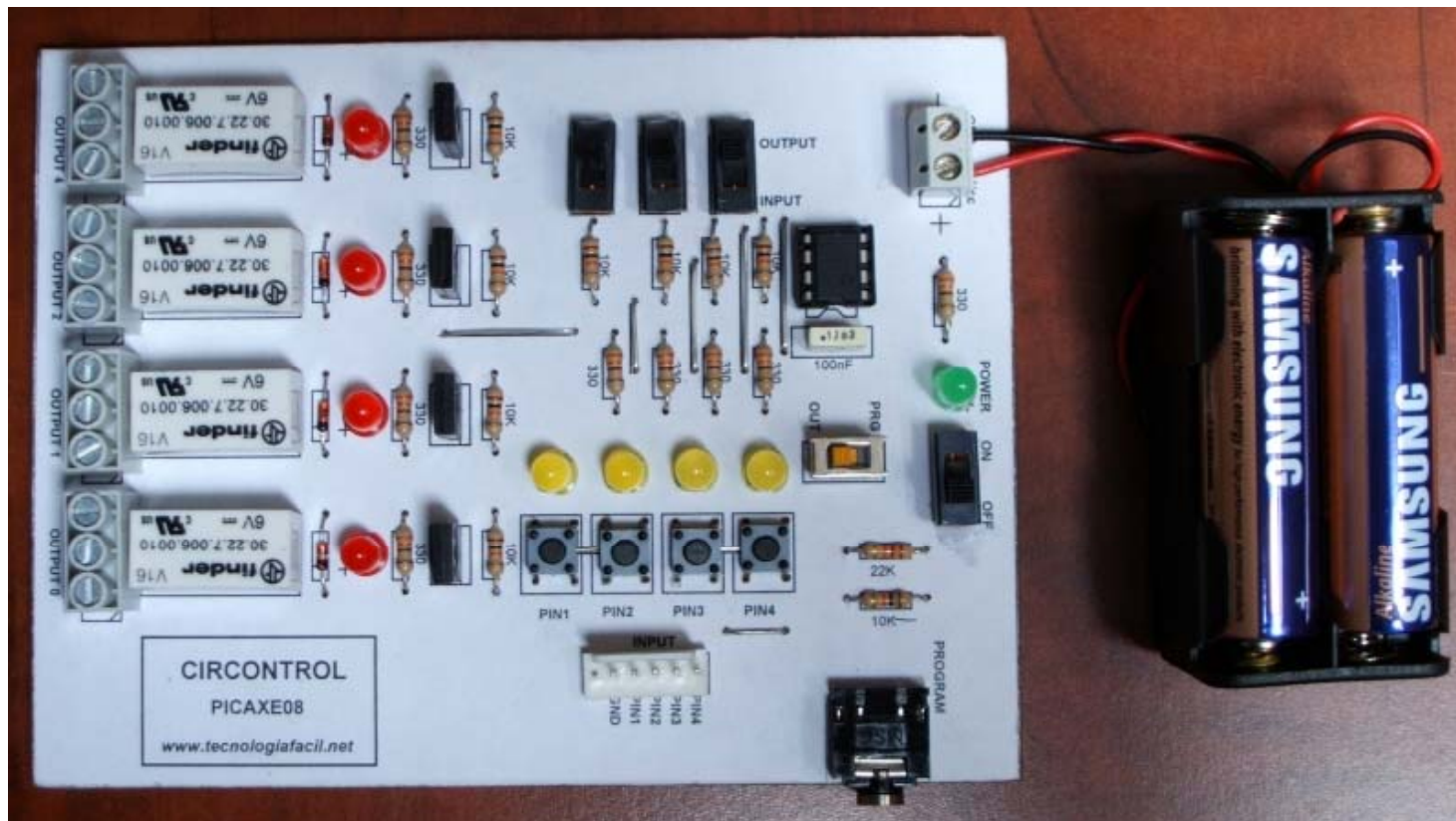
SISTEMA PICAXE-08

- Tarjeta de control y experimentación CIRCONTROL



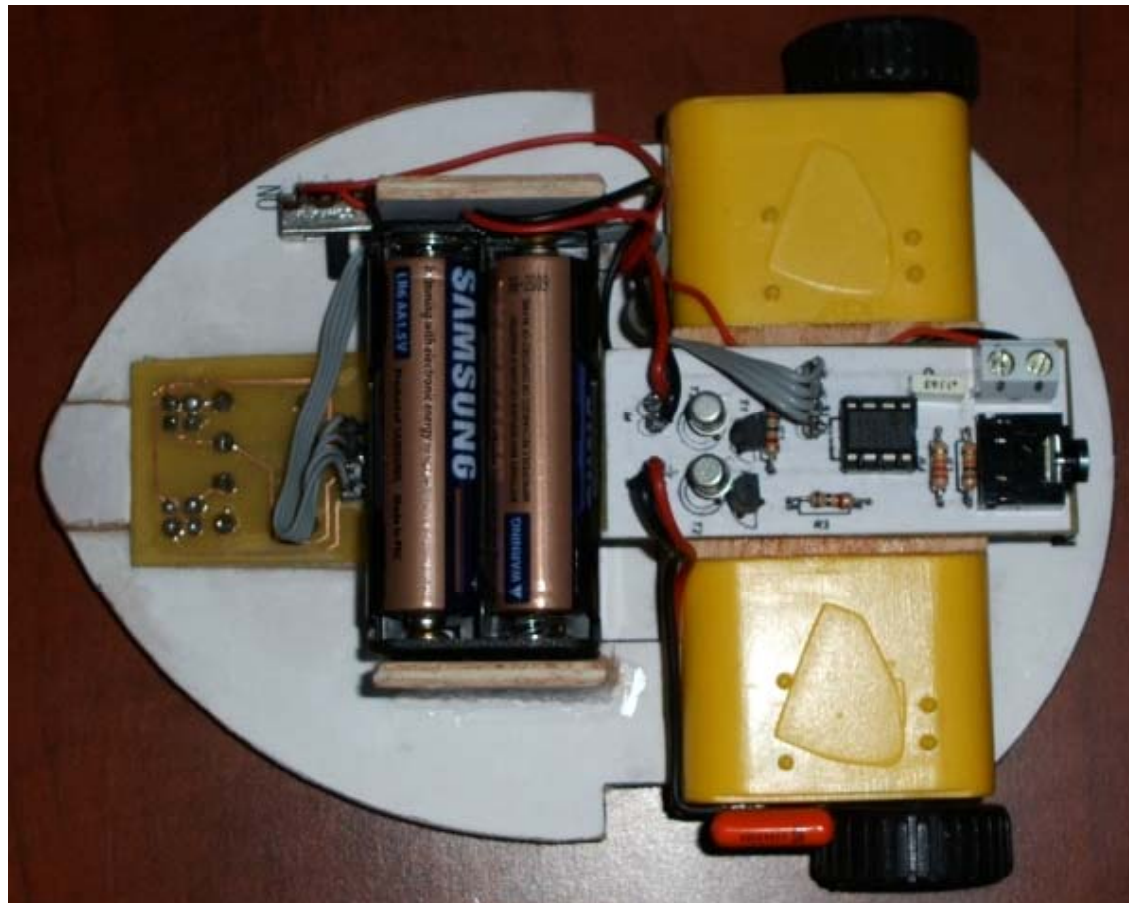
SISTEMA PICAXE-08

- Tarjeta de control y experimentación **CIRCONTROL**



PROYECTOS

■ Robot siguelíneas



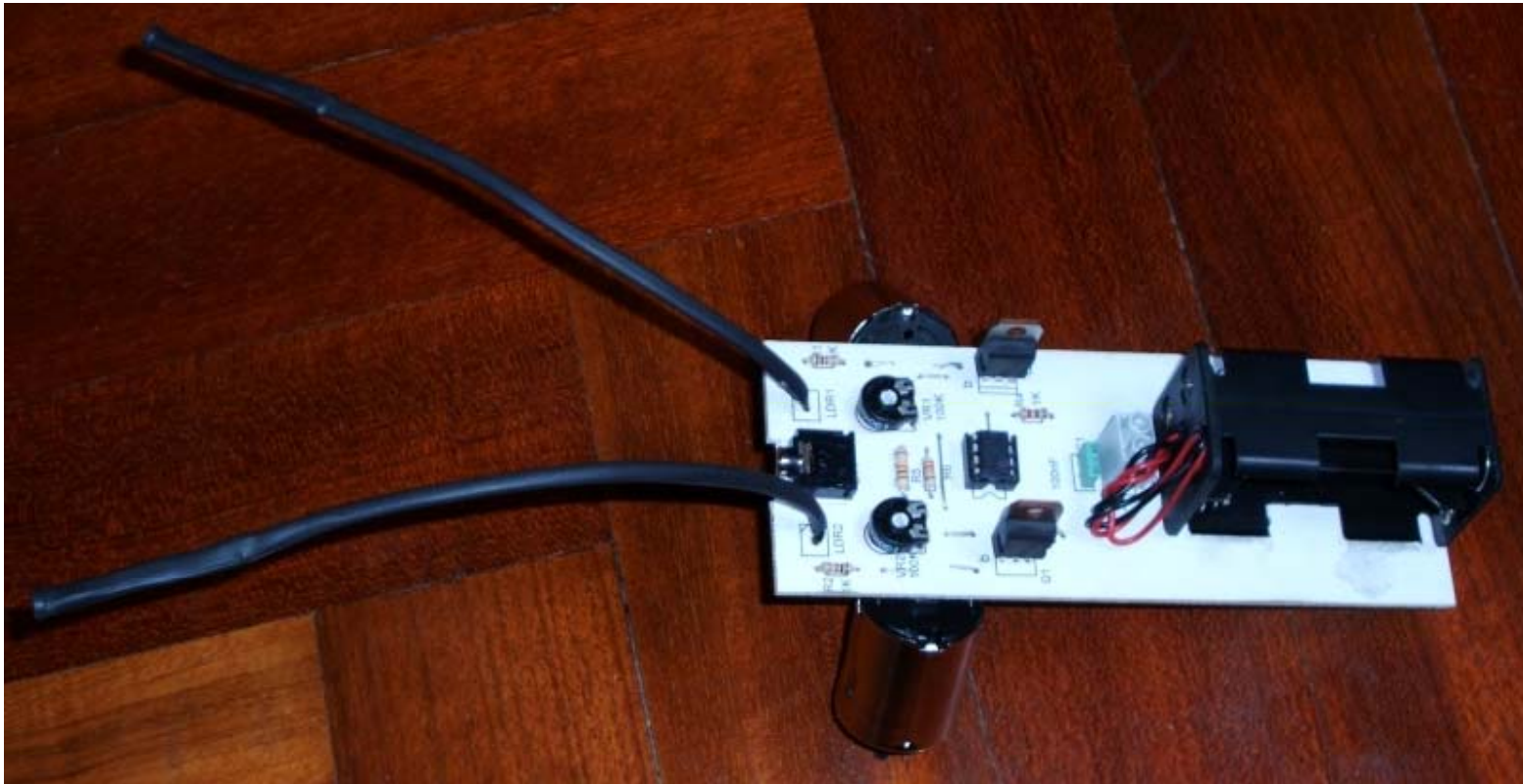


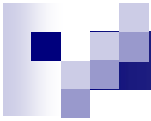
PROYECTOS

```
|'*****  
|***** ROBOT Siguelineas PICAKE-08 *****  
|*****  
'Sensor CNY70 input pin 3, patilla 4  
'Sensor CNY70 input pin 4, patilla 3  
'Output pin1, patilla 6 hacia Darlington control M Izqda  
'Output pin2, patilla 5 hacia Darlington control M Dcha  
  
'**** Bucle principal ****  
  
inicio:  
    input 3  
    input 4  
    if pin4 = 0 and pin3 = 0 then recto  
    if pin3 = 0 and pin4 = 1 then giraderecha  
    if pin3 = 1 and pin4 = 0 then giraizquierda  
    if pin4 = 1 and pin3 = 1 then rastrea  
    goto inicio  
  
'**** Gira derecha ****  
  
giraderecha:  
    high 2  
    low 1  
    goto inicio  
  
'**** Gira izquierda ****  
  
giraizquierda:  
    high 1  
    low 2  
    goto inicio  
  
'**** Recto ****  
  
recto:  
    high 2  
    high 1  
    goto inicio  
  
'**** Rastrea ****  
  
rastrea:  
    low 2  
    pulsout 1, 700  
    pause 20  
    low 1  
    pulsout 2, 700  
    pause 20  
    goto inicio
```

PROYECTOS

■ Robot Cokie Luz:





PROYECTOS

```
*****  
*****Cokie Luz*****  
*****Robot seguidor de luz*****  
*****
```

principal:

```
low 1  
low 2  
if pin3 = 0 then dcha  
if pin4 = 0 then izda  
goto principal
```

dcha:

```
high 2  
pause 100  
low 2  
goto principal
```

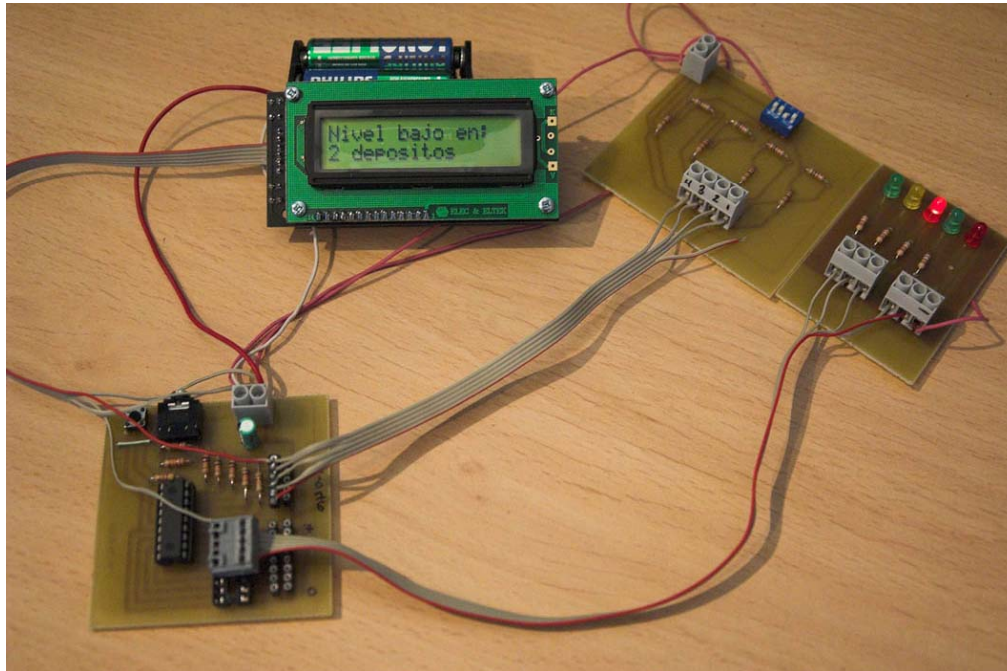
izda:

```
high 1  
pause 100  
low 1  
goto principal
```



PROYECTOS

■ Control depósitos:



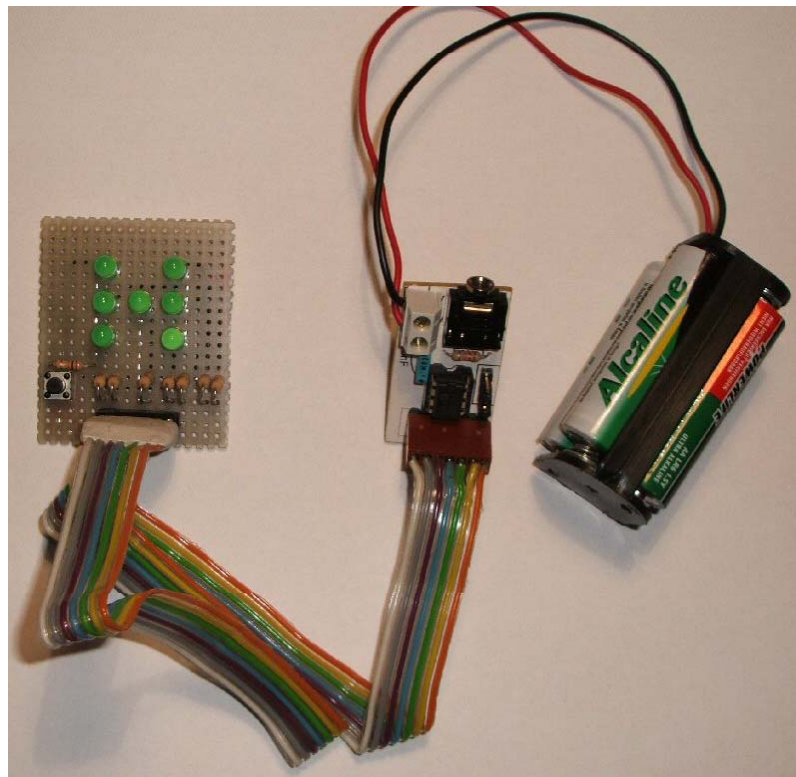
PROYECTOS

- Control de un depósito de agua:



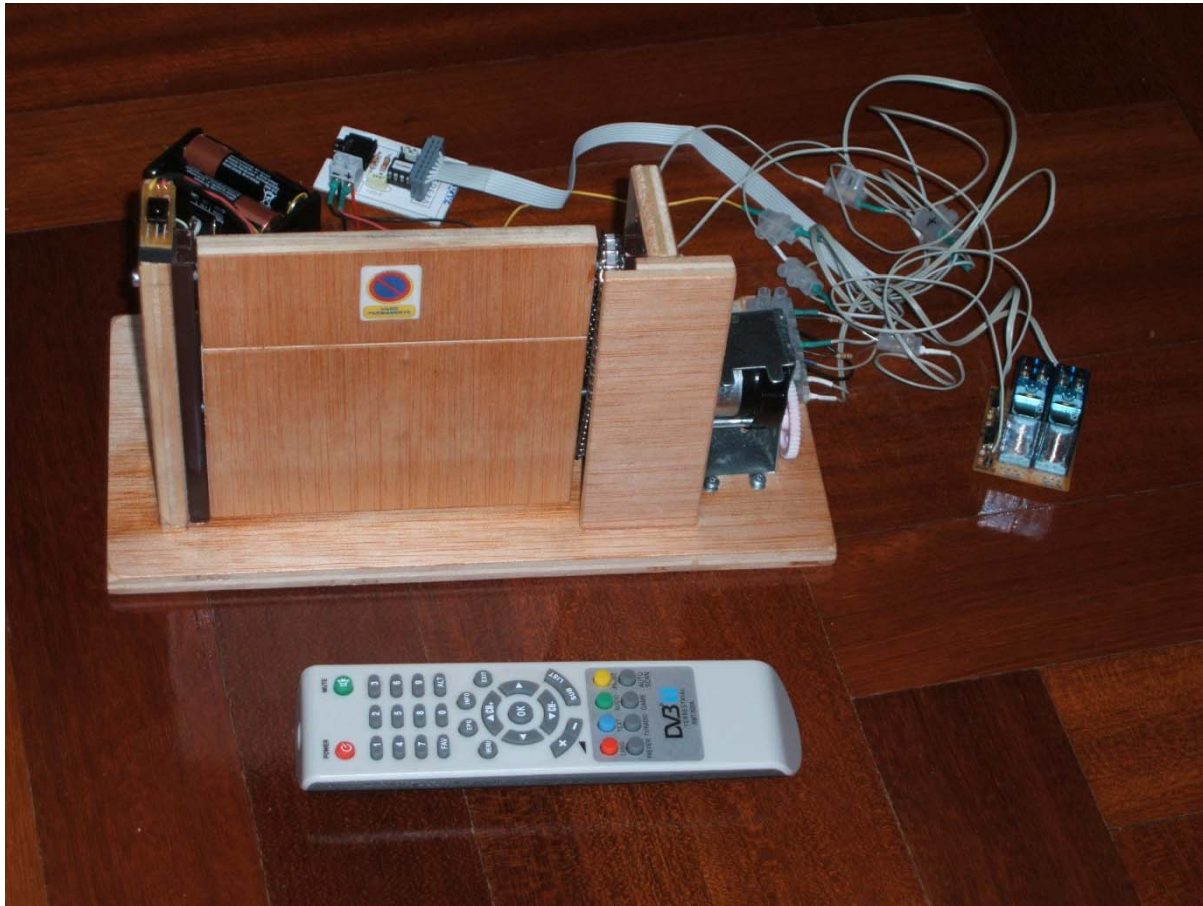
PROYECTOS

■ Dado con Leds:



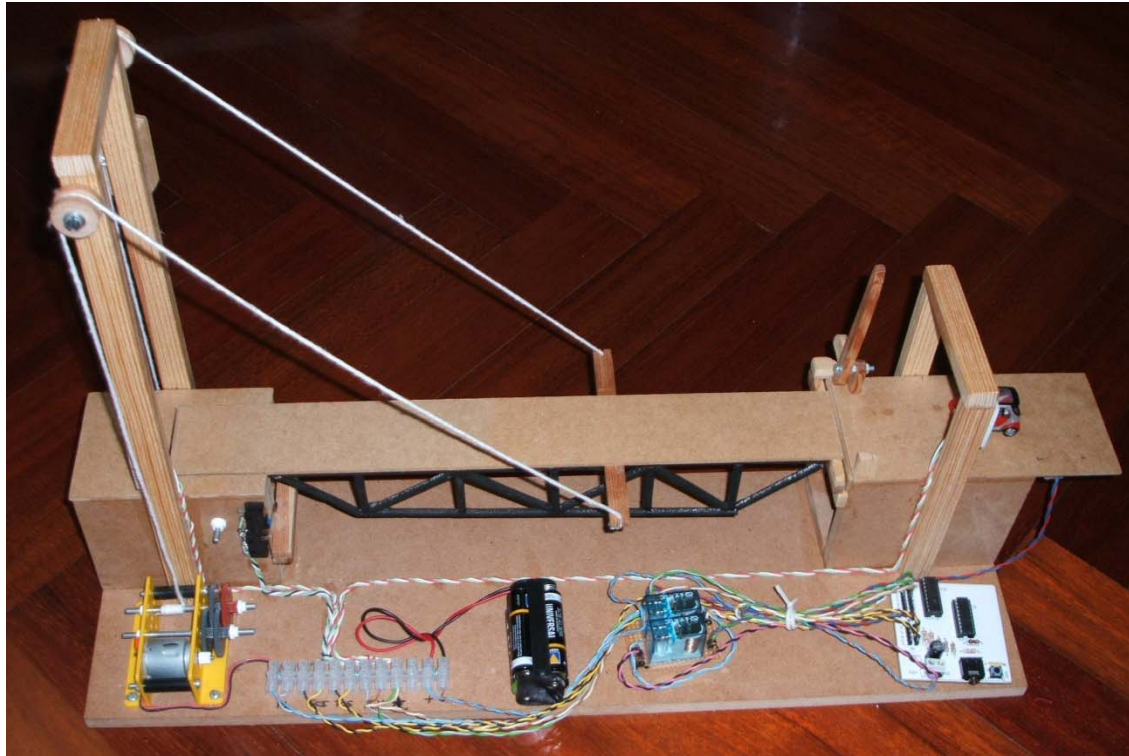
PROYECTOS

- Puerta de garaje con mando a distancia por IR:



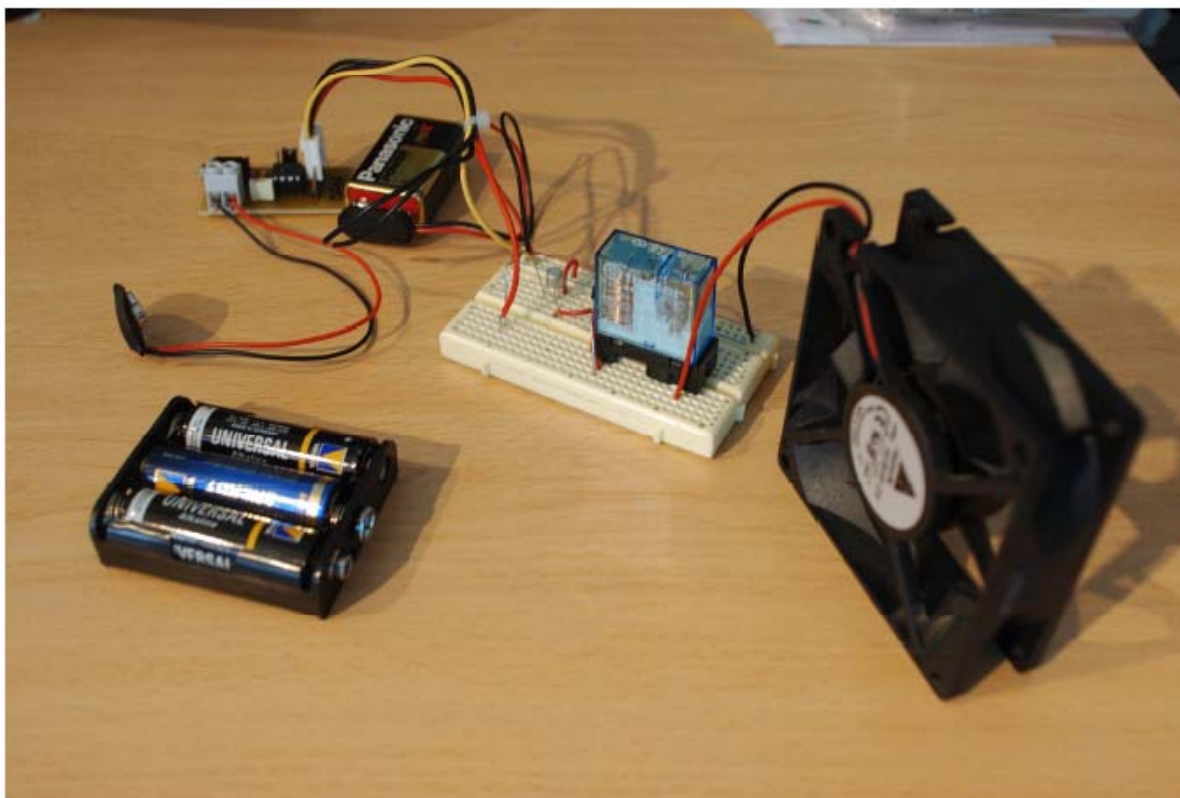
PROYECTOS

■ Puente levadizo:



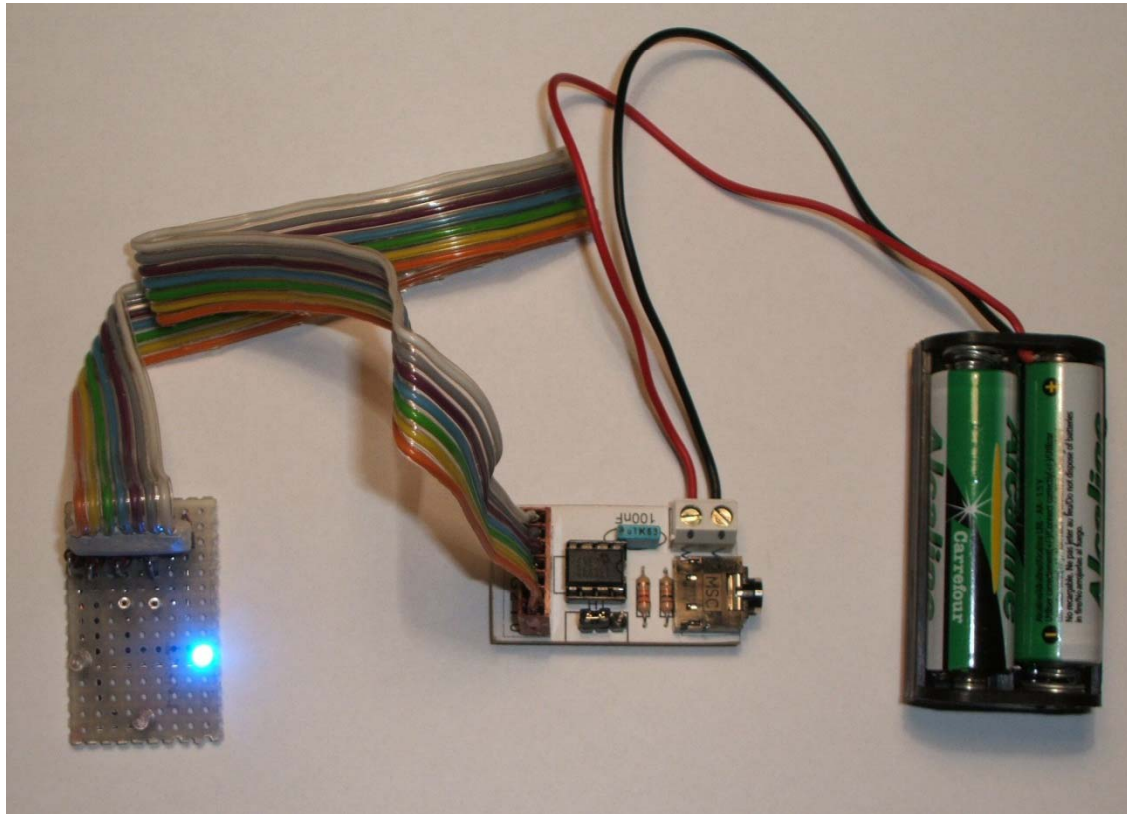
PROYECTOS

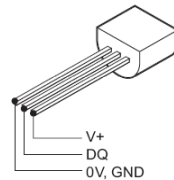
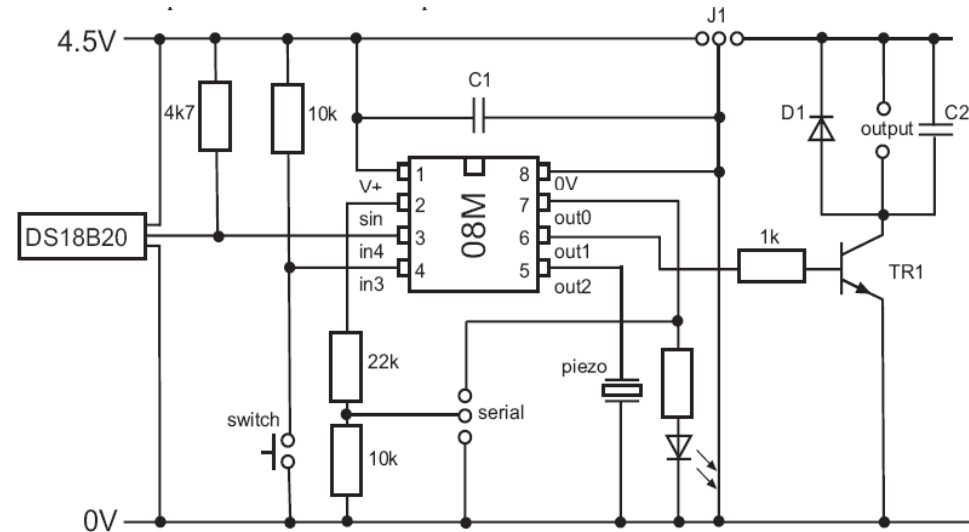
■ Control ventilador:



PROYECTOS

■ Triángulo emergencias:

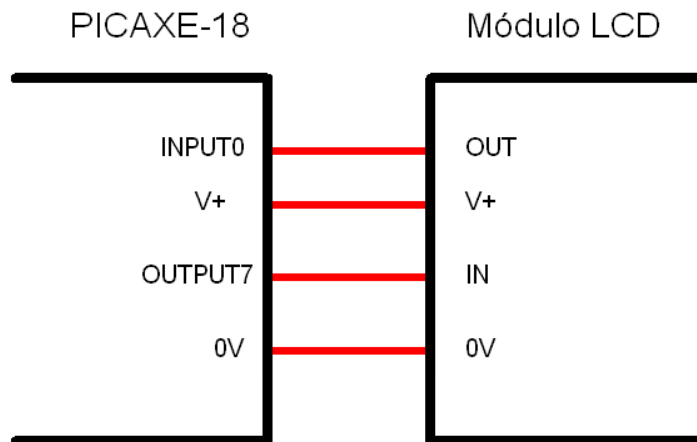




El **DS18B20** es un sensor digital de temperatura con un rango de -55°C a $+125^{\circ}\text{C}$. Tiene la ventaja de utilizar únicamente 3 hilos de conexión y produce una salida digital de alta precisión con una resolución $0,5^{\circ}\text{C}$, sin necesidad de utilizar un convertidor analógico-digital (ADC).

PROYECTOS

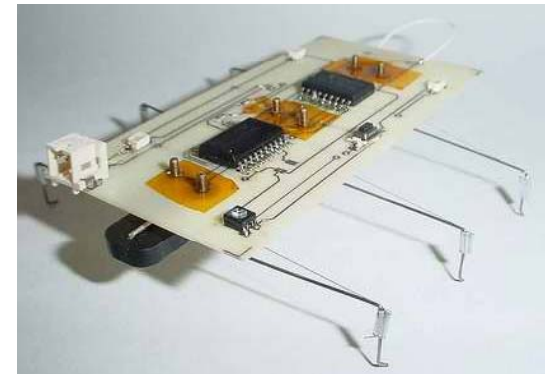
■ Control LCD:



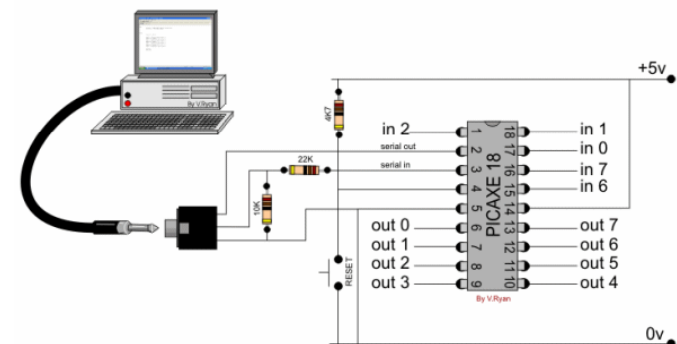
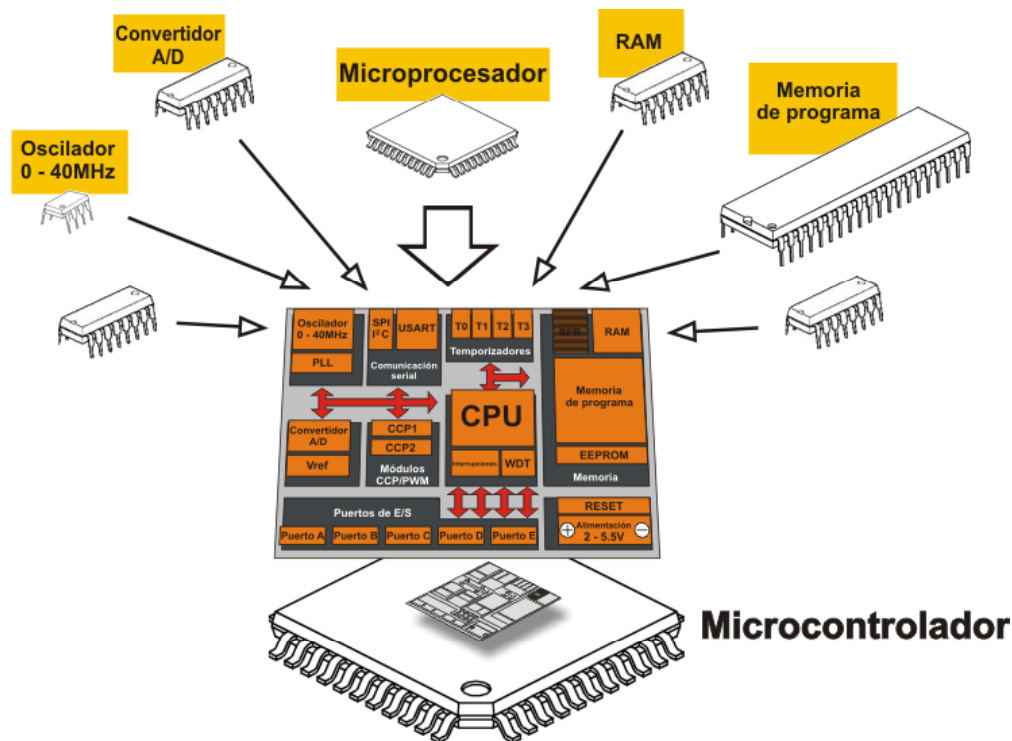
PROYECTOS

■ Próximos proyectos:

- Robot mouse con PICAXE08
- Robot hexópodo con músculos de alambre (nitinol)
- Control invernadero con PICAXE
- ...



Vamos a practicar un poco...



Cyberpet

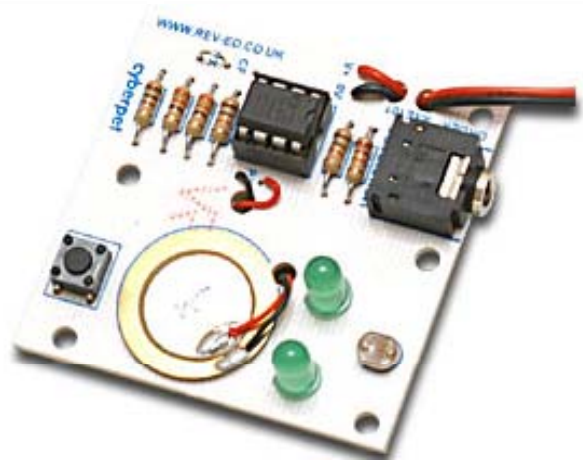


Diagrama de Bloque

El diagrama de bloque para su ciber-mascota puede verse así:

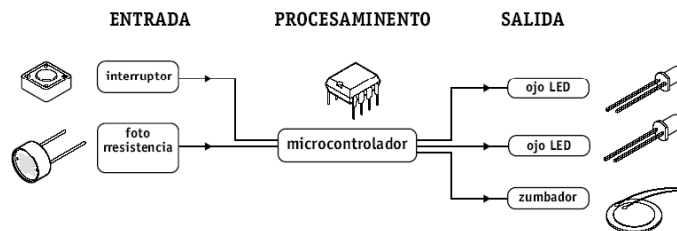
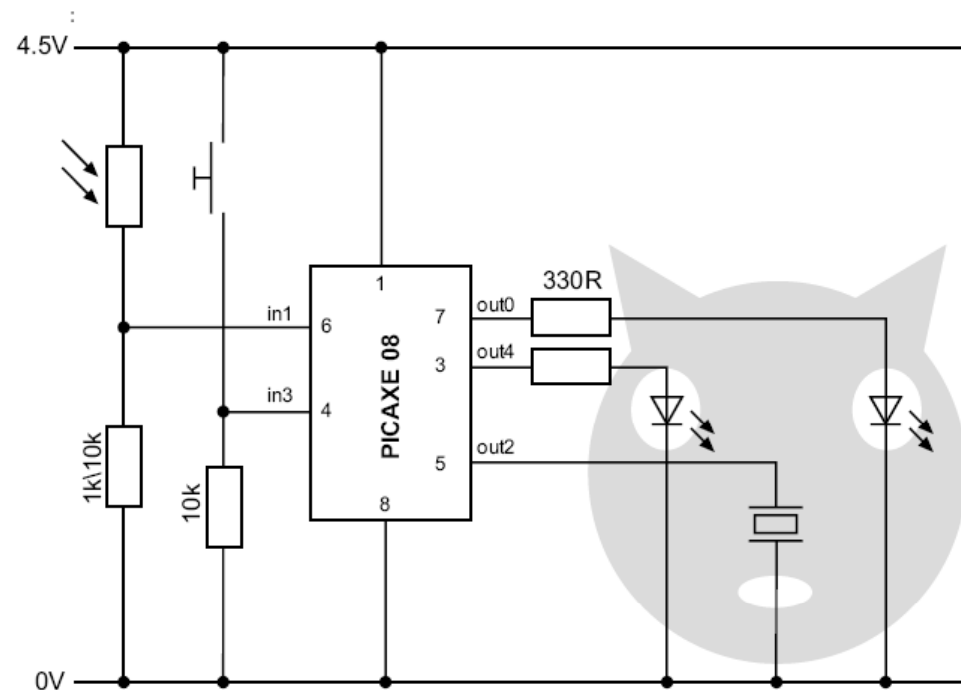


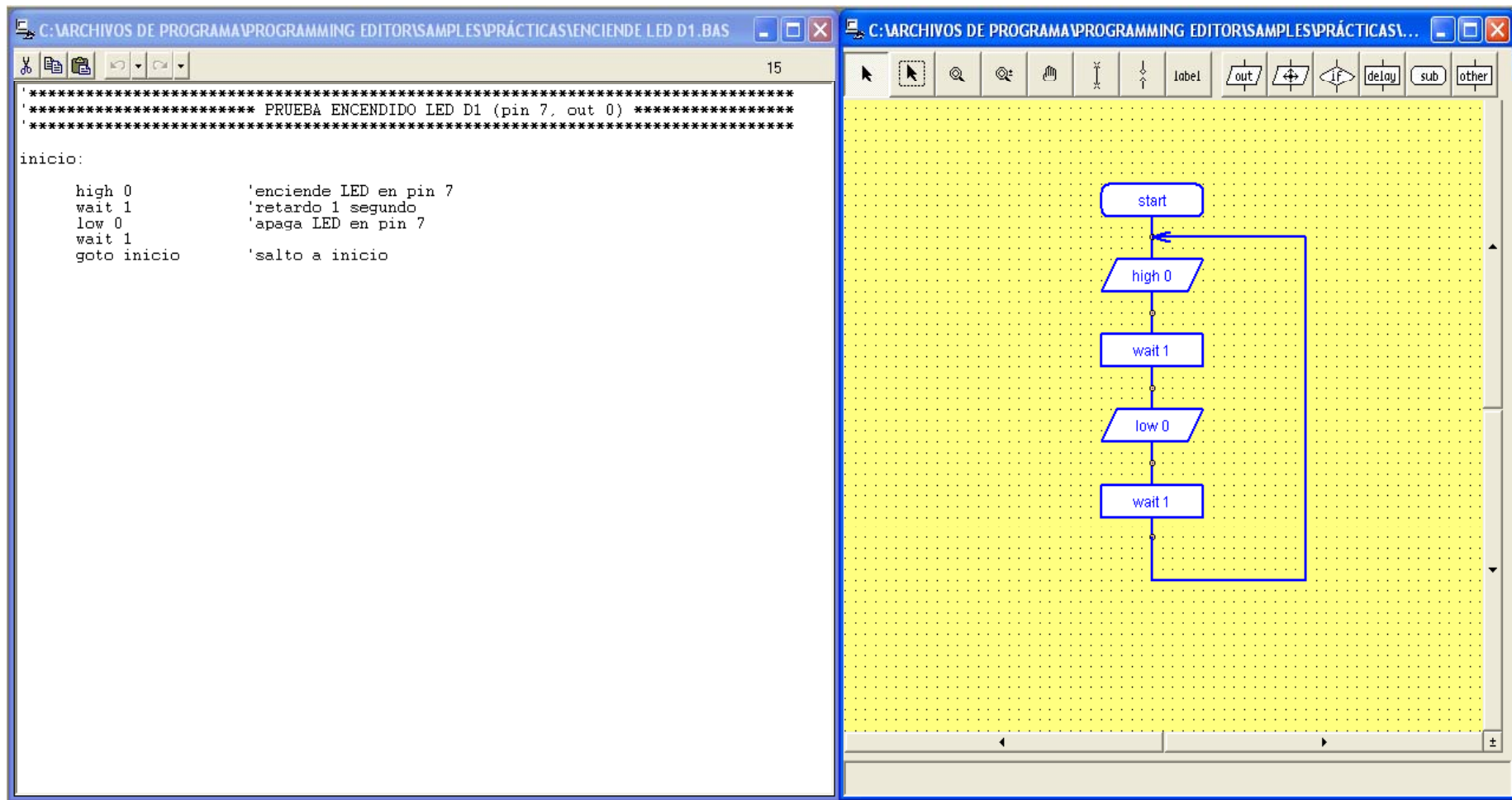
Diagrama del Circuito

A continuación se muestra el diagrama de circuito para el proyecto de la ciber-mascota



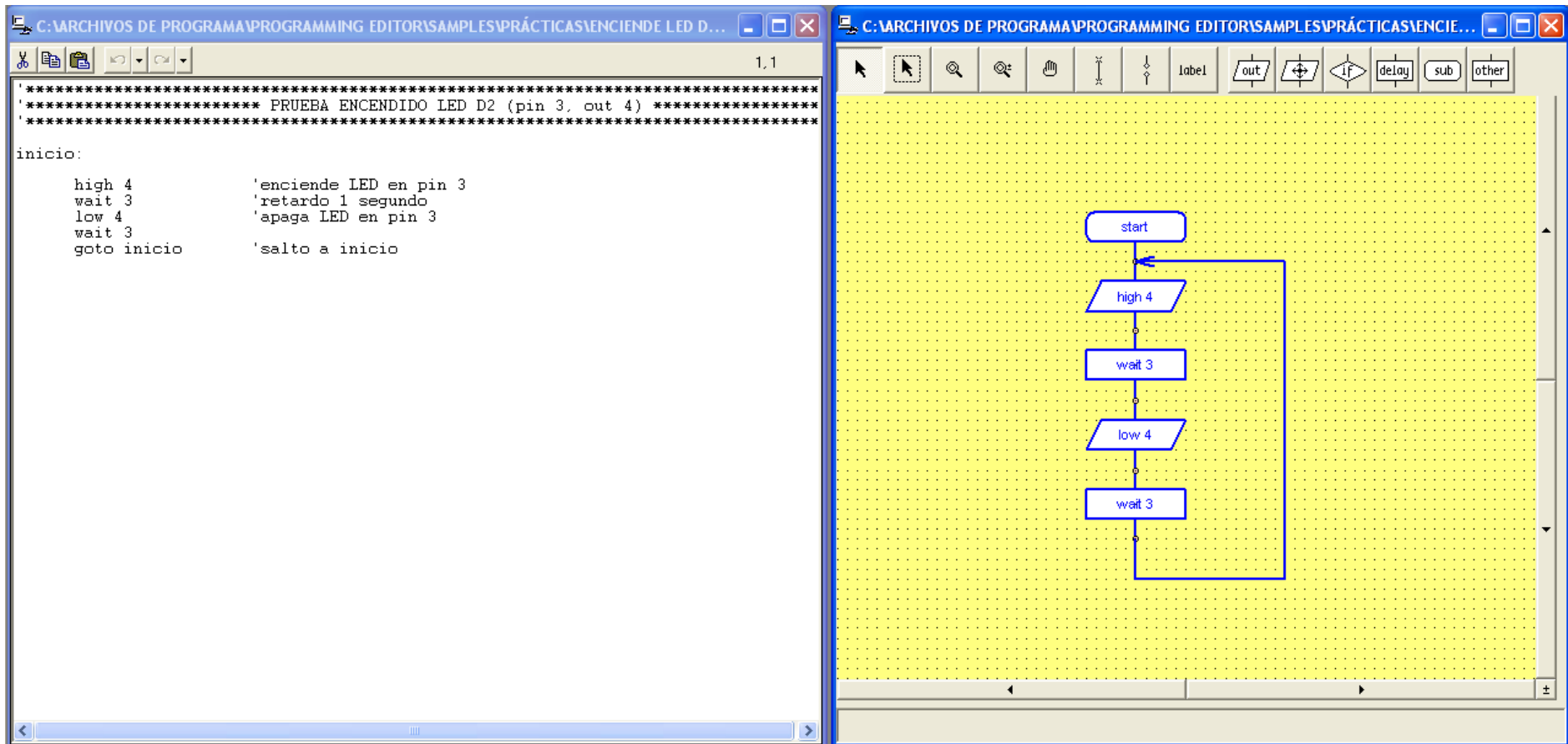
PRÁCTICAS

- Encendido y apagado de forma intermitente de un led conectado al pin 7 del PICAXE-08 (salida: PIN0)



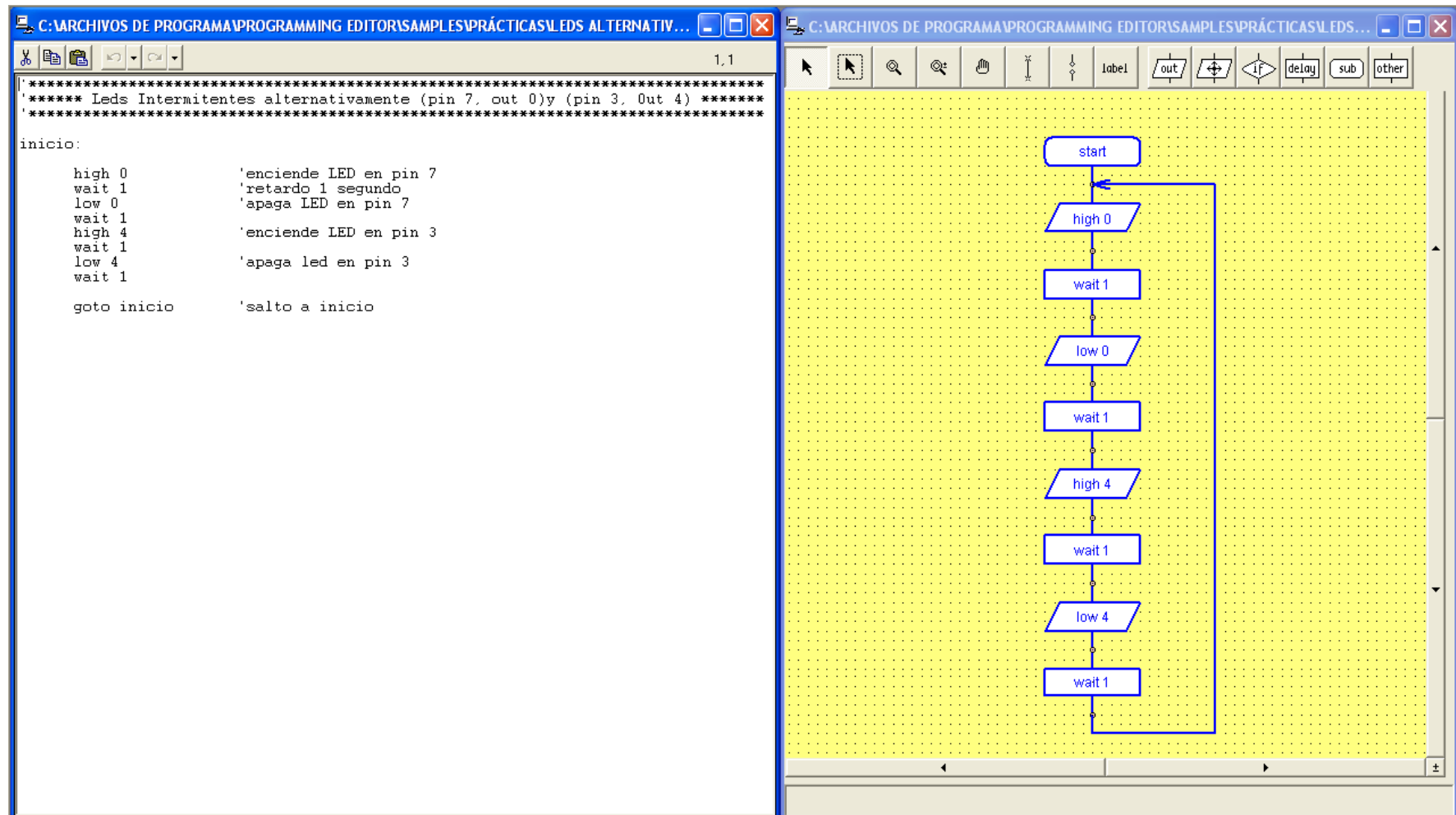
PRÁCTICAS

- Encendido y apagado de forma intermitente de un led conectado al pin 3 del PICAXE-08 (salida: PIN4)



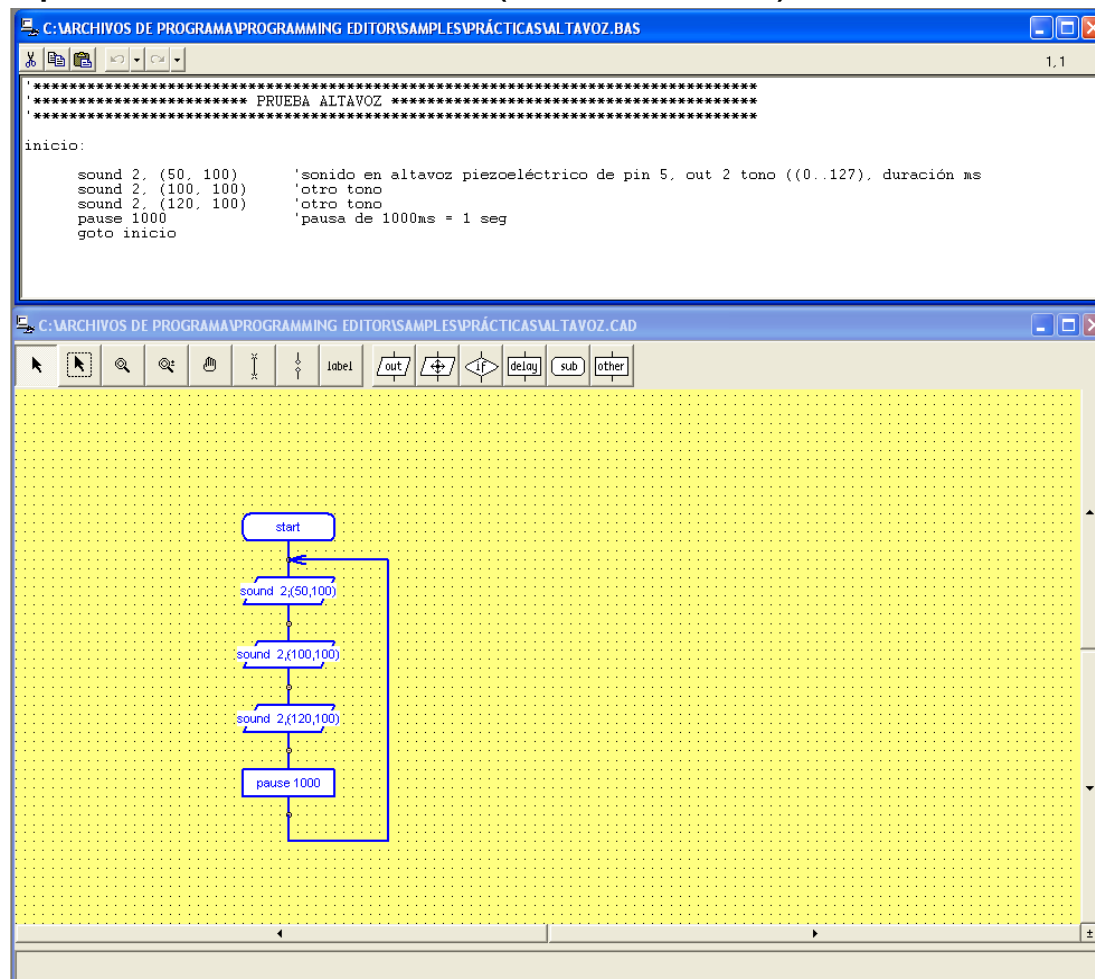
PRÁCTICAS

- Encendido y apagado alternativo de dos leds conectados a los pin 3 y pin 7 del PICAXE-08 /Salidas: PIN4 e PIN0)



PRÁCTICAS

- Generación de 3 tonos diferentes a través del altavoz piezoeléctrico conectado al pin 5 del PICAXE-08 (Salida: PIN2):



PRÁCTICAS

- Si se activa el interruptor conectado al pin 4 (Entrada: PIN-) enciende el led conectado al pin 7 del PICAXE-08 (Salida: PIN0):

The screenshot displays the PICAXE programming editor interface. The left pane shows the BASIC code for a program named 'PRUEBA INTERRUPTOR'. The code includes a loop that checks if input pin 3 is high; if so, it turns on an LED on pin 7, waits for 2 seconds, and then turns the LED off before looping back to the start.

```
***** PRUEBA INTERRUPTOR *****  
*****  
inicio:  
    if input3 is on then led          'si la entrada 3 es "1" salta a subrutina led sino a inicio  
    goto inicio                     'salto a inicio  
led:  
    high 0                          'enciende LED pin 7  
    wait 2                          'encendido durante un retardo de 2 segundos  
    low 0                           'apagado LED pin 7  
    goto inicio
```

The right pane shows the flowchart for the same program. It starts at a 'start' terminal, leading to a decision diamond 'pin3=1'. If the condition is true (Y), it proceeds through a sequence of blocks: 'high 0', 'wait 2', and 'low 0', before looping back to the decision diamond. If the condition is false (N), it also loops back to the decision diamond.

```
graph TD  
    Start([start]) --> Decision{pin3=1}  
    Decision -- Y --> High0[high 0]  
    High0 --> Wait2[wait 2]  
    Wait2 --> Low0[low 0]  
    Low0 --> Decision  
    Decision -- N --> Decision
```

PRÁCTICAS

- Tenemos conectada una LDR en el pin 6 (Entrada: PIN1) del PICAXE-08. Visualizaremos el valor que nos da el ADC a través del Debug de la aplicación (presenta el valor de las variables b0..b13, tanto en valor decimal como digital).

```
C:\ARCHIVOS DE PROGRAMA\PROGRAMMING EDITOR\SAMPLES\PRÁCTICAS\UMBRAL.BAS
1.1

*****
***** Chequeo de umbral de luz *****
*****
'LDR wn pin 6, input1

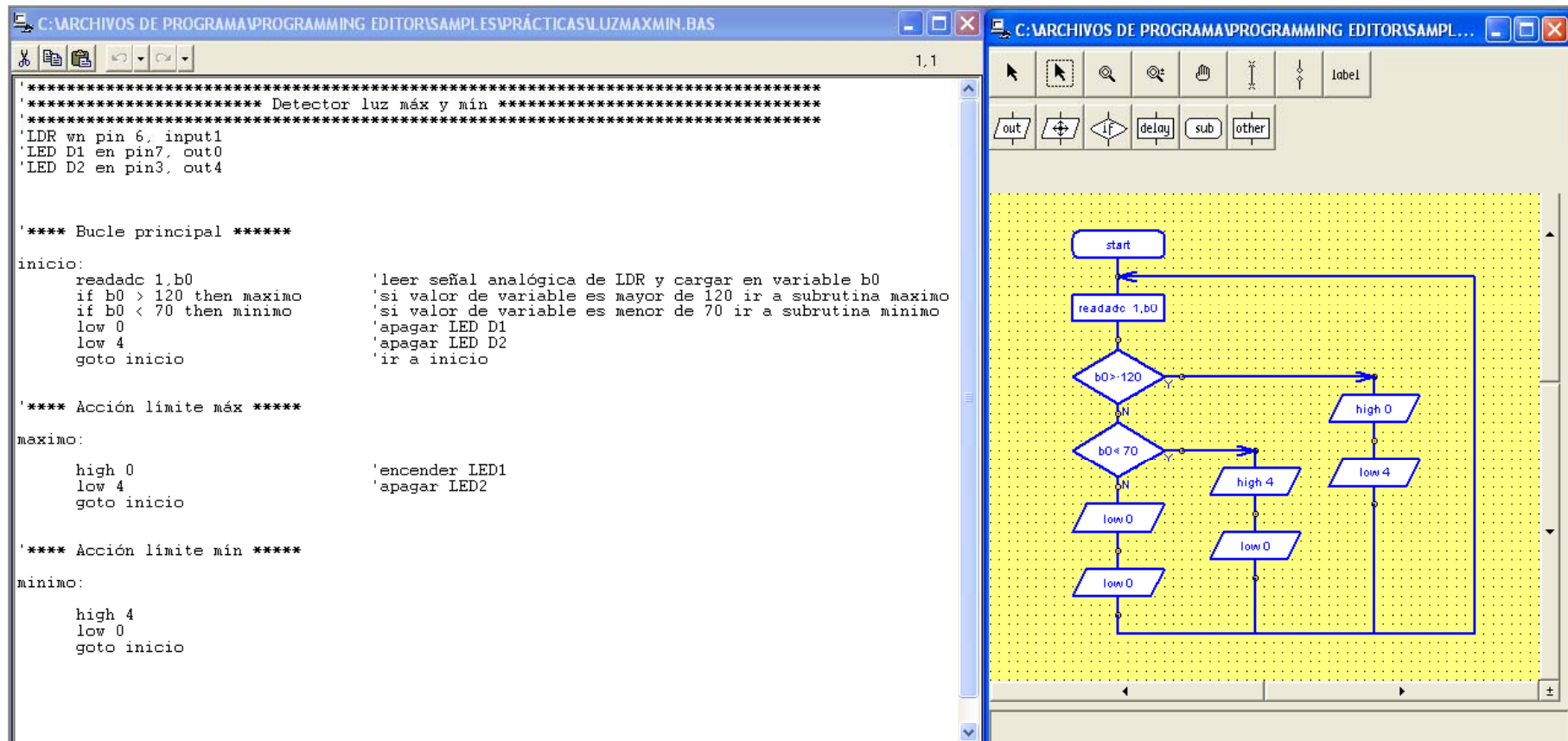
**** Bucle principal ****

inicio:
  readadc 1,b0      'lee la señal analógica de LDR y carga en variable b0
  debug b0          'transmite el valor b0 a la pantalla del PC
  pause 100         'pausa
  goto inicio       'saltar a inicio
```

Debug - (121)				
pins =	0	\$00	%00000000	---
dirs =	0	\$00	%00000000	---
b0 =	11	\$0B	%00001011	
b1 =	0	\$00	%00000000	---
b2 =	0	\$00	%00000000	---
b3 =	0	\$00	%00000000	---
b4 =	0	\$00	%00000000	---
b5 =	0	\$00	%00000000	---
b6 =	0	\$00	%00000000	---
b7 =	0	\$00	%00000000	---
b8 =	0	\$00	%00000000	---
b9 =	0	\$00	%00000000	---
b10 =	0	\$00	%00000000	---
b11 =	0	\$00	%00000000	---
b12 =	0	\$00	%00000000	---
b13 =	0	\$00	%00000000	---
Modo de Visualización				
☒ Byte ☐ Word				
Cerrar				

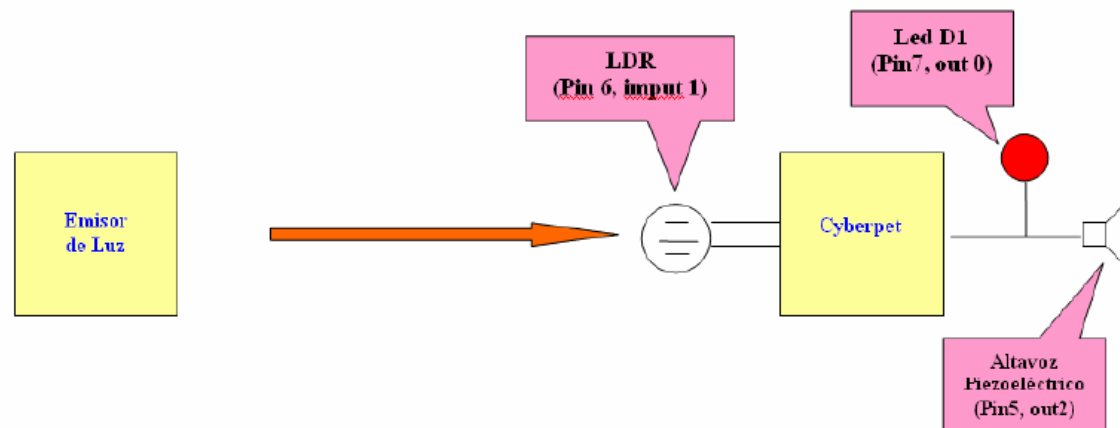
PRÁCTICAS

- Tenemos conectada una LDR en el pin 6 (Entrada: PIN1) del PICAXE-08. Si la intensidad de luz es alta (>120) se iluminará el Led 1, por el contrario, si la luz que incide sobre la LDR es baja (<70) se iluminará el Led 2.



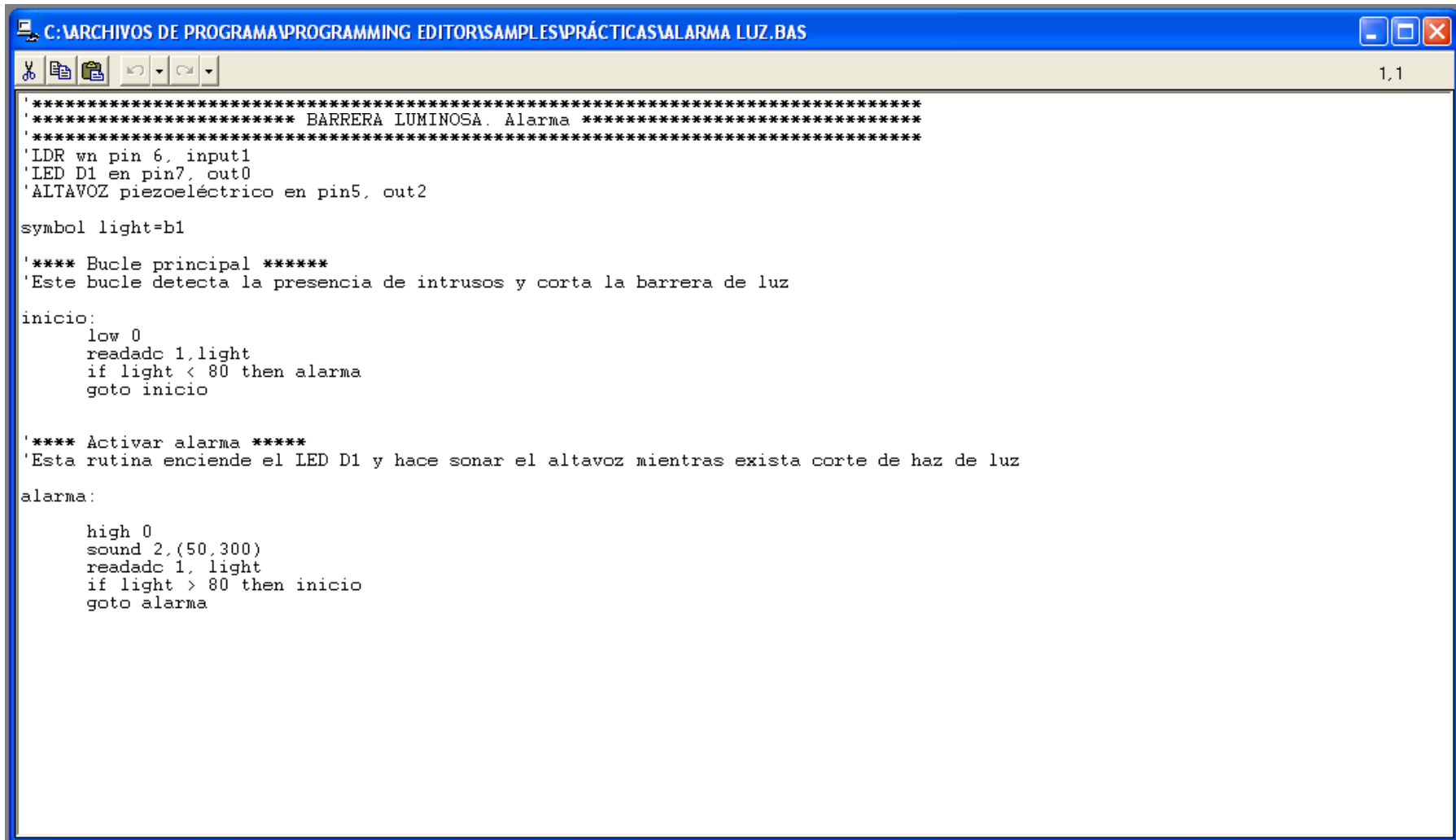
PRÁCTICAS

■ Tenemos conectada una LDR en el pin 6 (Entrada: PIN1) del PICAXE-08. Debemos generar un haz de luz unidireccional que deberá incidir sobre la superficie sensible de la LDR. Si la luz incide, no se disparará la alarma (Led y piezo activados), por el contrario si se corta el haz de luz (atravesado por un objeto móvil) esta se disparará activando una señal acústica y luminosa (altavoz piezoeléctrico y Led).



PRÁCTICAS

■ Programa “Barrera de luz”:



```
C:\ARCHIVOS DE PROGRAMA\PROGRAMMING EDITOR\SAMPLES\PRÁCTICAS\ALARMA LUZ.BAS
1,1

'*****
'***** BARRERA LUMINOSA. Alarma *****
'*****
'LDR wn pin 6, input1
'LED D1 en pin7, out0
'ALTAVOZ piezoeléctrico en pin5, out2

symbol light=b1

'**** Bucle principal ****
'Este bucle detecta la presencia de intrusos y corta la barrera de luz

inicio:
    low 0
    readadc 1,light
    if light < 80 then alarma
    goto inicio

'**** Activar alarma ****
'Esta rutina enciende el LED D1 y hace sonar el altavoz mientras exista corte de haz de luz

alarma:
    high 0
    sound 2,(50,300)
    readadc 1, light
    if light > 80 then inicio
    goto alarma
```

GRACIAS POR LA ATENCIÓN PRESTADA

