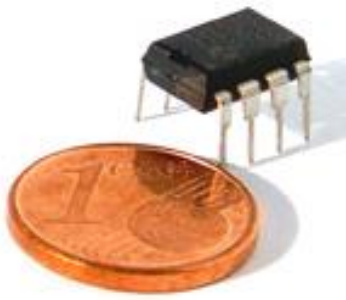


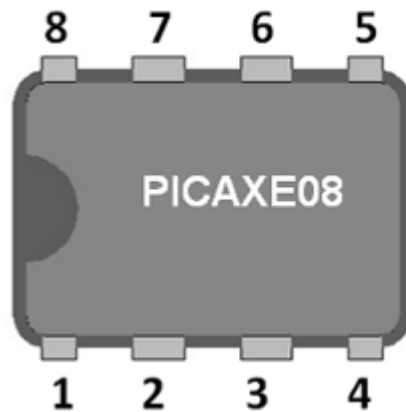
CONSTRUCCIÓN DE UN ENTRENADOR PICAXE 08 SOBRE CIRCUITO IMPRESO



Características del Picaxe08

El microcontrolador Picaxe08 es el más pequeño de la familia y aunque tiene poca memoria para almacenar líneas de programa, su simplicidad permite incluso realizar un entrenador en una placa de prototipos. Las características de este circuito son las siguientes:

Nº de pines	Memoria (líneas de programa)	Pines activos	Salidas	Entradas	Memoria Datos	Interrupciones
8	40	5	1-4	1-4	128	-



1. Alimentación (+)
2. Entrada serie
3. Pin 4
4. Pin 3
5. Pin 2
6. Pin 1
7. Pin 0/Salida serie
8. Masa

NOTA:

El pin 3 es de sólo entrada y el pin 0 de sólo salida.

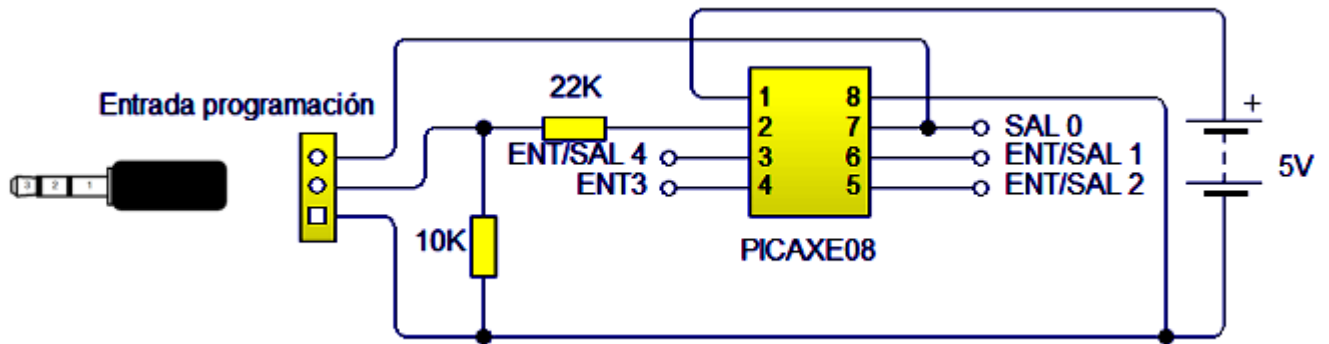
Se puede alimentar el circuito con una tensión comprendida entre 3 y 6 v.

El pin 0 se utiliza para la descarga del programa y como salida. Por este motivo, es necesario colocar un conmutador o un jumper para seleccionar la función del pin.

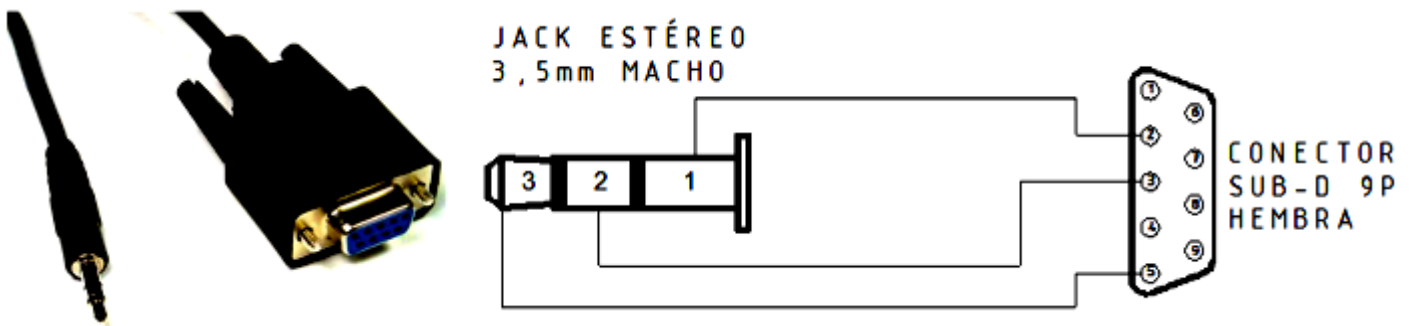
Este circuito puede suministrar unos 25 mA por salida, por lo que sólo se deben conectar directamente pequeñas cargas (led, altavoz etc...). Para conectar cargas que consuman corrientes superiores a 25 mA será necesario intercalar un transistor o circuito similar.



Circuito básico:

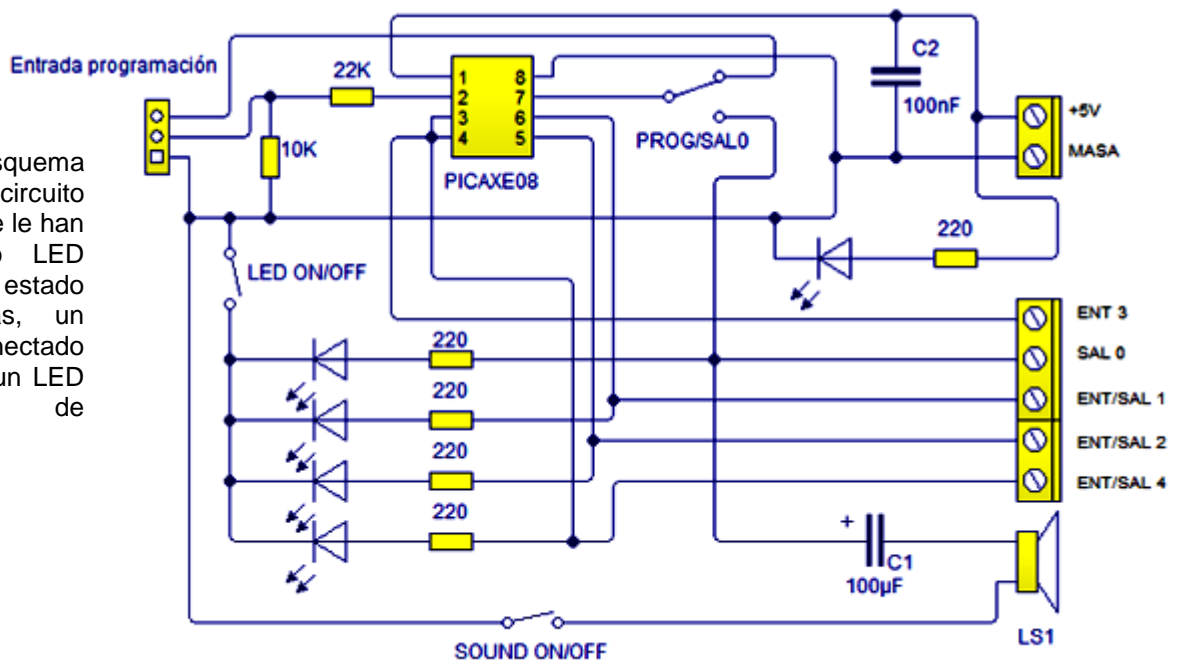


Cable de programación:

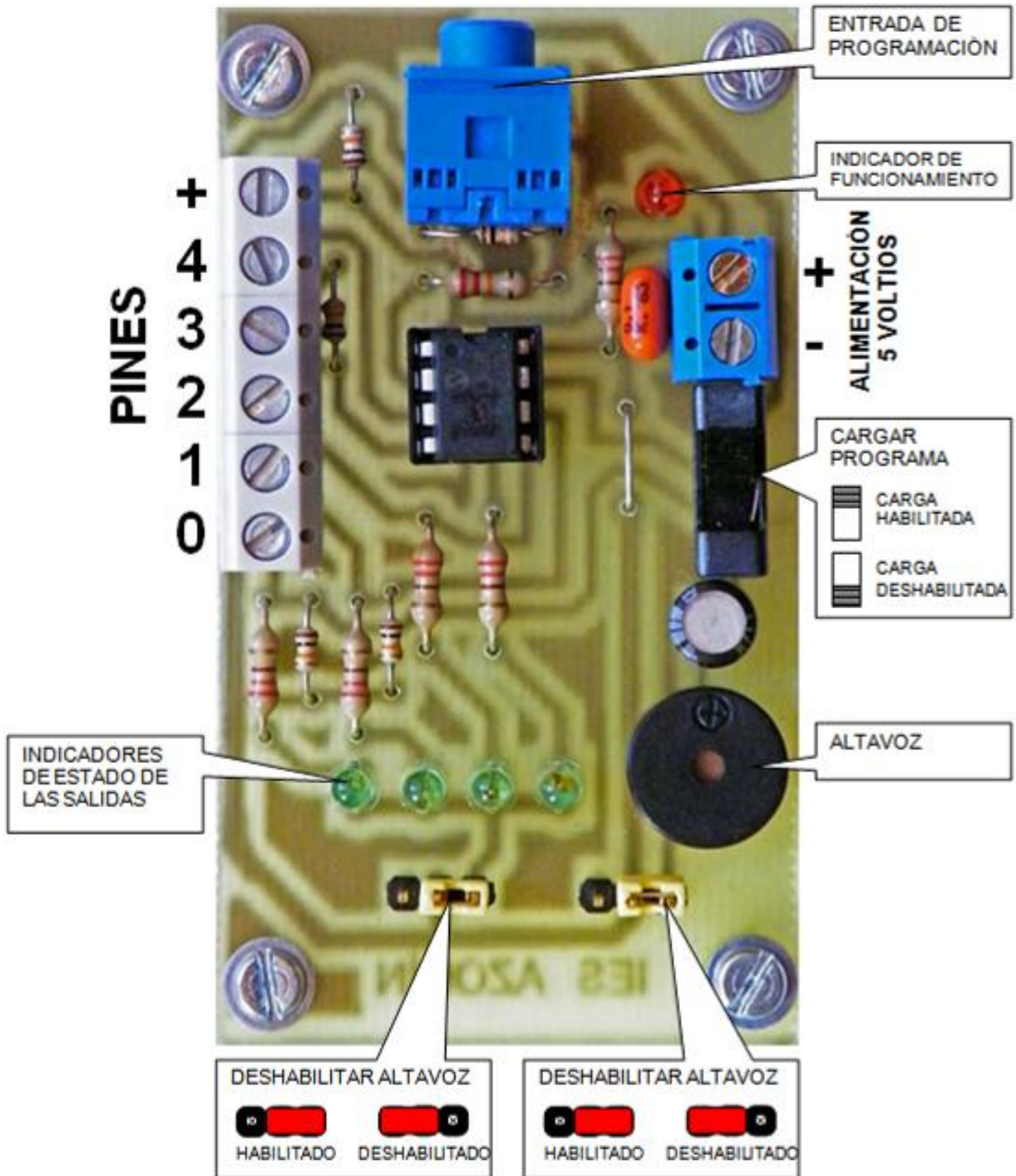


Esquema mejorado:

Este esquema representa el circuito básico, al que se le han añadido cuatro LED para indicar el estado de las salidas, un microaltavoz conectado a la salida 0 y un LED indicador de funcionamiento.



Resultado:



Prueba de la placa:

Para comprobar el correcto funcionamiento del circuito seguimos los siguientes pasos:

- Conectamos la alimentación (5v)
- Conectamos el cable de programación a la placa y al ordenador.
- Ponemos el interruptor en posición de carga.
- Abrimos el editor de programas e indicamos el tipo de chip que estamos usando (Picaxe 08).
- Escribimos uno de los programas de prueba y pulsamos el botón “cargar”.
- Cambiamos la posición del conmutador de carga.

1. Prueba de las salidas:

Este programa prueba las salidas. Es necesario colocar el jumper que habilita los LED en la posición correcta.

```
*****
!* prueba led *
*****
prueba: 'nombre del programa
high 0 'enciende LED en pin 0
wait 1 'retardo 1 segundo
high 1 'enciende LED en pin 1
wait 1 'retardo 1 segundo
high 2 'enciende LED en pin 2
wait 1 'retardo 1 segundo
high 4 'enciende LED en pin 4
wait 1 'retardo 1 segundo
```

2. Prueba del microaltavoz

Este programa prueba el altavoz conectado a la salida 0. Es necesario colocar el jumper que habilita el altavoz en la posición correcta.

```
*****
!* prueba sonido *
*****
prueba: 'nombre del programa

sound 0, (100,20) 'sonido en la salida 0 de tono 100 y duración 20
pause 100 'retardo de 100 milisegundos
sound 0, (110,30) 'sonido en la salida 0 de tono 110 y duración 30
pause 100 'retardo de 100 milisegundos
sound 0, (115,40) 'sonido en la salida 0 de tono 115 y duración 50
pause 100 'retardo de 100 milisegundos
sound 0, (120,50) 'sonido en la salida 0 de tono 120 y duración 50

goto prueba 'vuelve al principio del programa
```

3. Verificación de las entradas

Realizaremos el siguiente programa para verificar el funcionamiento de las entradas de la placa. En el ejemplo se ha hecho para la entrada 1, pero habrá que repetirlo para el resto (2, 3 y 4).

Una vez cargado el programa, se conecta un trozo de cable al positivo de la alimentación y una resistencia de 10k entre la entrada a probar y masa. Para probar la entrada en cuestión hay que tocar con el extremo del cable al borne de dicha entrada. Al hacer esto debe cambiar el sonido.

```

*****
!* prueba entrada 1 *
*****

prueba:                                'nombre del programa

sound 0, (7, 10)                        'sonido inicial de espera
pause 300                               'retardo 300 milisegundos

if input1 is on then sonido             'Si la entrada 1 está anivel
goto prueba                             'alto, se ejecuta la subrutina "sonido"
                                        'vuelve al inicio del programa

sonido:                                 'subrutina sonido

sound 0, (120, 10)                      'sonido agudo en 0 de duración 10 milisegundos
pause 30                                'retardo 30 milisegundos
sound 0, (120, 10)                      'sonido agudo en 0 de duración 10 milisegundos
pause 30                                'retardo 30 milisegundos
sound 0, (120, 10)                      'sonido agudo en 0 de duración 10 milisegundos
pause 30                                'retardo 30 milisegundos
sound 0, (120, 10)                      'sonido agudo en 0 de duración 10 milisegundos
pause 30                                'retardo 30 milisegundos
sound 0, (120, 10)                      'sonido agudo en 0 de duración 10 milisegundos
pause 30                                'retardo 30 milisegundos

goto prueba                             'vuelve al inicio del programa

```

Más programas sencillos:

```

*****
!* led movimiento *
*****

b0 = 70                                'define la variable b0 y le asigna el valor 70

led:                                    'nobre del programa

high 0                                  'enciende LED en pin 0
pause b0                                'retardo b0 milisegundos
low 0                                    'apaga LED en pin 0

high 1                                  'enciende LED en pin 1
pause b0                                'retardo b0 milisegundos
low 1                                    'apaga LED en pin 0

high 2                                  'enciende LED en pin 2
pause b0                                'retardo b0 milisegundos
low 2                                    'apaga LED en pin 0

high 4                                  'enciende LED en pin 4
pause b0                                'retardo b0 milisegundos
low 4                                    'apaga LED en pin 0

goto led                                 'vuelve al inicio

```

```
*****
* led movimiento 2 *
*****
b0 = 70
b1 = 100

led:

high 0      'enciende LED en pin 0
pause b0    'retardo b0 milisegundos

high 1      'enciende LED en pin 1
pause b0    'retardo b0 milisegundos

high 2      'enciende LED en pin 2
pause b0    'retardo b0 milisegundos

high 4      'enciende LED en pin 4
pause b0    'retardo b0 milisegundos

low 0       'apaga LED en pin 0
pause b1    'retardo b1 milisegundos

low 1       'apaga LED en pin 0
pause b1    'retardo b1 milisegundos

low 2       'apaga LED en pin 0
pause b1    'retardo b1 milisegundos

low 4       'apaga LED en pin 0
pause b1    'retardo b1 milisegundos

goto led    'vuelve al inicio
```

```
!*****
```

```
!* led movimiento 3 *
```

```
!*****
```

```
b0 = 70      'asigna el valor 70 a la variable b0  
b1 = 100     'asigna el valor 100 a la variable b1
```

```
led:        'nombre del programa
```

```
high 0      'enciende LED en pin 0  
pause b0    'retardo b0 milisegundos  
high 1      'enciende LED en pin 1  
pause b0    'retardo b0 milisegundos  
high 2      'enciende LED en pin 2  
pause b0    'retardo b0 milisegundos  
high 4      'enciende LED en pin 4  
pause b0    'retardo b0 milisegundos
```

```
low 0       'apaga LED en pin 0  
pause b1    'retardo b1 milisegundos  
low 1       'apaga LED en pin 0  
pause b1    'retardo b1 milisegundos  
low 2       'apaga LED en pin 0  
pause b1    'retardo b1 milisegundos  
low 4       'apaga LED en pin 0  
pause b1    'retardo b1 milisegundos
```

```
high 4      'enciende LED en pin 0  
pause b0    'retardo b0 milisegundos  
high 2      'enciende LED en pin 1  
pause b0    'retardo b0 milisegundos  
high 1      'enciende LED en pin 2  
pause b0    'retardo b0 milisegundos  
high 0      'enciende LED en pin 4  
pause b0    'retardo b0 milisegundos
```

```
low 4       'apaga LED en pin 0  
pause b1    'retardo b1 milisegundos  
low 2       'apaga LED en pin 0  
pause b1    'retardo b1 milisegundos  
low 1       'apaga LED en pin 0  
pause b1    'retardo b1 milisegundos  
low 0       'apaga LED en pin 0  
pause b1    'retardo b1 milisegundos
```

```
goto led    'vuelve al inicio
```