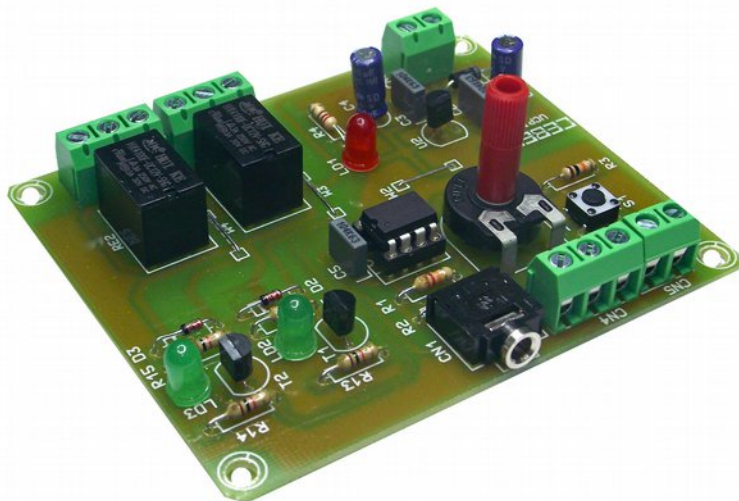




PICAXE 08M2 Circuit - 2 Relays Circuit PICAXE 08M2 - 2 Relais Circuito PICAXE 08M2 - 2 Reles UCPIC-5



Characteristic

Voltage : 12 V DC
Minimum consumption : 15 mA
Maximum consumption : 75 mA
Supported Picaxe : 08M2 /08M
Relay Maximum Load : 2 Amp / 230 VAC
Protection reverse polarity : yes
Recommended power supply : FE 103
Size : 72 x 87 x 30 mm.
Din rail : C-7566

It incorporates micro PICAXE 08M2. Pushbutton digital input, analog input potentiometer, two relays output 2 amps. Power indicator LEDs and relays. Connector recording. Indicator automation development requiring external control switch and potentiometer with, for example, sequential, timers.

Important : maximum allowable voltage PIN 3-4: 5VDC

Il intègre micro PICAXE 08M2. Entrée Pushbutton numérique, 4 relais de sortie 2 ampères. Indicateurs LED alimentation et relais. Connecteur pour enregistrement. Indicateur de développement d'automatisation exigeant commutateur de contrôle externe avec, par exemple, séquentielle, temporisateur.

Important : maximum PIN de tension admissible 3-4: 5VDC

Incorpora micro PICAXE 08M2. Entrada digital con pulsador, entrada analógica con potenciómetro, dos relés de salida 2 Amp. Leds indicadores de alimentación y relés. Conector de grabación. Indicador para el desarrollo de automatismo que requiera control externo con un pulsador y potenciómetro, por ejemplo, secuencial, temporizadores.

Importante : Tensión máxima admisible PIN 3-4 : 5VDC

- **Pin3:** It has the function button. If you activate its value will be 1 if it is not, its value is 0.
- **Pin4:** It has the function of potentiometer. To read its value use the readadc 4 instruction (variable where we store the value). Its value can range from 0 to 256 according to our needs
- **Pin1 And pin2 :** Correspond to the outputs, connected relays, indicated by LEDs.

Equiv. Pin versus PICAXE 08M2 PIN3=4 PIN4=3 PIN1=6 PIN2=5

- **Pin3:** Il a la touche de fonction. Si vous activez sa valeur sera 1 si elle est pas, sa valeur est 0.
- **Pin4:** Il a la fonction du potentiomètre. Pour lire sa valeur utiliser le readadc 4 instruction (variable à laquelle nous stockons la valeur). Sa valeur peut aller de 0 à 256 selon nos besoins
- **Pin1 Et PIN2:** Correspondent aux sorties, relais connectés, indiqués par des LED.

Equiv. Pin versus PICAXE 08M2 PIN3=4 PIN4=3 PIN1=6 PIN2=5

- **Pin3:** Tiene la función de pulsador. Si lo activamos su valor será 1, si no es así, su valor será 0.
- **Pin4:** Tiene la función de potenciómetro. Para leer su valor utilizaremos la instrucción readadc 4, (variable donde guardaremos el valor). Su valor puede variar de 0 a 256 según nuestras necesidades
- **Pin1 y pin2 :** Corresponden a la salidas, conecta relé, indicado por leds.

Equiv. Pin versus PICAXE 08M2 PIN3=4 PIN4=3 PIN1=6 PIN2=5

PRACTICE 1

Very simple program in which outputs turn on and off sequentially

Programme très simple dans lequel les sorties allumer et éteindre séquentiellement

Programa muy sencillo en el cual las salidas de activan y desactivan secuencialmente

```
main:
high 1
pause 2000
high 2
pause 2000
low 2
pause 2000
low 1
pause 2000
goto main
```

1The instruction pause = milliseconds. To change it, write "pause X", where X desired milliseconds. We can also use the wait instruction, which is measured in seconds; 4 = wait wait 4 seconds. To change the value, type "wait X", where X desired seconds. 1000 milliseconds = 1 second.

1Le instruction pause = millisecondes. Pour le modifier, écrire "pause X", où X millisecondes souhaité. Nous pouvons également utiliser l'instruction d'attente, qui est mesurée en secondes; 4 = attente attendre 4 secondes. Pour modifier la valeur, tapez "attente X", où X désiré secondes. 1000 millisecondes = 1 seconde.

1La instrucción pause = milisegundos. Para cambiarla, escribir "pause X", donde X los milisegundos deseados. También podemos utilizar la instrucción wait, que se mide en segundos; wait 4 = espera de 4 segundos. Para cambiar el valor, escribir "wait X", donde X los segundos deseados. 1000 milisegundos = 1 segundo.

PRACTICE 2

The first condition of this program refers to the button. Each time we activate the button, an output is activated. When both are activated and you press the button again will turn off and the program returns to the main .

La première condition de ce programme fait référence au bouton. Chaque fois que nous activons le bouton, une sortie est activée. Lorsque les deux sont activés et que vous appuyez à nouveau sur le bouton pour éteindre et le programme revient à la principale .

La primera condición de este programa hace referencia al pulsador. Cada vez que activemos el pulsador, se activará una salida. Cuando estén las dos activadas y pulsemos de nuevo el pulsador, se desactivarán y el programa volverá al main .

```
main:
if pin3=1 then condicion_1
if pin3=0 then main
condicion_1:
high 1
pause 200
if pin3=1 then condicion_2
if pin3=0 then condicion_1
condicion_2:
high 2
pause 200
if pin3=1 then apagar
if pin3=0 then condicion_2
apagar:
low 2
low 1
pause 200
goto main
```

Pause instruction 200, it is to avoid unsuccessful pulses

Pause instruction 200, il est d'éviter des impulsions infructueuses

Instrucción pause 200 , es para evitar pulsos fallidos

PRACTICE 3

These instructions depend on the potentiometer. Depend on their value will activate an exit or another. When the pot reaches its maximum value, all outputs are disabled.

Ces instructions dépendent du potentiomètre. Dépendre de leur valeur activera une sortie ou d'une autre. Lorsque le pot atteint sa valeur maximale, toutes les sorties sont désactivées.

Estas instrucciones dependen del potenciómetro. Depende de su valor activaremos una salida u otra. Cuando el potenciómetro llegue a su valor máximos, todas las salidas activadas se desactivarán.

```

main:
potenciometro:
readadc 4, b0
if b0<130 then led_1
if b0>130 and b0<230 then led_2
if b0>230 then low_leds
led_1:
low 2
high 1
goto potenciometro
led_2:
low 1
high 2
goto potenciometro
low_leds:
low 2
low 1
goto potenciometro

```

2 Values are approximate. The user can change between 0 and 255 modifying the value of 130 or 230 by choice.

2 Les valeurs sont approximatives. L'utilisateur peut changer entre 0 et 255 modifiant la valeur de 130 ou 230 par choix.

2 Los valores son orientativos. El usuario puede cambiarlos entre 0 y 255 modificando el valor de 130 ó 230 por el que desee.

PRACTICE 4

Practice similar to the above, the difference is that in this condition have a push-dependent.

Pratique analogue à ce qui précède, la différence est que, dans cette condition ont un bouton-dépendant.

Práctica similar a la anterior, la diferencia es que en esta tenemos una condición que depende del pulsador.

```

main:
if pin3=1 then potenciometro
if pin3=0 then main
potenciometro:
readadc 4, b0
if b0<130 then led_1
if b0>130 and b0<230 then led_2
if b0>230 then low_leds
led_1:
low 2
high 1
goto potenciometro
led_2:
low 1
high 2
goto potenciometro
low_leds:
low 2
low 1
goto potenciometro

```

The guide values. The user can change them by modifying the value of 130 or 230 by the desired between 0 and 255.

Les valeurs de guidage. L'utilisateur peut modifier en changeant la valeur de 130 ou 230 par la souhaitée entre 0 et 255.

Los valores orientativos . El usuario puede cambiarlos modificando el valor de 130 ó 230 por el que desee entre 0 y 255.

PRACTICE 5

We have a status button and, if it is true, the outputs are activated and deactivated sequentially ³.

Nous avons un bouton d'état et, si elle est vraie, les sorties sont activées et désactivées séquentiellement ³.

Tenemos una condición de pulsador y, si ésta se cumple, las salidas se activarán y desactivarán secuencialmente ³.

```

main:
if pin3=1 then secuencia_leds
if pin3=0 then main
secuencia_leds:
high 1
pause 1000
high 2
pause 1000
low 2
pause 1000
low 1
pause 1000
goto main

```

The instruction pause = milliseconds. To change it, write "pause X", where X desired milliseconds. We can also use the wait instruction, which is measured in seconds; 4 = wait wait 4 seconds. To change the value, type "wait X", where X desired seconds. 1000 milliseconds = 1 second.

Le instruction de pause = millisecondes. Pour le modifier, écrire "pause X", où X souhaité millisecondes. Nous pouvons également utiliser l'instruction wait, qui est mesurée en secondes; 4 = attente 4 secondes. Pour modifier la valeur, tapez "wait X", où X désiré secondes. 1000 millisecondes = 1 seconde.

La instrucción pause = milisegundos. Para cambiarla, escribir "pause X", donde X los milisegundos deseados. También podemos utilizar la instrucción wait, que se mide en segundos; wait 4 = espera de 4 segundos. Para cambiar el valor, escribir "wait X", donde X los segundos deseados. 1000 milisegundos = 1 segundo.

PRACTICE 6

We have a button and if this condition is met outputs are activated and deactivated according sequentially potentiometer value.

Nous avons un bouton et si cette condition est remplie sorties sont activées et désactivées selon la valeur du potentiomètre séquentiellement.

Tenemos una condición pulsador y si esta se cumple las salidas se activaran y desactivaran secuencialmente según valor potenciómetro .

```
main:
if pin3=0 then main
if pin3=1 then time_on
time_on:
readadc 4, b0
for b1=0 to b0
high 1
pause 1000
low 1
high 2
pause 1000
low 2
next b1
goto main
```

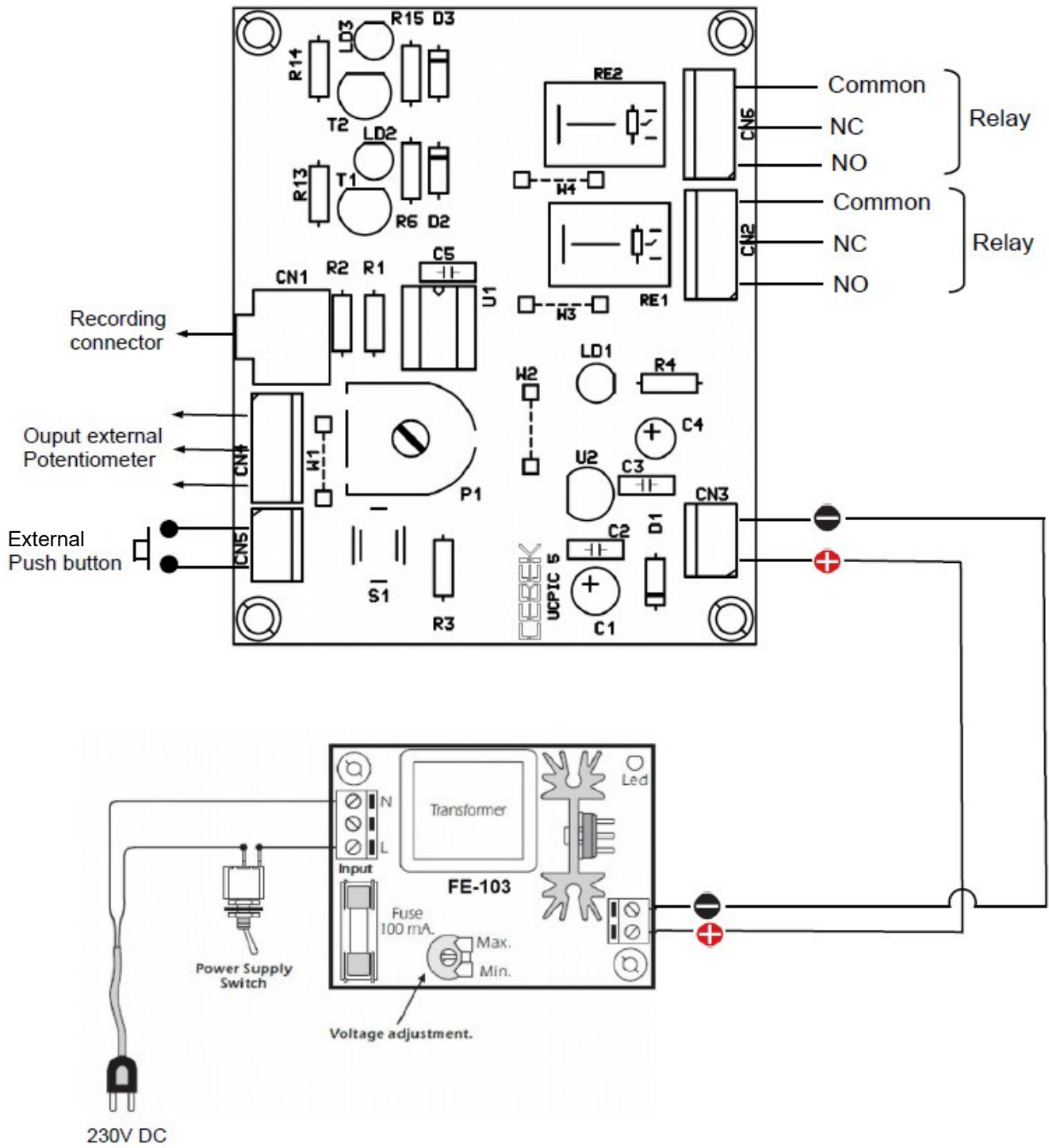
3The instruction pause = milliseconds. To change it, write "pause X", where X desired milliseconds. We can also use the wait instruction, which is measured in seconds; 4 = wait wait 4 seconds. To change the value, type "wait X", where X desired seconds. 1000 milliseconds = 1 second.

3Le instruction de pause = millisecondes. Pour le modifier, écrire "pause X", où X souhaité millisecondes. Nous pouvons également utiliser l'instruction d'attente, qui est mesurée en secondes; 4 = attente attendre 4 secondes. Pour modifier la valeur, tapez "attente X", où X désiré secondes. 1000 millisecondes = 1 seconde.

3La instrucción pause = milisegundos. Para cambiarla, escribir "pause X", donde X los milisegundos deseados. También podemos utilizar la instrucción wait, que se mide en segundos; wait 4 = espera de 4 segundos. Para cambiar el valor, escribir "wait X", donde X los segundos deseados. 1000 milisegundos = 1 segundo.



Cebek [®] is a registered trademark of the Fadisel group



Important: Ne jamais retirer ou insérer le picaxe sous tension, endommager de façon Irréparable

www.cebekit.com - info@cebekit.com