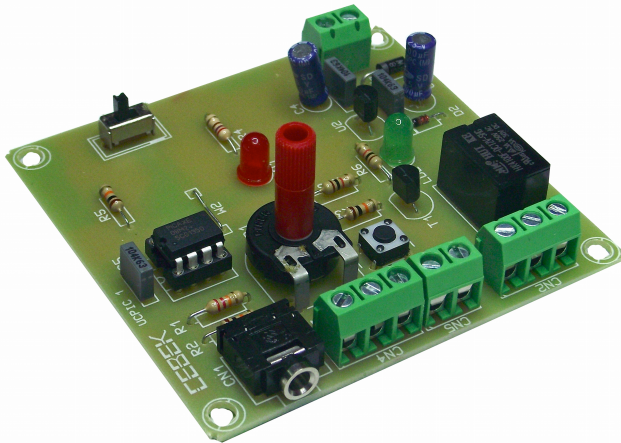


Circuit PICAXE 08M2 tempos Tempos Circuit PICAXE 08M2 Circuito PICAXE 08M2 tempos UCPIC1



Technical Characteristic

Voltage : 12 V DC
 Minimum consumption : 18 mA
 Maximum consumption : 48 mA
 Picaxe suportados : 08M2 /08M
 Relay Maximum Load : 2 Amp / 230 VCA
 Protection reverse polarity : yes
 Recommended source : FE 103
 Size : 72 x 76 x 30 mm.
 Din rail : C 7565

Picaxe incorporates 08 micro M2. Digital input with push, pot adjustment, function switch, 2A relay output, and power indicators, relay recording connector suitable for development of automation that require external control with a button and a potentiometer, for example a timer.

Picaxe intègre 08 micro M2. Entrée numérique avec pousoir pot ajustement, conmetateur funtion sortie relais 2A, et le connecteur d'enregistrement puissance indicateurs Relais approprié pour le développement de l'automatisation qui nécessitent un contrôle externe avec un bouton et un potentiomètre, par exemple une minuterie.

Incorpora micro Picaxe 08 M2 . Entrada digital con pulsador, pot ajuste, conmutador de función, relé de salida 2A, leds indicadores de alimentación y relé, conector de grabación indicado para desarrollo de automatismos que requieren control externo con un pulsador y un potenciómetro, por ejemplo un temporizador.

IMPORTANT: Maximum allowable voltage PIN 3-4 : 5V DC

IMPORTANT: Tension maximale admissible PIN 3-4 : 5V DC

IMPORTANTE : Tensión máxima admisible PIN 3-4 : 5V DC

- **Pin1** : Its function switch, used in some functions depending on the condition you want (pin1 "on" or "off")
 - **Pin3** : It has the push function. If we activate its value is 1, if not, the value will be 0.
 - **Pin4** : Its function potentiometer. To read the value use the readadc 4 instruction (variable where we store the value). Its value can range from 0-256 as our special needs
 - **Pin2** : Corresponds to output connects relay, indicated by a LED
- Equiu pin versus PICAXE 08M2 PIN1=6 PIN2=5 PIN3=4 PIN4=3

- **Pin1** : Son sélecteur de fonction, utilisé dans certaines fonctions selon l'état que vous voulez (pin1 "on" ou "off")
 - **Pin3** : A la fonction de poussée. Si nous activons sa valeur est 1, sinon, la valeur sera de 0.
 - **Pin4** : Il a la función potentiomètre. Pour lire sa valeur utiliser les quatre instruction readadc (variable où nous stockons la valeur). Sa valeur peut aller de 0 à 256 pour répondre à nos besoins
 - **Pin2** : correspond à la sortie, connectez relais, indiqué par une LED
- Equiu pin versus PICAXE 08M2 PIN1=6 PIN2=5 PIN3=4 PIN4=3

- **Pin1** : Tiene la función de conmutador, lo utilizamos en algunas funciones según la condición que queramos (pin1 "on" u "off")
 - **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
 - **Pin2** : Corresponde a la salida, conecta relé, indicado por un led
- Equiu pin versus PICAXE 08M2 PIN1=6 PIN2=5 PIN3=4 PIN4=3

PRACTICE 1

Easy program of timer with pause. If pin3 on, the output is set for 5000 milliseconds. This time is marked by the instruction "pause 5000"

Simple programme de temporisateur avec pause. Si pin3 sur, la sortie est activ pour 5000 millisecondes. Ce temps est marquée par l'instruction «pause 5000"

Sencillo programa de temporizador con pausa. Si pin3 está activado, la salida se activa durante 5000 milisegundos. Este tiempo viene marcado por la instrucción "pause 5000".

```

main:
if pin3=1 then tempo
if pin3=0 then main
goto main
tempo:
high 2
pause 5000
low 2
goto main

```

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

1 - L'instruction de pause = millisecondes. Pour le changer, écrire "pause X", où X millisecondes souhaité. Nous pouvons également utiliser la commande Wait, ce qui mesuré en secondes; 4 = attendent attendre 4 secondes. Pour modifier la valeur, tapez «Wait X», où X souhaite secondes. 1000 millisecondes = 1 seconde.

1 - 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 2

Push status. If pin3 on, perform "for" loop is repeated as many times as the value of the potentiometer. We add a delay of 1000 milliseconds, marked by a pause = statement 1000.

Poussez le statut. Si pin3 sur, effectuez boucle "for" est répété autant de fois que la valeur du potentiomètre. Nous ajoutons un délai de 1000 millisecondes, marquée par l'instruction pause = 1000.

Condición de pulsador. Si pin3 activado, realizamos bucle "for" que se repetirá tantas veces como el valor del potenciómetro. Le añadimos un delay de 1000 milisegundos, marcado por la instrucción pause =1000.

```

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

```

2 - pause instruction = milliseconds. To change it, write "pause X", where X desired milliseconds. We may 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.

2 - instruction de pause = millisecondes. Pour le changer, écrire "pause X", où X millisecondes souhaité. Nous pouvons également utiliser l'instruction Wait, qui est mesuré en secondes; 4 = attendent attendre 4 secondes. Pour modifier la valeur, tapez «Wait X», où X souhaite secondes. 1000 millisecondes = 1 seconde.

2 - 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 3

Cyclic program, the output is activated and deactivated for 1000 milliseconds (pause = 1000.)

Programme cyclique, la sortie est activée et désactivée pour 1000 millisecondes (pause = 1000.)

Programa cíclico, la salida se activa y se desactiva durante 1000 milisegundos (pause =1000.)

```

Main:
high 2
pause 1000
low 2
pause 1000
goto main

```

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

3 - instruction de pause = millisecondes. Pour le changer, écrire "pause X", où X = millisecondes souhaité. Nous pouvons également utiliser la commande Wait, ce qui mesuré en secondes; 4 = attendent attendre 4 secondes. Pour modifier la valeur, tapez «Wait X», où X souhaite secondes. 1000 millisecondes = 1 seconde.

3 - La instrucción pause = milisegundos. Para cambiarla, escribir "pause X", donde X son 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 4

Push status. Symbol = constant, ie time_1 = 6 for all instructions. Des 1 loop is repeated until the value of the constant 6. That is, five cycles are performed. We add a delay of 1000 milliseconds marked by the instruction "pause 1000". To change the constant symbol write X = Y, where X is the name of the constant, and the value of the constante .

Poussez le statut. Symbol = constante, à savoir Date_1 = 6 pour toutes les instructions. Des une boucle est répétée jusqu'à ce que la valeur de la constante 6. Autrement dit, cinq cycles sont effectués. Nous ajoutons un délai de 1000 millisecondes marqués par l'instruction «pause 1000». Pour changer le symbole écriture constante X = Y, où X est le nom de la constante, et la valeur de la constante.

Condición de pulsador. Symbol = constante, es decir time_1 = 6 durante todas las instrucciones. Bucle se repite des de 1 hasta el valor de la constante 6. Es decir, se realizan 5 ciclos. Le añadimos un delay de 1000 milisegundos marcado por la instrucción "pause 1000". Para cambiar la constante, escribir symbol X = Y, donde X es el nombre de la constante e Y el valor de la constante .

```
symbol time_1=6
main:
if pin3=1 then time_on
if pin3=0 then main
time_on:
for b1=1 to time_1
high 2
pause 1000
next b1
low 2
goto main
```

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

4 - L'instruction de pause = millisecondes. Pour le changer, écrire "pause X", où X millisecondes souhaité. Nous pouvons également utiliser la commande Wait, ce qui mesuré en secondes; 4 = attendent attendre 4 secondes. Pour modifier la valeur, tapez «Wait X», où X souhaite secondes. 1000 millisecondes = 1 seconde.

4 - 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 5

This program must meet two condiciones.pin1 = 1 and pin3 = 1. A constant throughout the program time_1 be worth 4. Loop "for" will be held as often as the constant mark. We added a double delay: pause pause 1000 + 3000 = 4000 msec Delay overall. To change the constant symbol write X = Y, where X is the name of the constant, and the value of the constant.

Ce programme doit répondre à deux condiciones.pin1 = 1 et pin3 = 1. Une constante tout au long du programme Date_1 vaudra 4. Loop "pour" se tiendra aussi souvent que la marque constante. Nous avons ajouté une double retard: Pause 1000 + 3000 = 4000 ms retard global. Pour changer le symbole écriture constante X = Y, où X est le nom de la constante, et la valeur de la constante.

En este programa debemos cumplir dos condiciones.pin1 = 1 y pin3 = 1. Una constantes, durante todo el programa time_1 tendrá un valor de 4. Bucle "for" se realizará tantas veces como marca la constante. Le añadimos un doble delay: pause 1000 + pause 3000. Delay total = 4000 milisegundos. Para cambiar la constante, escribir symbol X = Y, donde X es el nombre de la constante e Y el valor de la constante.

```

symbol time_1=4
main:
if pin1=0 then pulsador
if pin1=1 then time_on
pulsador:
if pin3=0 then main
if pin3=1 then time_on
time_on:
for b0=0 to time_1
high 2
pause 1000
next b0
low 2
pause 3000
goto main

```

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

5 - La commande de pause = millisecondes. Pour le changer, écrire "pause X", où X millisecondes souhaité. Nous pouvons également utiliser la commande wait, ce qui mesuré en secondes; 4 = attendent attendre 4 secondes. Pour modifier la valeur, tapez «wait X», où X souhaite secondes. 1000 millisecondes = 1 seconde .

5 - 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

Cycle times. The program will not stop unless we disconnect the feeding plate. Two constants, and stopped_time time_1 = 3 = 4. To change the constant symbol write X = Y, where X is the name of and Y constant constant value. We have two loops "for" that undertake as many times as the constant check. One holds the output 2 deactivated and the other remains active. The two loops have a delay of 1000 milliseconds marked by the instruction "pause 1000".

Les temps de cycle. Le programme ne sera pas arrêter si nous ne débrancher le alimentation assiette. Deux des constantes, et stopped_time Date_1 = 3 = 4. Pour changer le symbole écriture constante X = Y, où X est le nom de et Y valeur constante constante. Nous avons deux boucles "pour" que entreprendre autant de fois que le contrôle constant. On tient la sortie 2 désactivé et l'autre reste actif. Les deux boucles ont un retard de 1000 millisecondes marqués par l'instruction «pause 1000".

Tiempos cíclicos. El programa no parará a no ser que desconectemos la placa de la alimentación. Dos constantes, time_1 = 3 y stopped_time = 4. Para cambiar la constante, escribir symbol X = Y, donde X es el nombre de la constante e Y el valor de la constante. Tenemos dos bucles "for" que se realizaran tantas veces como marque la constante. Uno mantiene la salida 2 desactiva y el otro la mantiene activa. Los dos bucles tienen un delay de 1000 milisegundos marcado por la instrucción "pause 1000".

```

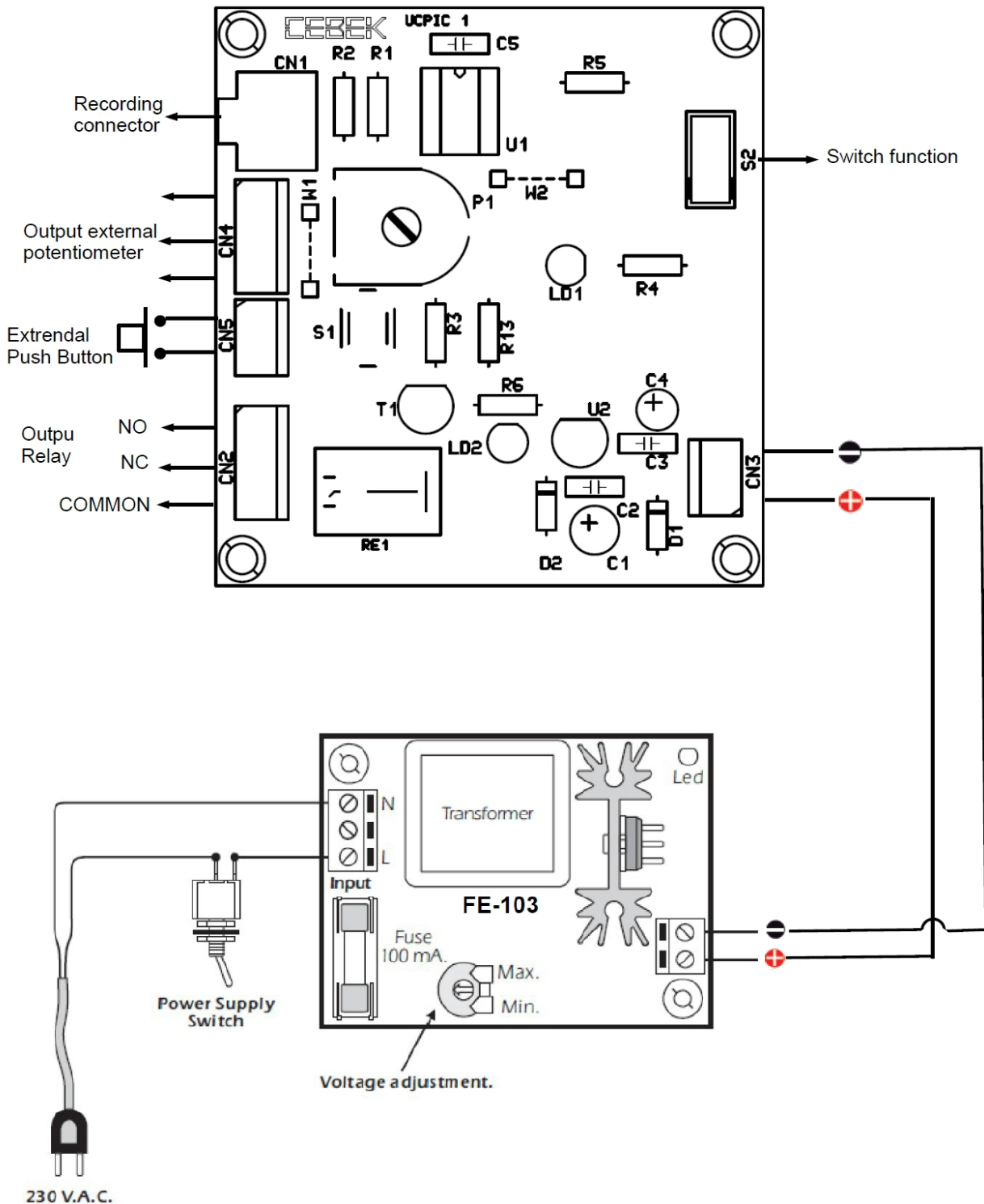
symbol time_1=3
symbol stopped_time=4
main:
if pin3=1 then time_on
if pin3=0 then time_off
goto main
time_on:
for b1=0 to time_1
high 2
pause 1000
next b1
goto time_off
time_off:
for b1=0 to stopped_time
low 2
pause 1000
next b1
goto time_on

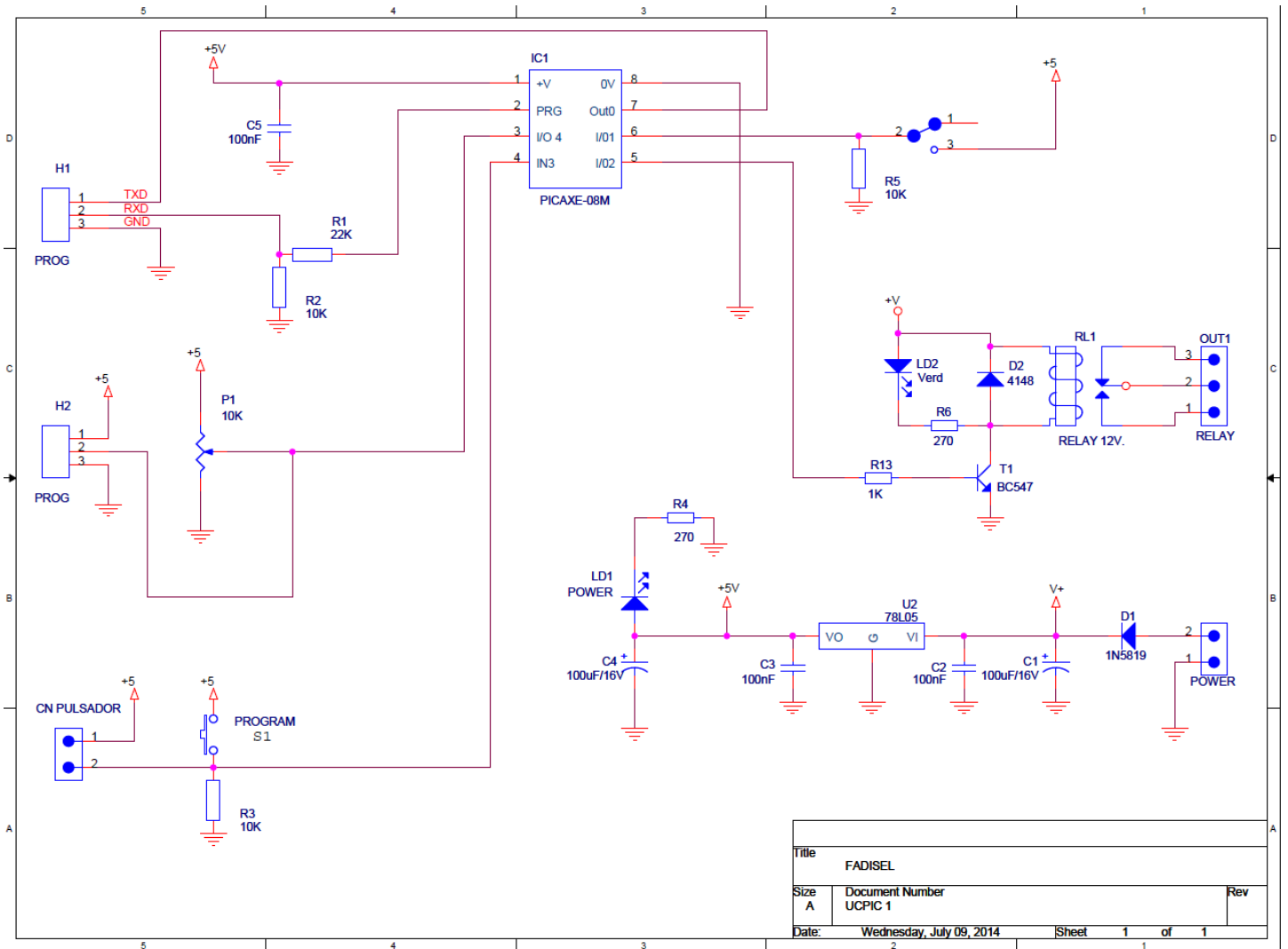
```

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

6 - instruction pause = millisecondes. Pour le changer, écrire "pause X", où X millisecondes souhaité. Nous pouvons également utiliser la commande wait, ce qui mesuré en secondes; 4 = attendent attendre 4 secondes. Pour modifier la valeur, tapez «wait X», où X souhaite secondes. 1000 millisecondes = 1 seconde.

6 - La instrucción pause = milisegundos. Para cambiarla, escribir "pause X", dónde 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", dónde X los segundos deseados. 1000 milisegundos = 1 segundo.





Important: Never remove or insert the picaxe with the power on, irreparably damage it.

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

Importante : Nunca extraiga o inserte el picaxe con la alimentación conectada , lo dañaría irreparablemente



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