



## Práctica 1.2.

## Set - Reset

El contenido difundido o publicado compromete exclusivamente a los autores.

Proyectos de innovación: CÉLULA ROBOTIZADA



## INDICE

<b>1. Bloque de ejercicios para familiarizarse con el código Nivel 1. (Entradas, Salidas, Set, Reset, Temporización).....</b>	<b>3</b>
<hr/>	
1.1.- ENUNCIADO .....	3
1.2.- Conocimientos previos .....	3
1.3.- Descripción de las entradas, salidas e instrucciones a utilizar.....	3
1.4.- Vista para la simulación.....	4
1.5.- Programa a realizar.....	4



## 1. BLOQUE DE EJERCICIOS PARA FAMILIARIZARSE CON EL CÓDIGO NIVEL 1. (ENTRADAS, SALIDAS, SET, RESET, TEMPORIZACIÓN)

---

### 1.1.- ENUNCIADO.

**Si se pulsa marcha se enciende el indicador luminoso del botón marcha, si se pulsa paro se apaga el indicador del pulsador.**

### 1.2.- Conocimientos previos.

Partiendo del ejercicio anterior, donde hemos visto como se pone a 0 ó a 1 una salida, hay que aclarar que el valor que se introduce a una variable lo mantendrá hasta que se modifique, por lo tanto, se puede utilizar en variables booleanas como un SET o un RESET, en función de si se le introduce un 1 o un 0.

Como suele ser habitual, todos los paros se suelen cablear en lógica negativa, es decir, con contactos normalmente cerrados NC, por lo tanto, cuando no esté pulsado a la entrada se tendrá un 1, y cuando se pulse se tendrá un 0.

### 1.3.- Descripción de las entradas, salidas e instrucciones a utilizar.

Nombre	Descripción	Dirección	Función
S1	Pulsador de Marcha	Bit 3	Entrada
S2	Pulsador de Paro	Bit 4	Entrada
L1	Indicador Luminoso Pulsador Marcha	Bit 0	Salida



#### 1.4.- Vista para la simulación.



#### 1.5.- Programa a realizar.

El programa queda con el siguiente aspecto:

```

'-----
'Bloque de ejercicios para familiarizarse con el código Nivel 1.
'(Entradas, Salidas, Set, Reset, Temporización)
'Ejercicio 2
'
'Si se pulsa marcha se enciende el indicador luminoso del botón,
'al soltar el pulsador se apaga dicho indicador.
'-----
'-----
'Se definen las variables que se van a utilizar.
'-----
DEF IO S1 = BIT, 3 'Se define el pulsador de marcha como el bit 3
DEF IO S2 = BIT, 4 'Se define el pulsador de paro como el bit 4
DEF IO L1 = BIT, 0 'Se define el indicador luminoso del pulsador de
                    'marcha como el bit 0
'-----
'Programa
'-----

'Se declara la etiqueta de salto al inicio del programa
*Inicio

'Si el pulsador de marcha está activo se activará la salida del indicador
luminoso, y se mantendrá activo hasta que se indique lo contrario
If S1=1 then
    L1=1
EndIf
'Si el pulsador de paro está activo se desactivará la salida del indicador
luminoso
'Atención: El pulsador de paro es NC, por lo tanto habrá que tenerlo en
cuenta a la hora de programarlo
If S2=0 then
    L1=0
EndIf

'Al finalizar el programa se salta al inicio para que este sea cíclico.
Goto *Inicio

```