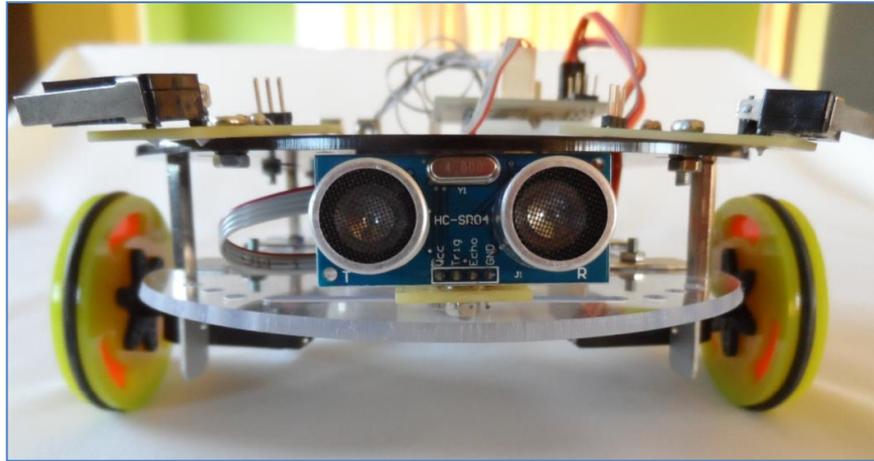


Unidad 2.1 - Programación de Actuadores - LED

# Robótica Educativa

Robot T-17

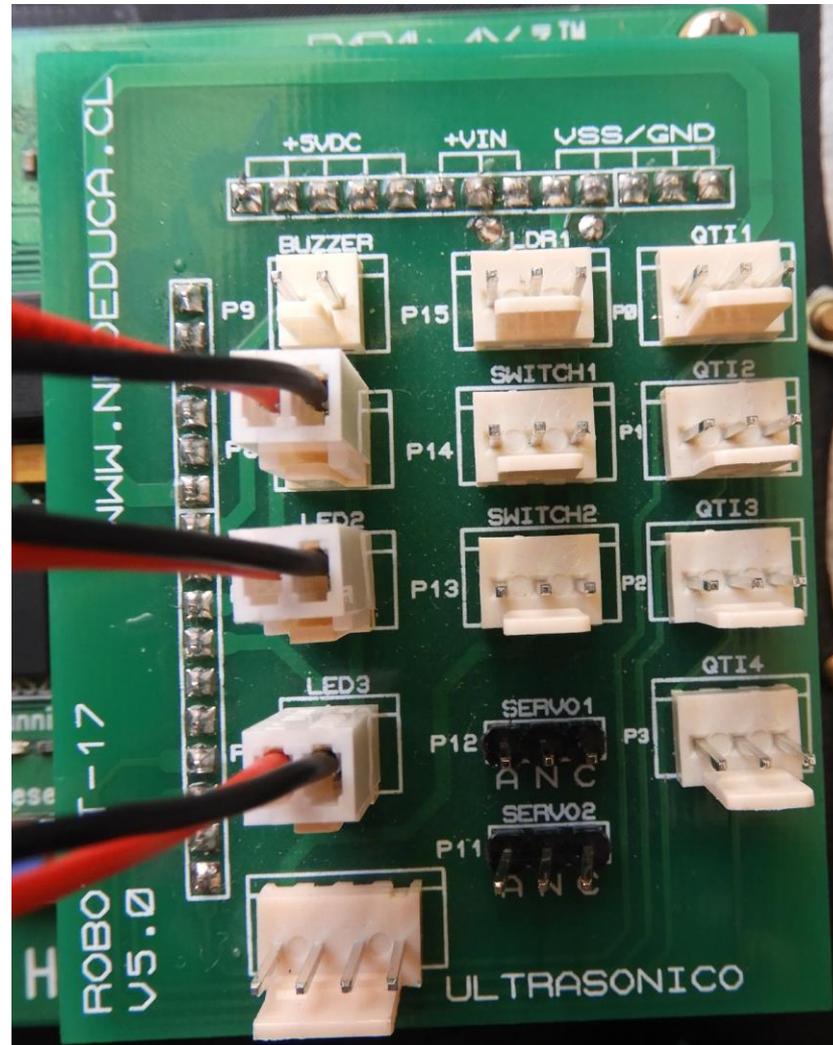


# Actuador LED



- **LED:** Diodo Emisor de Luz (*Light Emitted Diode*)
- En la Shield T-17 están conectados a los pines **P6**, **P7** y **P8** (o P8, P9, P10 según el modelo de Shield).
- Para hacer las conexiones en el tablero de circuitos (breadboard), se recomienda utilizar los mismos pines.
- Es posible programarlos con softwares como el *BasicStamp Editor* o *SCRIBBLER*.

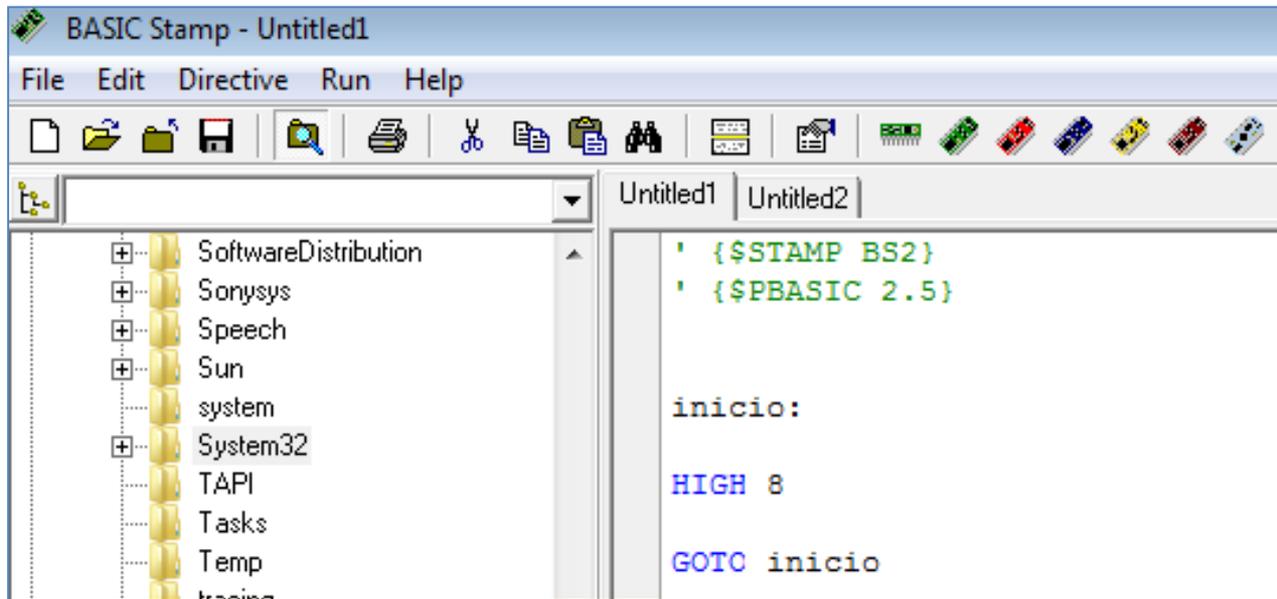
# Conexión LED



# Comandos **HIGH** y **LOW**

## LUCES

- **HIGH 8**, energiza línea 8
- **LOW 8**, desenergiza línea 8



The screenshot shows the BASIC Stamp IDE interface. The window title is "BASIC Stamp - Untitled1". The menu bar includes "File", "Edit", "Directive", "Run", and "Help". The toolbar contains various icons for file operations and hardware-related functions. The left pane shows a file explorer with folders like "SoftwareDistribution", "Sonsys", "Speech", "Sun", "system", "System32", "TAPI", "Tasks", "Temp", and "tracking". The main editor area shows the following code:

```
' {$STAMP BS2}
' {$PBASIC 2.5}

inicio:

HIGH 8

GOTO inicio
```

# Comando **PAUSE**



## RECUERDA QUE

La unidad de tiempo que utiliza la placa es milisegundo.

**PAUSE** 1000

ROBOT T-17

# DESAFÍOS



# DESAFÍO 1



Desarrollar un código que energice un LED por 1000 milisegundos y luego que se desenergice.

# DESAFÍO 2



Desarrolla un código que encienda tres leds en forma simultánea por 1000 milisegundos y luego apagarlos al mismo tiempo.

# DESAFÍO 3



Desarrolla un código que encienda tres leds en forma simultánea por 1000 milisegundos y luego apagarlos al mismo tiempo. Mostrar un mensaje por pantalla cuando comience y cuando termine de ejecutarse el programa.

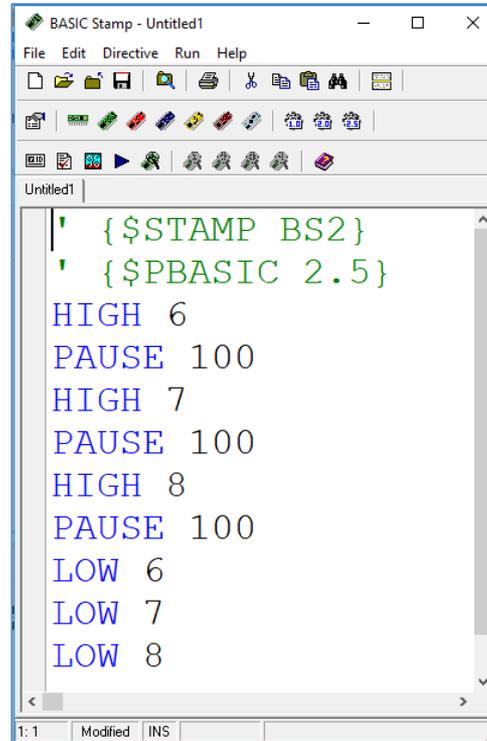
# DESAFÍO 4



Desarrolla un código que encienda tres leds en forma simultánea por 1000 milisegundos y luego apagarlos de uno en uno con pausas de medio segundo.

# DESAFÍO 5

Describa en palabras la acción que realizan los actuadores LED durante la siguiente rutina:



```
BASIC Stamp - Untitled1
File Edit Directive Run Help
[Icons]
[Icons]
[Icons]
Untitled1
' {$STAMP BS2}
' {$PBASIC 2.5}
HIGH 6
PAUSE 100
HIGH 7
PAUSE 100
HIGH 8
PAUSE 100
LOW 6
LOW 7
LOW 8
1:1 Modified INS
```

# DESAFÍO 6



Desarrolla un código que encienda tres leds de uno en uno con pausas de 500 milisegundos y luego apagarlos todos al mismo tiempo.

# DESAFÍO 7



Desarrolla un código que encienda tres leds de uno en uno con pausas de 500 milisegundos y luego apagarlos todos al mismo tiempo.

Mostrar por pantalla un mensaje indicando el color del led en el momento en que se enciende.

# DESAFÍO 8



Desarrolla un código que encienda tres leds de uno en uno con pausas de 500 milisegundos y luego apagarlos de uno en uno con pausas de medio segundo.

# DESAFÍO 9



Desarrolla un código que haga parpadear un led 3 veces. Debe permanecer encendido por medio segundo y apagado por un segundo.

# Ciclo **DO-LOOP**



- Permite crear un ciclo.
- Siempre finaliza con **LOOP**.
- El ciclo es infinito.
- Los formato de programación se presenta a continuación:

# Variantes DO-LOOP

## Ciclo Infinito:

```
Do  
  Debug "texto"  
loop
```

# DESAFÍO 10



Desarrollar un código que permita observar el parpadeo infinito de un LED cada 500 milisegundos. Utilice el ciclo **DO-LOOP**

# DESAFÍO 11



Desarrollar un código que permita observar el parpadeo infinito de tres LEDs, en forma simultánea, cada un segundo. Utilice el ciclo **DO-LOOP**

# Ciclo GOTO

- Desarrolla un ciclo llamando a un proceso.
- El proceso a repetir debe tener etiqueta, para llamarla cuando sea necesario.
- El formato de programación se presenta a continuación:

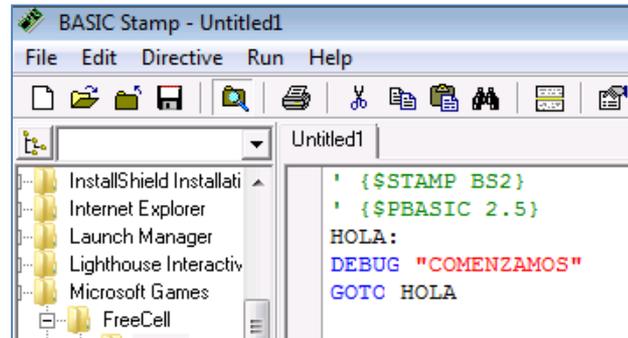
```
etiqueta:  
DEBUG "COMENTARIO"  
GOTO etiqueta
```

# Ciclo GOTO



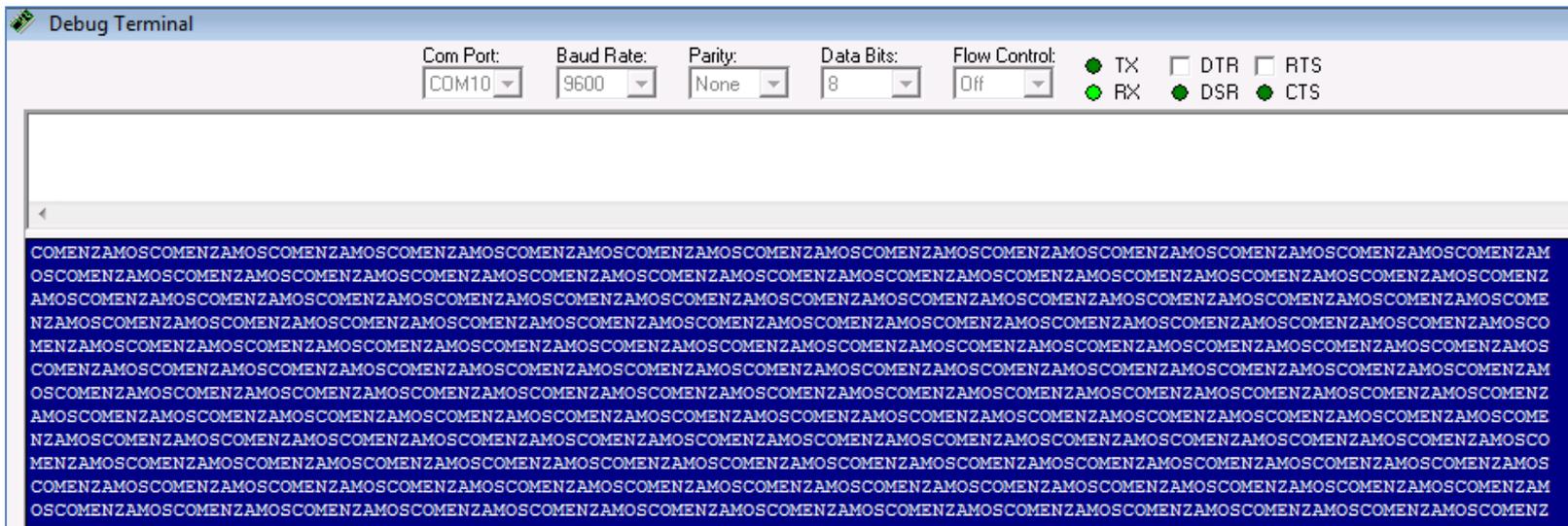
Desarrolla un código que muestre por pantalla **“COMENZAMOS”** infinitas veces.

# Ciclo GOTO



The screenshot shows the BASIC Stamp software interface. The title bar reads "BASIC Stamp - Untitled1". The menu bar includes "File", "Edit", "Directive", "Run", and "Help". The toolbar contains icons for file operations and execution. The left pane shows a file explorer with folders like "InstallShield Installati", "Internet Explorer", "Launch Manager", "Lighthouse Interactiv", "Microsoft Games", and "FreeCell". The main editor area contains the following code:

```
' {$STAMP BS2}
' {$PBASIC 2.5}
HOLA :
DEBUG "COMENZAMOS"
GOTO HOLA
```



The screenshot shows the Debug Terminal window. The title bar reads "Debug Terminal". The configuration bar includes:

- Com Port: COM10
- Baud Rate: 9600
- Parity: None
- Data Bits: 8
- Flow Control: Off
- TX:
- RX:
- DTR:
- DSR:
- RTS:
- CTS:

The terminal output area shows a continuous stream of the text "COMENZAMOS" repeated many times, indicating that the program is running in a loop.

# DESAFÍO 12



Desarrollar un código que permita observar el parpadeo infinito de un LED cada 500 milisegundos. Utilice el comando **GOTO**.

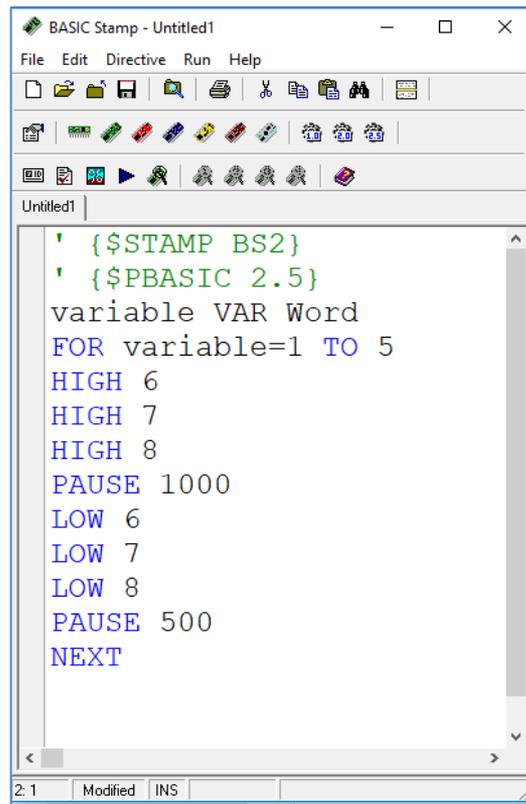
# DESAFÍO 13



Desarrollar un código que permita observar el parpadeo infinito de tres LEDs, en forma simultánea, cada un segundo. Utilice el comando **GOTO**.

# DESAFÍO 14

Describir la acción que ejecuta el siguiente programa:



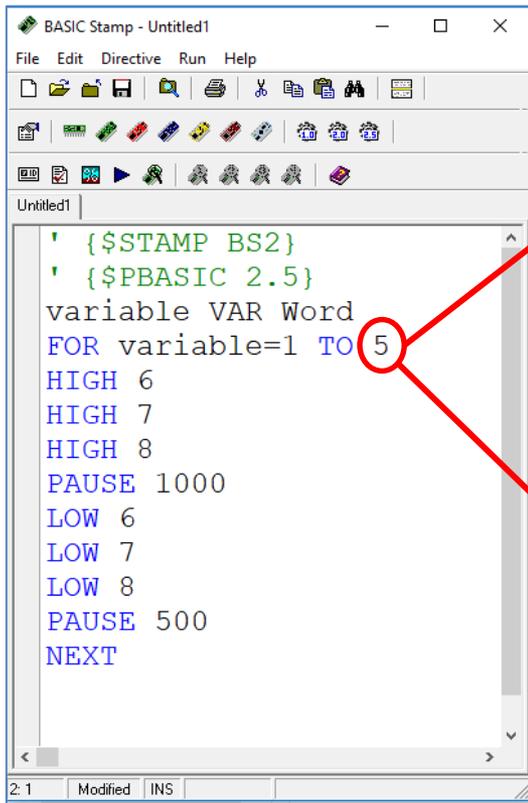
```

' {$STAMP BS2}
' {$PBASIC 2.5}
variable VAR Word
FOR variable=1 TO 5
HIGH 6
HIGH 7
HIGH 8
PAUSE 1000
LOW 6
LOW 7
LOW 8
PAUSE 500
NEXT

```

# DESAFÍO 15

Describir la acción que ejecuta el siguiente programa:



```

' {$STAMP BS2}
' {$PBASIC 2.5}
variable VAR Word
FOR variable=1 TO 5
HIGH 6
HIGH 7
HIGH 8
PAUSE 1000
LOW 6
LOW 7
LOW 8
PAUSE 500
NEXT

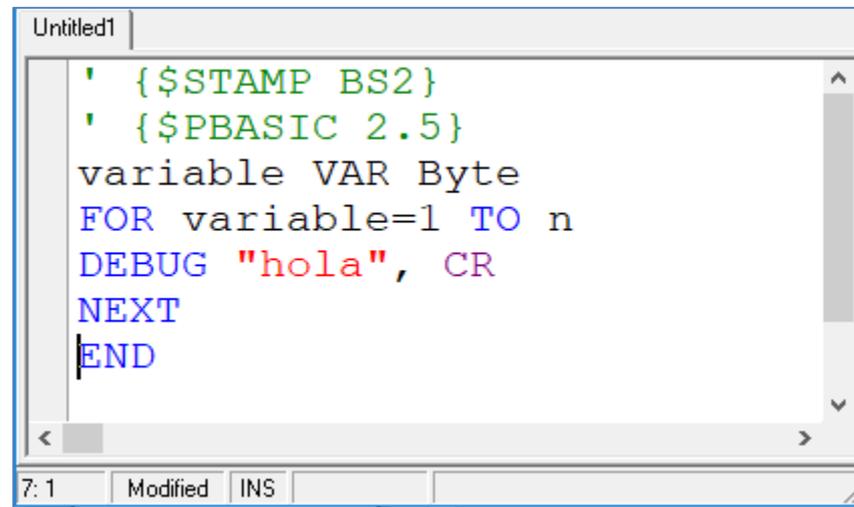
```

¿Qué ocurre con cuando cambio el 5 por 10?

¿Qué ocurre con cuando cambio el 5 por 20?

# Ciclo FOR-TO-NEXT

- No necesita contador.
- Necesita número de inicio y fin.
- Necesita definir la variable contador, con anterioridad.
- El formato de programación se presenta de la siguiente forma:



```
Untitled1
' {$STAMP BS2}
' {$PBASIC 2.5}
variable VAR Byte
FOR variable=1 TO n
DEBUG "hola", CR
NEXT
END
```

7: 1 Modified INS

# Ciclo FOR-TO-NEXT

```
Untitled1
' {$STAMP BS2}
' {$PBASIC 2.5}
hola VAR Word
FOR hola=1 TO 10
DEBUG "hola", CR
NEXT
END
```

9: 1 Modified INS

```
Debug Terminal
```

```
hola
```

¿Cuántas veces escribirá hola?

# DESAFÍO 16



Desarrollar una rutina que permita observar en menos de un minuto:

- El parpadeo de un LED, quince veces.
- El parpadeo de dos LEDs, cinco veces en forma simultánea.
- El parpadeo de tres LEDs, cinco veces en forma simultánea.

# DESAFÍO 17



Crear una rutina que permita hacer parpadear los tres LEDs, cinco veces, por separado.

# DESAFÍO 18



Desarrollar una rutina que encienda tres LEDs en forma simultánea, donde sólo uno parpadee tres veces y luego se apaguen todos al mismo tiempo.

# DESAFÍO 19

Describir la acción que ejecutan los actuadores LED durante la siguiente programación:

```
Untitled1
' {$STAMP BS2}
' {$PBASIC 2.5}
HIGH 6
HIGH 7
HIGH 8
PAUSE 500
LOW 7
LOW 8
PAUSE 500
HIGH 7
HIGH 8
PAUSE 500
LOW 7
LOW 8
PAUSE 500
HIGH 7
HIGH 8
PAUSE 500
LOW 7
LOW 8
LOW 6
```

21: 18 Modified INS

# DESAFÍO 20



Crear un código que permita mostrar por pantalla una historia de cinco líneas y que cuando cada línea aparezca, encienda un led distinto al anterior en cada línea mostrada en pantalla.

Unidad 2.1 - Programación de Actuadores - LED

# Robótica Educativa

Robot T-17

