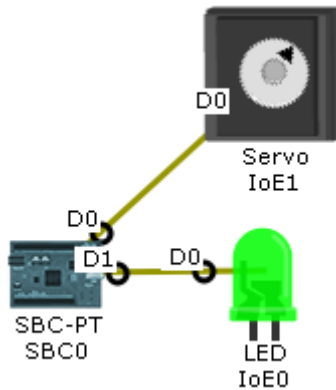


## Packet Tracer: simulación de dispositivos de IoT

### Topología



### Objetivos

#### Parte 1: armar el circuito

- Coloque los componentes en el espacio de trabajo lógico
- Conecte los componentes

#### Parte 2: programar la computadora de una sola placa (SBC)

- Ejecute el programa predeterminado
- Modifique el programa predeterminado

### Aspectos básicos/situación

Packet Tracer se desarrolló para simular los dispositivos de IoT. Este tutorial lo guiará por el proceso de colocar componentes en el espacio de trabajo lógico, conectar los componentes y, luego, programar la computadora de una sola placa (SBC) para controlarlos.

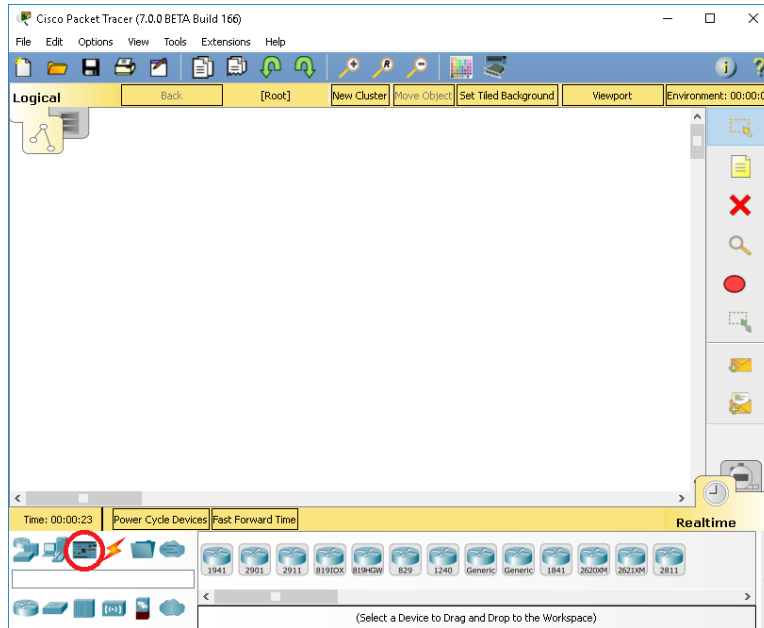
### Recursos necesarios

- PC con Packet Tracer versión 7.0 o posterior instalado

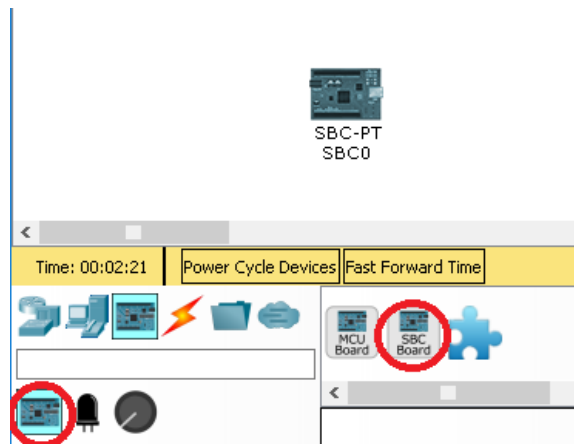
## Parte 1: Desarrollo del circuito

### Paso 1: Coloque los componentes en el espacio de trabajo lógico.

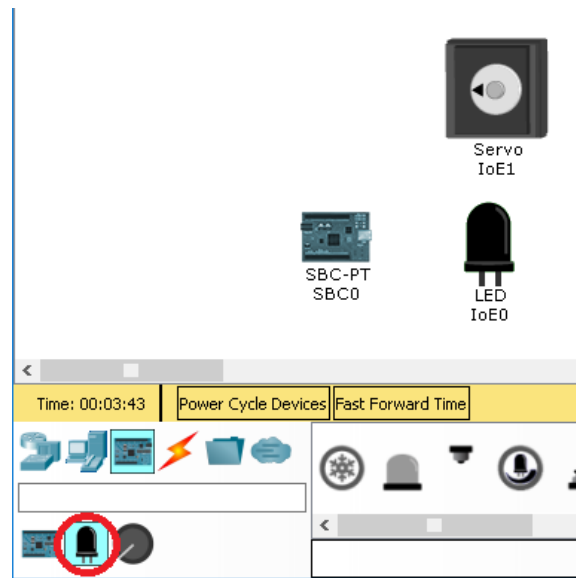
- Abra **Packet Tracer 7.0** o posterior y seleccione el icono **Componentes**.



- Coloque una **placa de SBC** en el **espacio de trabajo lógico**.

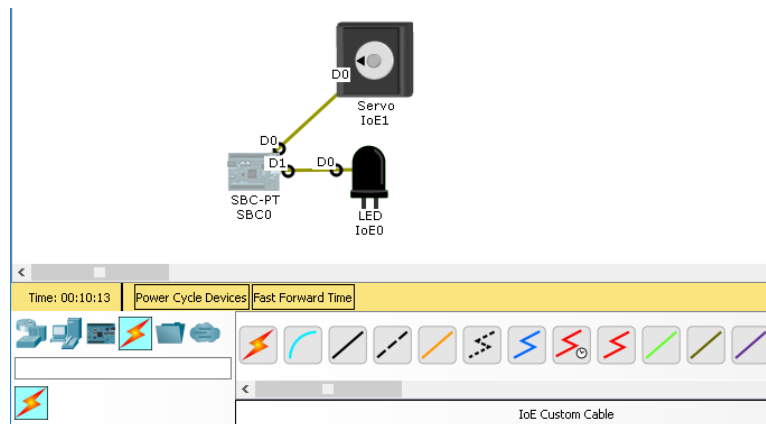


- c. Coloque un **LED** y un **servomotor** en el **espacio de trabajo lógico**.



## Paso 2: Conecte los componentes.

- Haga clic en el icono **Conexiones** y seleccione un **cable personalizado de IoT**; conecte **SBC0 D0** a **Servo0 D0**.
- Seleccione otro **cable personalizado de IoT** y conecte **SBC0 D1** a **LED D0**.

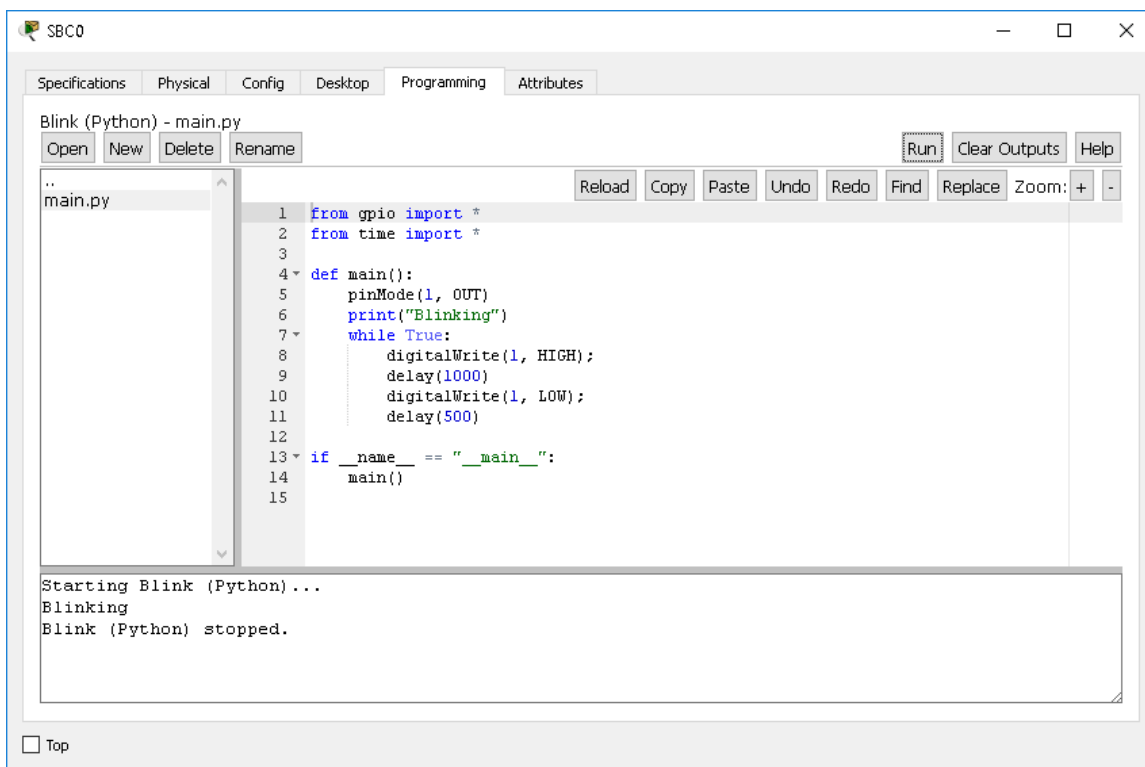


## Parte 2: Programar la computadora de una sola placa (SBC)

### Paso 1: Ejecute el programa predeterminado.

- Haga doble clic en **SBC0** y seleccione la ficha **Programación**.
- Haga doble clic en **Parpadeo (Python)** en el panel izquierdo para abrirlo.

- c. Haga doble clic en **main.py** para revelar el código predeterminado de Python.



- d. Haga clic en el botón **Ejecutar** para ejecutar el código predeterminado. Regrese al espacio de trabajo lógico. El LED debe destellar.
- e. Regrese a la ficha de Programación SBC0 y haga clic en el botón **Detener** para detener la ejecución de programas.

### Paso 2: Modifique el programa predeterminado.

- a. Copie la línea 8 del código fuente y péguelo a continuación. Haga lo mismo con la línea 11 (antes línea 10).
- b. Modifique las líneas nuevas de código a fin de que se lea:

```
customWrite(0, 127);
```

y

```
customWrite(0, -127);
```

Reload Copy

```
1  from gpio import *
2  from time import *
3
4  def main():
5      pinMode(1, OUT)
6      print("Blinking")
7      while True:
8          digitalWrite(1, HIGH);
9          customWrite(0, 127);
10         delay(1000)
11         digitalWrite(1, LOW);
12         customWrite(0, -127);
13         delay(500)
14
15  if __name__ == "__main__":
16      main()
17
```

c. ejecute el programa modificado. El servomotor ahora debe moverse junto con el LED parpadeante.

## Reflexión

¿Qué podría modificarse para que el servomotor active las direcciones opuestas mientras el LED parpadea?

---

---