

AG | Blog

El blog plasma la innovación tecnológica que ingresamos al país, bajo una perspectiva practica que permita su implementación instantanea.

Microcontrolador Raspberry Pi PICO

EN 30 ABRIL, 202130 ABRIL, 2021 / POR BENJIMFLORES

Como todos sabemos, Raspberry Pi es una marca muy conocida a nivel mundial por sus famosos mini ordenadores que desde el 29 de febrero de 2012 no han parado de venderse. Sin embargo, en la actualidad tiene otros proyectos interesantes como la creación de sus propios microcontroladores, pues a principios de este año dio a conocer un microcontrolador que ellos diseñaron, el cual esta basado en la arquitectura de procesadores **ARM Cortex-M0+** y se le dió el nombre de **RASPBERRY Pi PICO**.

Por lo que en esta ocasión veremos cuales son las cualidades de esta nueva tarjeta y porque se ha posicionado como una de las favoritas en tan poco tiempo.

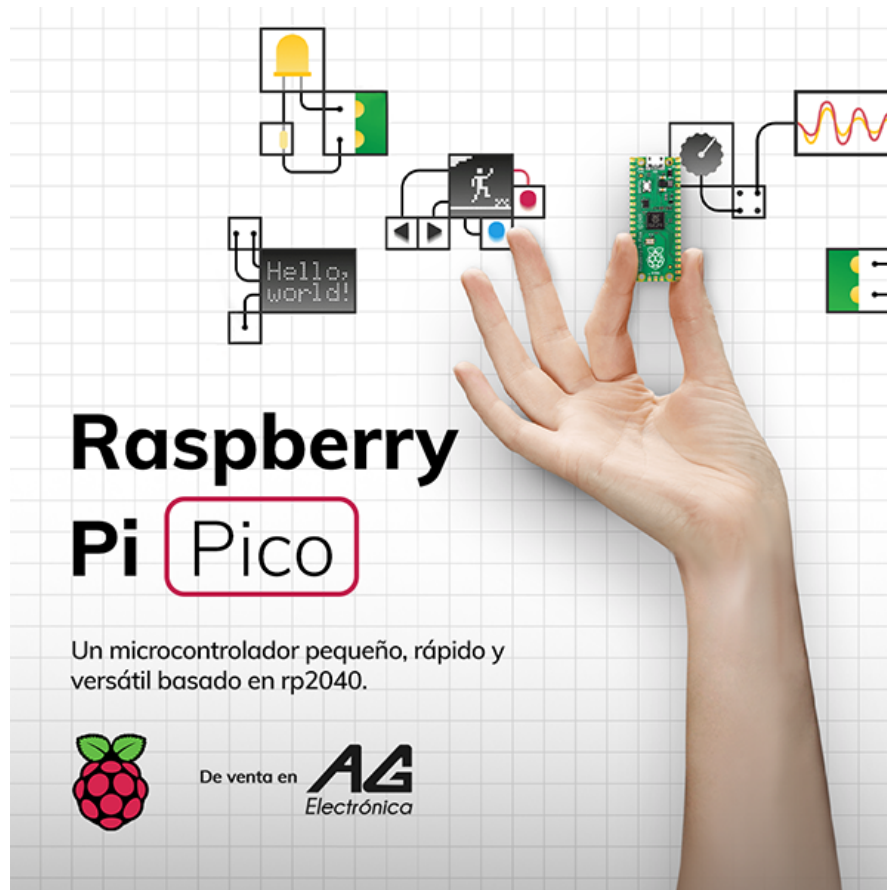
Conociendo Raspberry Pi PICO

Raspberry Pi P (<https://bit.ly/3aLMcoe>) **ICO** (<https://bit.ly/3aLMcoe>) es una maravilla en miniatura que pone en la palma de la mano tecnología que sustenta casi todo tipo de aplicaciones, desde sistemas básicos de domótica hasta la automatización de ambientes mas complejos de la industria. Ya sea que se esté buscando aprender sobre el lenguaje de programación MicroPython, dar los primeros pasos en computación física o construir un proyecto de hardware, Raspberry Pi Pico y su increíble comunidad apoyará en cada paso del camino.

Aunque el tamaño de esta tarjeta es similar al de una barra de goma de mascar, **Raspberry Pi P** (<https://bit.ly/3aLMcoe>) **ICO** (<https://bit.ly/3aLMcoe>) tiene una sorprendente cantidad de potencia gracias al chip ubicado en la parte central: el microcontrolador RP2040.

Esta tarjeta no está diseñada para reemplazar a la tradicional Raspberry Pi, ya que es una clase diferente de dispositivo conocida como **computadora de placa única**, dado que puede usar su Raspberry Pi para jugar, escribir historias y navegar por la web. Por otra parte, Raspberry Pi PICO **está diseñada para proyectos de computación física en los que controla cualquier cosa, desde LED y botones hasta sensores, motores e incluso otros microcontroladores.**

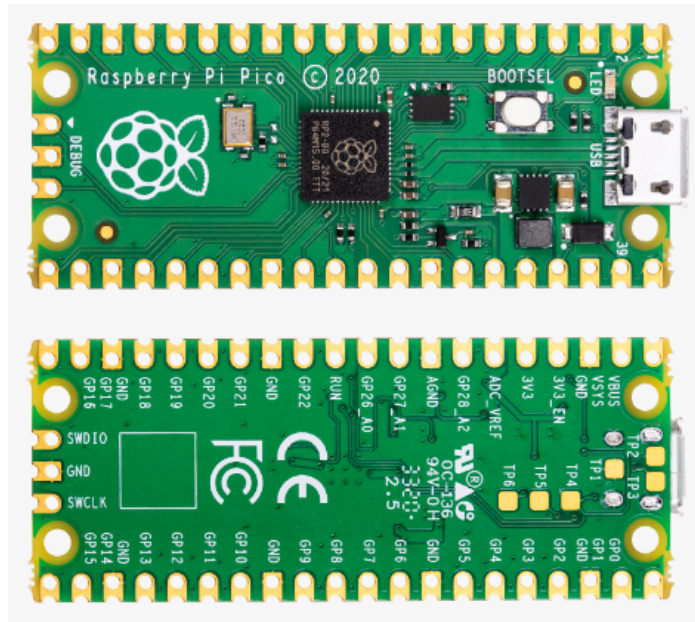
<https://www.facebook.com/AGElectronicsMexico/> (<https://www.facebook.com/AGElectronicsMexico/>) (<https://www.facebook.com/AGElectronicsMexico/>) (<https://www.facebook.com/AGElectronicsMexico/>) (<https://www.facebook.com/AGElectronicsMexico/>) (<https://www.facebook.com/AGElectronicsMexico/>) (<https://www.facebook.com/AGElectronicsMexico/>) (<https://www.facebook.com/AGElectronicsMexico/>) (<https://www.facebook.com/AGElectronicsMexico/>) (<https://www.facebook.com/AGElectronicsMexico/>) (<https://www.facebook.com/AGElectronicsMexico/>)



Especificaciones

Raspberry Pi PICO (*"PICO"* para abreviar), tiene pin headers muy similares a los que componen las entradas y salidas de propósito general (GPIO) de una Raspberry Pi, pero mientras que la mayoría de las computadoras Raspberry Pi vienen con los pin headers metálicos físicos ya conectados, RPi PICO no lo hace. La razón de esto es porque nos dan la libertad de colocar pin headers macho o hembra, rectos o de ángulo (dependiendo de la necesidad de cada proyecto).

Es claro que al ser un microcontrolador no va a tener algún tipo de periférico de computadora como USB host, puerto Ethernet o HDMI, pero si tiene un conector micro-USB que nos permitirá conectar nuestra *PICO* a la PC para programarla así como se observa en la siguiente imagen.



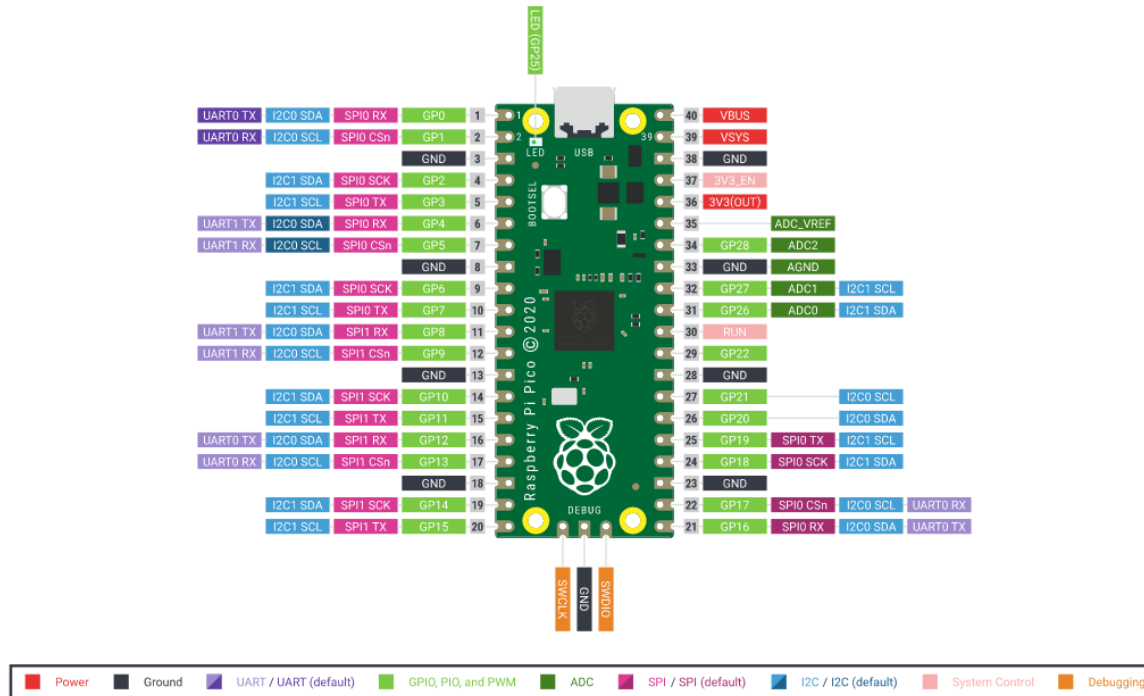
Vista superior e inferior de RPI PICO.

Características técnicas

Ya que hemos visto de manera superficial lo que contiene nuestra *PICO*, a continuación detallamos algunas características que todo aquel que empiece a utilizar la tarjeta debe conocer para su correcto funcionamiento, así como la definición de cada uno de los pines para una mayor comprensión del microcontrolador.

- **Chip del microcontrolador:** RP2040 diseñado por Raspberry pi en Reino Unido
- **Memoria Flash:** 2MB
- **Frecuencia de reloj:** 133 MHz
- **SRAM:** 264K
- **Procesador:** ARM Cortex-M0+
- **Interfaces de comunicación:** 2-UART, 2-SPI, 2-I2C, 16-PWM
- **Temperatura (C):** -20° a 85°
- **Voltaje de operación:** 1.8–5.5VCD
- **Puertos de entrada y/o salida digitales (GPIO):** 26

- **ADC:** 3 Canales – 12bits
- **Programación:** Drag-and-drop usando almacenamiento masivo a través de USB
- **Modos:** reposo e inactivo para un bajo consumo
- **Sensor de temperatura interno**



Pines
Rpi
PICO

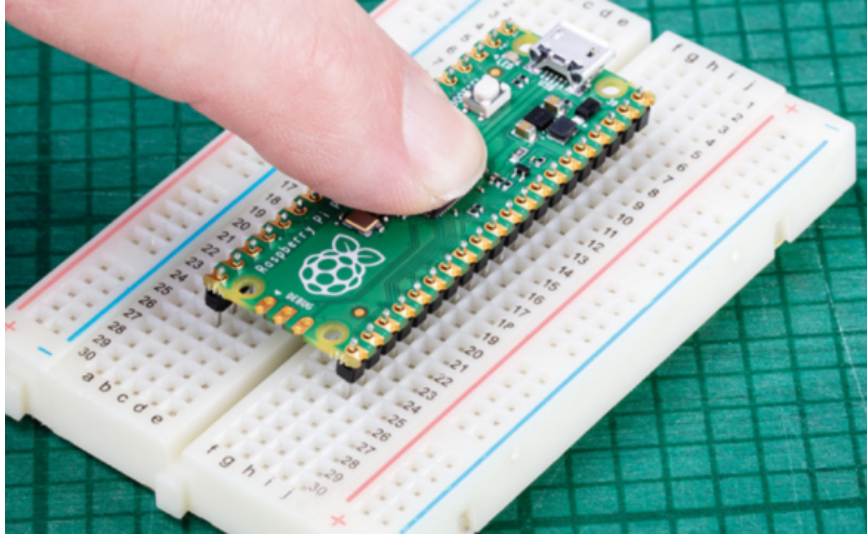
Soldando los Pin Headers

Al obtener nuestra [Raspberry Pi PICO](https://bit.ly/3aLMcoe) (<https://bit.ly/3aLMcoe>), notaremos que NO viene con pin headers hembra o macho como sucede en los mini ordenadores Raspberry Pi. Sin embargo, la forma más fácil de usar la *PICO* es montarla en una protoboard. Para ello, antes se deberán soldar los pin headers utilizando un cautín, algo de soldadura, una esponja de limpieza y dos tiras de pin headers macho de 20 pines de 2,54 mm.

Si ya tiene una protoboard, puede usarla para facilitar el proceso de soldadura como se ve en la imagen inferior.

A veces, los pin headers de 2,54 mm se proporcionan en tiras de más de 20 pines, si los que tiene son más largos, simplemente cuente 20 pines desde un extremo y mire el plástico entre los pines 20 y 21, verá que tiene una pequeña muesca a cada lado. Este es un punto de ruptura: coloque los headers en ambas manos izquierda y derecha y doble la tira, se romperá limpiamente, dejándolo

con una tira de exactamente 20 pines. Si la banda de encabezado restante tiene más de 20 pines, vuelva a hacer lo mismo para tener dos bandas de 20 pines.



Protoboard
de apoyo
para
soldar los
pin
headers de
su PICO.

Después de soldar los pin headers con ayuda del cautín y soldadura convencional de 1mm 0.8mm, ya podemos empezar a desarrollar nuestros proyectos como lo haríamos con cualquier otro microcontrolador, solo recordemos tener listo un cable mini USB para su conexión a la computadora. Cabe mencionar que es compatible con los sistemas operativos mas comerciales como lo son Windows, Mac o Linux.

Programación

A continuación explicaremos el proceso para empezar a programar nuestra PICO, cabe mencionar que se puede realizar con dos tipos de lenguaje de programación como son: C++ y MicroPython.

En esta ocasión, utilizaremos el código de programación en MicroPython que contiene el núcleo principal de instrucciones de Python y una adecuación de códigos para manipular el hardware de nuestro microcontrolador.

Primero, necesitaremos descargar el firmware para la programación de PICO en MicroPython, ya que por default cuando se adquiere la tarjeta PICO no es posible programarla directamente. Para ello nos dirigimos a la pagina de Raspberry Pi en el siguiente link:

<https://www.raspberrypi.org/documentation/rp2040/getting-started/#getting-started-with-micropython>
(<https://www.raspberrypi.org/documentation/rp2040/getting-started/#getting-started-with-micropython>).

Ya en el sitio oficial, daremos scroll hasta la parte donde nos aparezca las pestañas descritas en la siguiente imagen para descargar el firmware en C++ o MicroPython, aunque como mencionamos anteriormente, el ejemplo que realizaremos será con MicroPython.



Pestaña
Micro
Python
para la
descarga
del
respectivo
firmware.

Una vez en la pestaña, procedemos a realizar unos sencillos pasos como a continuación describimos textualmente y con la imagen gif al inferior:

- 1.- Dar clic en un botón verde que dice “**Download UF2 file**” para instalar MicroPython más fácilmente.
- 2.- Mantenga presionado el botón “BOOTSEL” y conecte su RPi PICO al puerto USB de su Raspberry Pi u otra computadora. Suelte el botón “BOOTSEL” después de que su Pico esté conectada.
- 3.- Se montará como un dispositivo de almacenamiento masivo llamado RPI-RP2.
- 4.- Arrastre y suelte el archivo MicroPython **UF2** en el volumen RPI-RP2. Su RPi Pico se reiniciará.
- 5.- Ahora está ejecutando MicroPython y puede empezar a programar.

MicroPython

Ahora que ya tenemos la posibilidad de empezar a crear nuestros proyectos, algunos podríamos preguntarnos ¿Qué es MicroPython? y prácticamente es una implementación completa del lenguaje de programación Python versión 3 que se ejecuta directamente en hardware integrado como el caso de Raspberry Pi PICO. Obtiene un indicador interactivo (el REPL) para ejecutar comandos inmediatamente a través de USB Serial y un sistema de archivos incorporado.

Para el caso de nuestra RPi PICO, MicroPython incluye módulos para acceder a hardware específico del chip a bajo nivel y no solo de RPi PICO, sino algunos otros microcontroladores conocidos como Nodemcu (<https://www.agelectronica.com/index.php?n=NODEMCU+LUA-WIFI>) o microbit (<https://www.agelectronica.com/detalle.php?p=MB80-US>) por mencionar algunos.

Si deseas conocer más sobre el tema, te invitamos a leer los siguiente artículos:

Introducción a la electrónica con micro:bit (<https://agelectronica.blog/2019/07/09/introduccion-a-la-electronica-con-microbit/>)

Programando NodeMCU con MicroPython (<https://agelectronica.blog/2019/08/14/programando-nodemcu-con-micropython/>)

Hola Mundo!

Es bien sabido en el mundo de los microcontroladores o la programación en general, que el primer programa para testear que todo funcione bien su nombre es “***Hola Mundo!***” y en PICO (<https://bit.ly/3aLMcoe>) no es la excepción, por lo que a continuación mostraremos como llevarlo a cabo.

Existen muchas herramientas que nos permitirán programar nuestra RPi PICO, pero la mas recomendable es ***Thonny*** ya que es un IDE que desarrollo la fundación Raspberry Pi, el cual es de código abierto y cuenta con optimizaciones para instalarse en todo sistema operativo incluyendo en un mini ordenador Raspberry Pi.

Da click aquí (<https://thonny.org/>), para descargar esta herramienta.

En la siguiente imagen, podremos ver rápidamente su apartado de descargas y como luce el IDE al abrir el link.

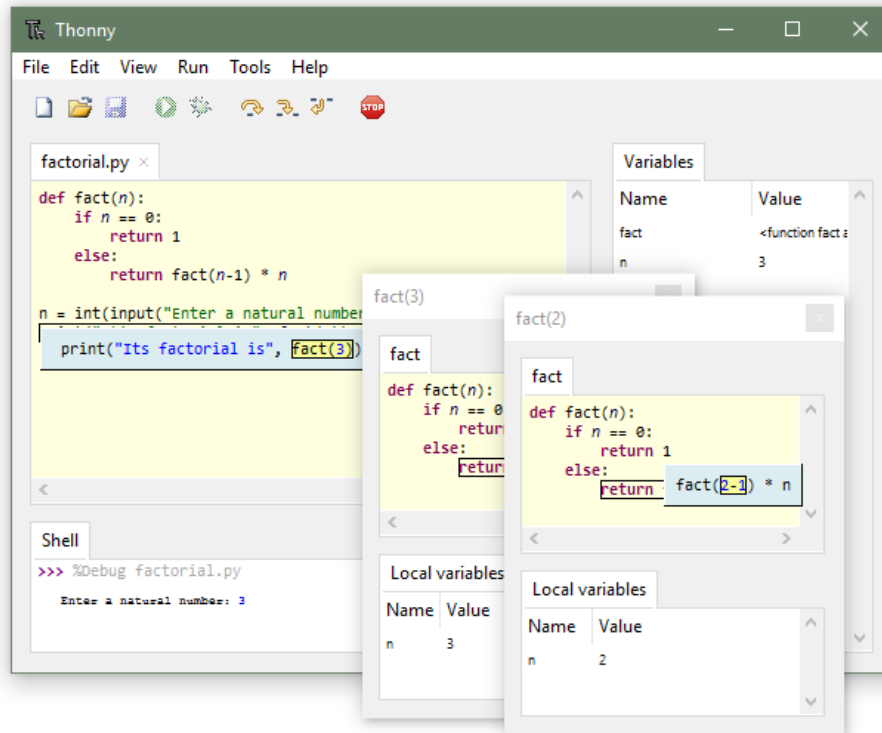
Thonny

Python IDE for beginners



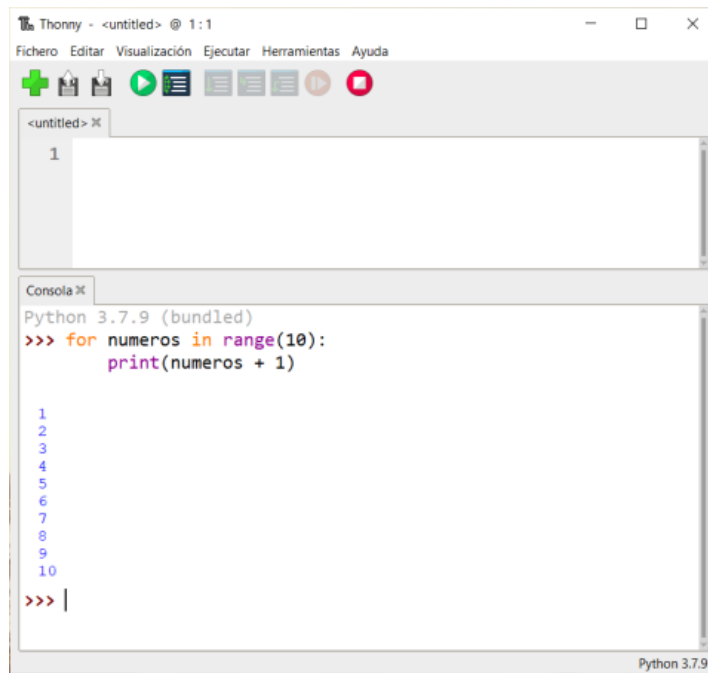
Download version **3.3.6** for
[Windows](#) • [Mac](#) • [Linux](#)

NB! On Windows you may receive a warning dialog from Defender. Just click "More info" and "Run anyway".



Página principal del IDE Thonny.

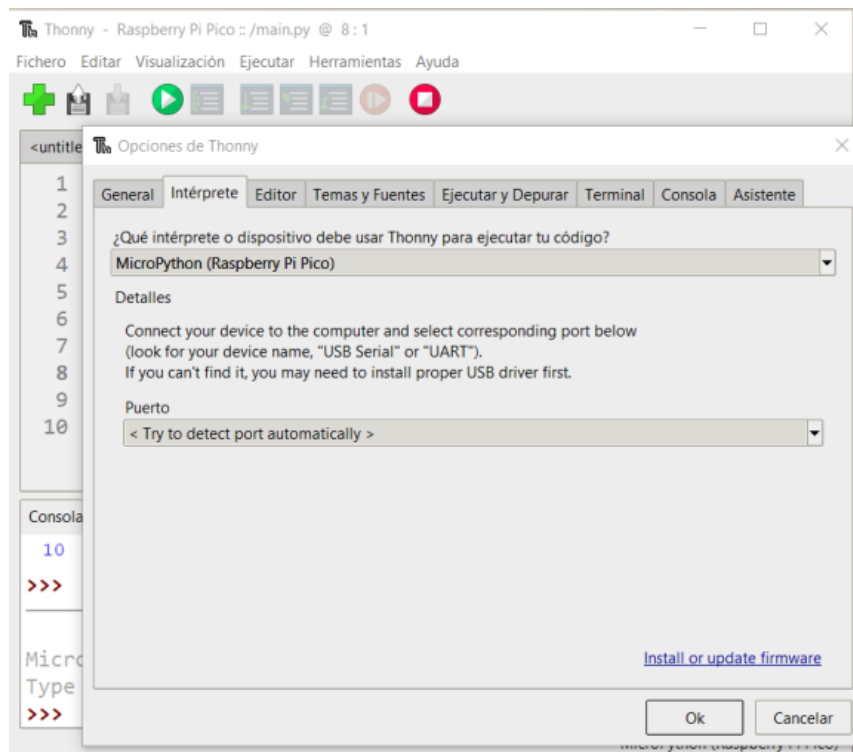
Una vez que se descargó el IDE procedimos a instalarlo, para ello solo bastará con dar click derecho, elegir instalar y dar siguiente como cualquier programa habitual. En este caso debemos tener también conectada nuestra RPi PICO a la PC para que el IDE la reconozca en automático y logremos programarla, de esta manera si todo salió bien deberíamos tener el IDE dividido en dos pantallas que son de arriba hacia abajo, la primera donde se estará creando el código que se cargará a la memoria flash de la RPi PICO y la segunda pantalla (la de abajo), es para el REPL es decir un intérprete de MicroPython para ejecutar código sin tener que cargarlo aún a la PICO.



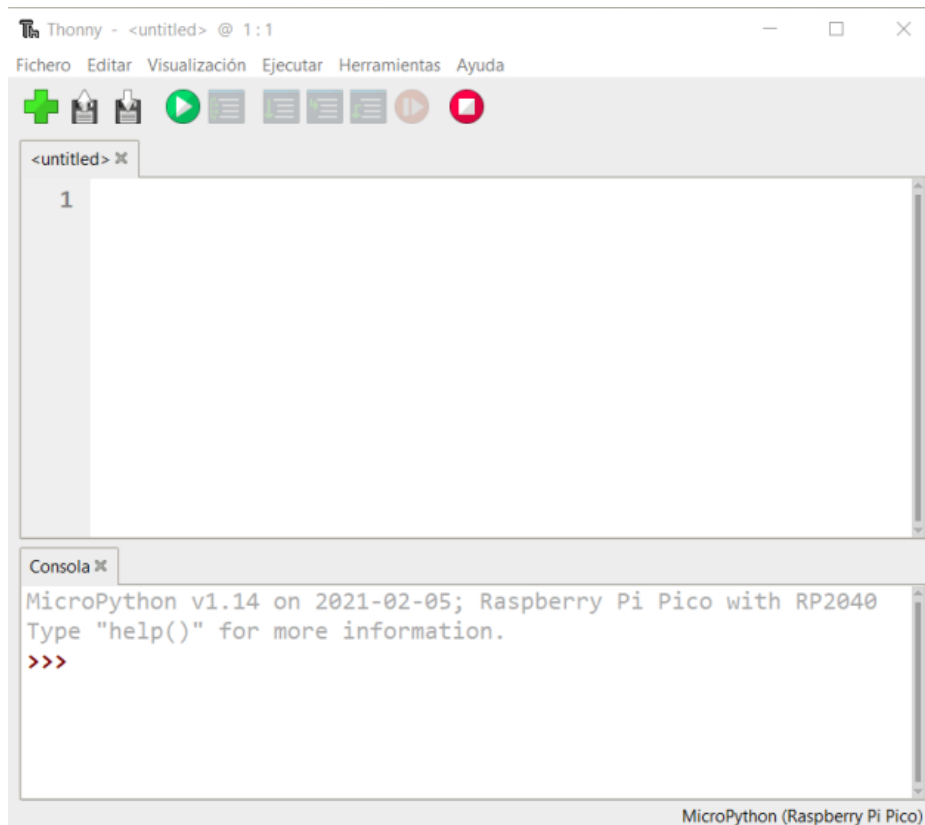
Por default el IDE viene configurado para actuar como un interprete del lenguaje Python en general, por lo que necesitamos cambiar este interprete por el de PICO.

Para ello, nos dirigimos a la barra de menú y seleccionamos herramientas -> opciones y aparecerá una ventana emergente con un par de pestañas de las cuales seleccionaremos "Intérprete".

Ahora tendremos una lista de opciones de interpretes por lo que elegiremos la opción marcada como **MicroPython (Raspberry Pi Pico)**, también podemos elegir el puerto serial COM de nuestra PICO pero es preferible dejar la opción como < Try detect port automatically > para siempre lo seleccione por nosotros como vemos en la siguiente imagen.



Una vez realizada toda esta configuración, podremos observar que en la ventana de REPL aparece un mensaje señalando que se está usando ahora el interprete de PICO para el microcontrolador RP2040 como se ve a continuación.



IDE
Thonny
con
interprete
RPI PICO.

Finalmente en la ventana de edición del código, escribimos un programa para hacer el “Hola mundo!” que en hardware es encender y apagar un led.

El led de color verde que ya viene conectado por default en nuestra tarjeta PICO en el GPIO25, el código se muestra en la siguiente imagen junto con comentarios que explican el mismo, después de escribir el código solo hay que dar click en el botón verde con el triangulo blanco para subir el programa a nuestra PICO.



Thonny - Raspberry Pi Pico :: /main.py @ 15:17

Fichero Editar Visualización Ejecutar Herramientas Ayuda

[main.py] x

```
1 #Hola mundo - blynk RPI PICO
2 #importamos de la libreria machine la clase Pin y Timer
3 from machine import Pin, Timer
4
5 #Implementamos la clase Pin a la variable led
6 #Pasando el GPIO 25 como salida
7 #El GPIO 25 tambien tiene conectado un LED
8 led = Pin(25, Pin.OUT)
9 #Empleamos la clase timer
10 tim = Timer()
11
12 #Creamos una funcion para conmutar el estado del led
13 def tick(timer):
14     global led
15     led.toggle()
16
17 #Finalizamos implementando un timer para que repita
18 #la funcion de form periodica cada segundo
19 tim.init(freq=1, mode=Timer.PERIODIC, callback=tick)
```

Consola x

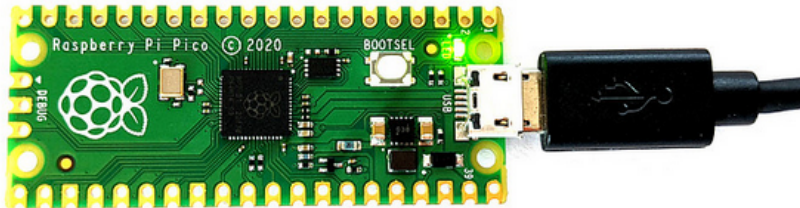
```
MicroPython v1.14 on 2021-02-05; Raspberry Pi Pico with RP2040
Type "help()" for more information.
>>> %Run -c $EDITOR_CONTENT
>>>
```

MicroPython (Raspberry Pi Pico)

Captura
de

pantalla
de la
ejecución
del
programa
en
nuestra
PICO.

De esa forma veríamos que el led que viene sobre la placa PICO (como se puede observar en la imagen inferior) empezará a encender y apagar a cada segundo, con esto presente podremos empezar a integrar más componentes a nuestra protoboard y realizar proyectos mas complejos con MicroPython, además de ser un lenguaje de programación con una curva de aprendizaje muy rápida.



Prueba de la
conmutación de
led de
encendido y
pagado
indefinidamente.

Si estás interesado en conocer más sobre este u otro tema no dudes en escribirnos en los comentarios y seguirnos en nuestras redes sociales para conocer más de las tecnologías mas innovadoras del mundo de la electrónica.

Adquiere tu RASPBERRY Pi PICO aquí:
<https://bit.ly/3aLMcoe> (**<https://bit.ly/3aLMcoe>**)

Encuentra nuestras sucursales y horarios de atención aquí:
<https://bit.ly/38fRSG> (**<https://bit.ly/38fRSGe>**)

¡SÍGUENOS EN NUESTRAS REDES SOCIALES!



AG Electrónica



@ag_electronica



AG Electrónica

◀ [CÓDIGO ABIERTO](#) ▶ [ELECTRÓNICA](#) ▶ [INGENIERÍA](#) ▶ [MICROCONTROLADOR](#) ▶ [MICROCONTROLADORES](#) ▶ [PICO](#) ▶ [PROGRAMACIÓN](#) ▶ [RASPBERRY](#) ▶ [RASPBERRY](#)
PI ▶ [RASPBERRY PI PICO](#) ▶ [RASPI](#)

[BLOG DE WORDPRESS.COM.](#)