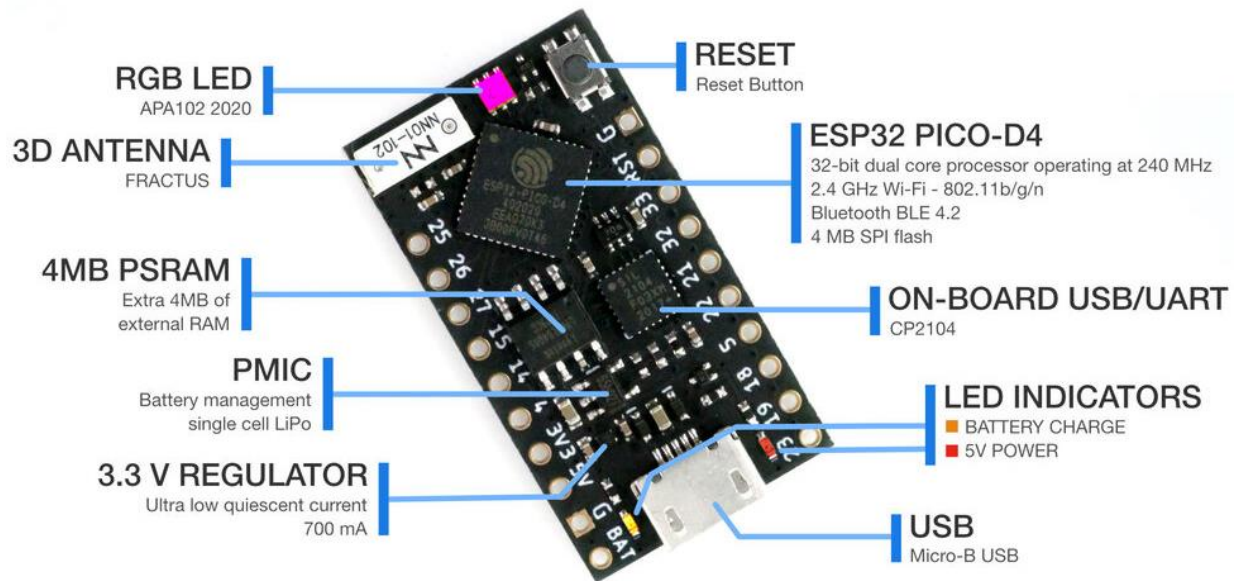


Getting Started



CONGRATULATIONS!

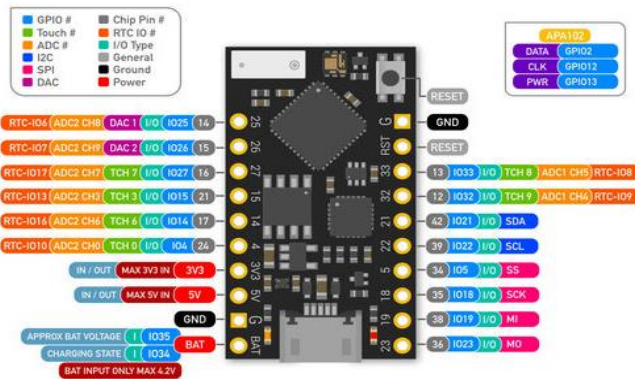
If you are visiting this page, it's likely you are a proud owner of a TinyPICO - congratulations!

If you don't already own a TinyPICO, there are bunch of different places around the world you can grab one from...

We've put together this getting started guide to help you get familiar with your new TinyPICO, and showcase some of the main features of the board - including an FAQ covering some of the questions we were asked during the crowd funding campaign.

MAIN FEATURES

Your TinyPICO is pretty spec'd up! We've taken a lot of time and care to make it small, performant and as powerful and useful as possible, given our self imposed size constraints ;)



- 32Bit Dual Core 240MHz
 - 2.4GHz Wifi - 802.11b/g/n
 - Bluetooth BLE 4.2
 - 4MB SPI Flash
 - 4MB Extra PSRAM
 - APA102 RGB LED
 - USB + Serial/UART for programming
 - 700mA 3.3V LDO Regulator
 - L1PO Battery Management
 - Optimised power path for low power battery usage
 - JST pads on the bottom supports PH & MicroBlade
 - 14x GPIO broken out
 - Breadboard friendly
 - Oh, It's tiny! Just 18mm x 32mm
- ESP32 PICO D4
MICROPYTHON
ESPRESSIF IDF
ARDUINO IDE

Although the TinyPICO uses a PICO-D4 chip from Espressif, it's still the same powerful dual core ESP32 that you already might know and love. Just way smaller than you might be used to using!

That means you can do anything on the TinyPICO that you can do on any other ESP32 development board, within the constraints of the 14 GPIO exposed.

There are already hundreds of libraries and tutorials and videos online that cover everything from using an ESP32 to play music, serve web pages, output vision on a VGA screen, connect to RGB LED matrix panels and so much more. It's a really well supported platform, that is growing even better every day.

HOOKIN IT UP

All you need to do to get started is to plug your TinyPICO into your computer using a Micro-B USB cable.

Your TinyPICO ships with MicroPython pre-installed and has some hello world code on it that runs when you power it up.

Depending on when your TinyPICO was packed during the campaign, you will see one of 2 visual displays on the onboard RGB LED

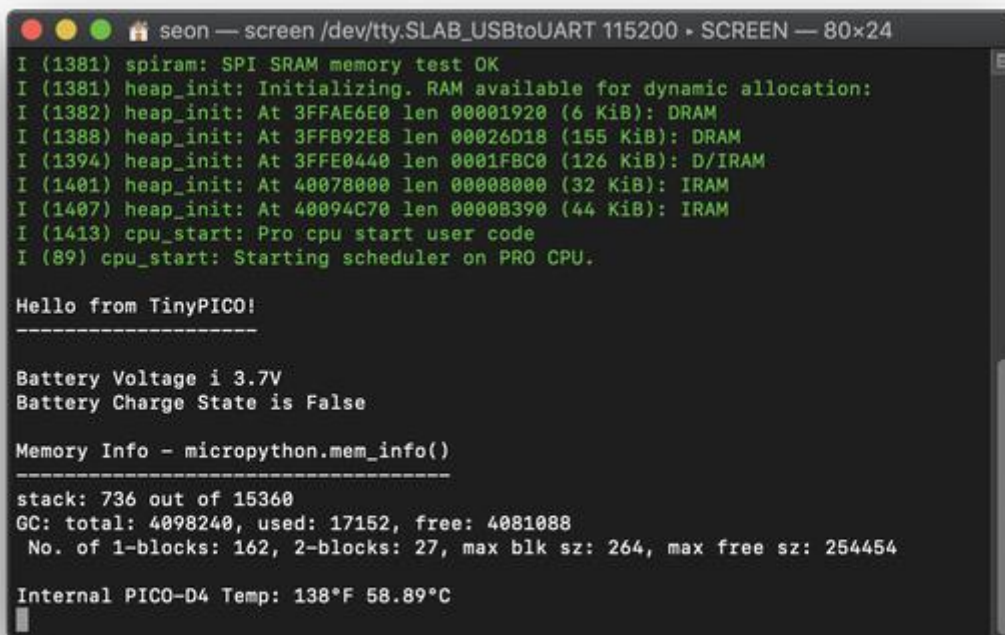
A CYCLING RAINBOW

or

3 FLASHES OF GREEN, THEN A CYCLING RAINBOW

The 3 flashes in green are a RAM test result, indicating the extra PSRAM is working correctly. This extra visual indicator was added later in production to improve my testing speed on boards as they were being packed.

If you open a serial terminal with the TinyPICO plugged in, you will see some hello world info displayed after it boots. Here is an example of the output running on a Mac using **screen** in a terminal window:



```
seon — screen /dev/tty.SLAB_USBtoUART 115200 - SCREEN — 80x24
I (1381) spiram: SPI SRAM memory test OK
I (1381) heap_init: Initializing. RAM available for dynamic allocation:
I (1382) heap_init: At 3FFAE6E0 len 00001920 (6 KiB): DRAM
I (1388) heap_init: At 3FFB92E8 len 00026D18 (155 KiB): DRAM
I (1394) heap_init: At 3FFE0440 len 0001FBC0 (126 KiB): D/IRAM
I (1401) heap_init: At 40078000 len 00008000 (32 KiB): IRAM
I (1407) heap_init: At 40094C70 len 0000B390 (44 KiB): IRAM
I (1413) cpu_start: Pro cpu start user code
I (89) cpu_start: Starting scheduler on PRO CPU.

Hello from TinyPICO!
-----

Battery Voltage i 3.7V
Battery Charge State is False

Memory Info - micropython.mem_info()
-----
stack: 736 out of 15360
GC: total: 4098240, used: 17152, free: 4081088
No. of 1-blocks: 162, 2-blocks: 27, max blk sz: 264, max free sz: 254454

Internal PICO-D4 Temp: 138°F 58.89°C
```

The output will continue to display the PICO-D4 internal temperature every few seconds just to show you continued output so you know the TinyPICO is present.

MICROPYTHON

As mentioned above, your TinyPICO has MicroPython installed on it already, and the version it's been flashed with is now v1.11 from the mainline repository + a bunch of fixes and improvements that have been implemented since v1.11 was released.

If you would like to update the version of MicroPython on your TinyPICO, you can follow the steps in the RE-INSTALLING MICROPYTHON on our [coding examples page](#).

RE-INSTALLING OR UPDATING MICROPYTHON

Even though your TinyPICO ships with MicroPython installed, you can just connect it to the Arduino IDE and flash it with C++ code. If you do that, MicroPython will be erased from your board in the process of flashing it, so what if you want to get MicroPython back on there?

Or what happens if you just want to update to a newer MicroPython firmware?

There are 2 ways to get MicroPython onto your TinyPICO (or any other ESP32 development board):

INSTALL THE LATEST PRE-BUILT FIRMWARE

You can [download](#) and deploy the latest MicroPython firmware for your TinyPICO.

NOTE: *The TinyPICO (ESP32) uses a specific set of command line parameters to flash MicroPython - check the download link above for the correct **esptool** command.*

COMPILE MICROPYTHON FROM SOURCE

If you are adventurous, you can download the Espressif IDF and MicroPython source and [compile it yourself](#).

NEW TO MICROPYTHON?

I made some videos about what MicroPython is and how to use it on my [YouTube channel](#). Here is the first video in the playlist:

[MicroPython #1 - Lets Get Started](#)

If you are just looking for a way to get your code onto your TinyPICO, there are lots of ways to tackle this on the command line with apps like [rshell](#) & [Ampy](#) , but the [code with mu](#) IDE is one of the easiest, so I made a video about that too!

[MicroPython Made Easier!](#)

ARDUINO IDE

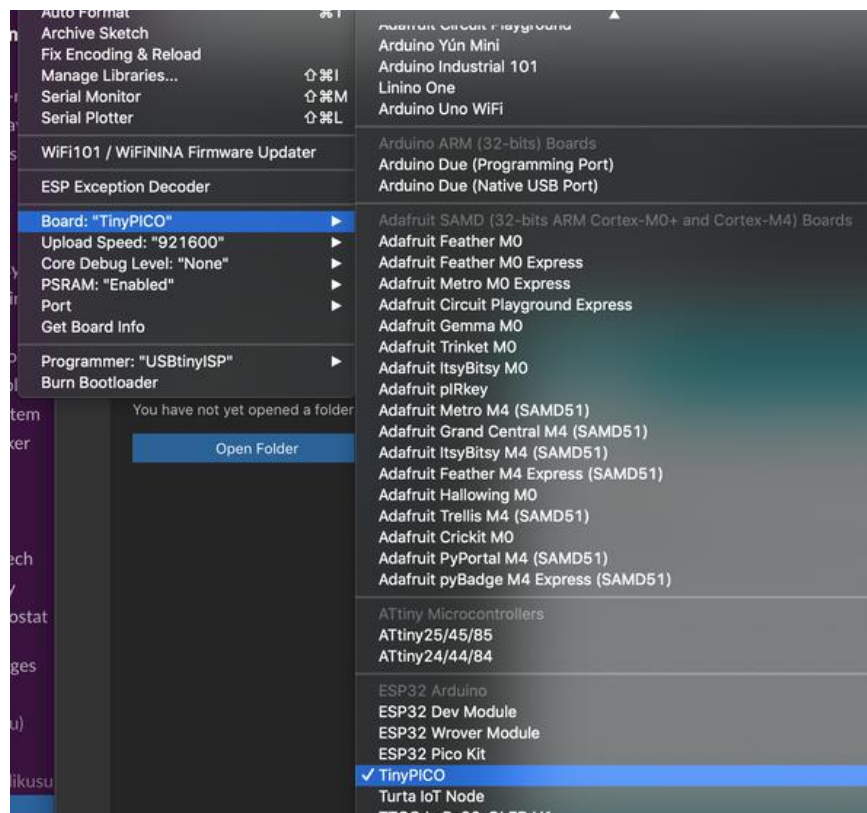
If you plan on using the Arduino IDE to program your TinyPICO, all you need to do is plug it in and flash it. MicroPython will be removed in the process and your TinyPICO will act just like any other development board you have used before.

Version 1.0.6 of the ESP32 Arduino Core is now available in the Arduino IDE boards manager - please make sure you are on the latest version.



NOTE! If your **TinyPICO** is the first ESP32 development board you've used, remember to follow [these steps](#) to make sure you have installed everything you need before hand.

If everything has been installed correctly, you'll need to select TinyPICO from the ESP32 boards list, and then you are set to go!



WHAT DO I DO NOW?

Now all you need need to do is check out our [coding examples](#) to find out where to get your TinyPICO helper library and get started coding!!!!

POWER FROM BATTERY

Your TinyPICO comes with 2 types of JST connectors and the board has pads on the back where you can solder either of the connectors.

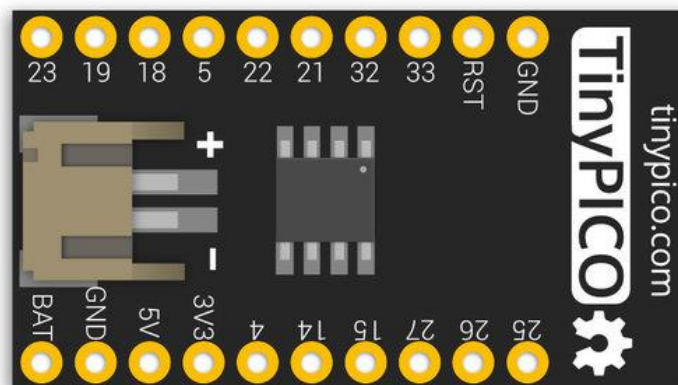
You can also connect a battery to the BAT & GND pins on the headers, if you don't want to solder a JST connector on the board.

You can use any size, single cell Li-Po battery to power your TinyPICO. Obviously, the larger the *mA*H rating of the battery, the longer it will last.

To get optimal battery life, remember to shut down the RGB LED if you are not using it, or whenever it's not lit up.

Please use LiPo-Batteries with charge protection included on them. Most hobby RC batteries do not have over discharge protection, and if a battery discharges too much, it can damage it and it might never be able to hold a proper charge again.

NOTE: When connecting a battery to your TinyPICO, please check the polarity of the battery, as not every battery is wired the same way. The IC we use for charging does have in-built reverse voltage protection, but it's better not to tempt fate!



FAQ

- **MY RED POWER LED DOESN'T LIGHT UP WHEN RUNNING OFF BATTERY**
- **MY ORANGE CHARGE LED BLINKS WHEN NO BATTERY IS CONNECTED**
- **HOW FAST WILL MY BATTERY CHARGE?**
- **SOMETIMES MY BATTERY DOESN'T START CHARGING WHEN I PLUG IT IN**
- **I'VE SET MY RGB LED TO BLACK/CLEAR, BUT IT'S STILL DRAWING A LOT OF CURRENT**
- **WHAT WIFI RANGE CAN I EXPECT?**
- **I'VE PLUGGED MY TINYPICO IN BUT I CAN'T SEE IT IN THE ARDUINO IDE OR IN /DEV/**
- **I GET "FAILED TO CONNECT TO ESP32: TIMED OUT WAITING FOR PACKET HEADER" WHEN I TRY TO FLASH MY TINYPICO**
- **MY TINYPICO GETS HOT, ESPECIALLY WHEN RUNNING ON WIFI, IS THAT OK?**
- **WHAT VOLTAGE RANGE CAN THE 5V PIN ACCEPT?**

MY RED POWER LED DOESN'T LIGHT UP WHEN RUNNING OFF BATTERY

That is by design! The power and charge LEDs and corresponding 2K resistors drain the battery, so they have been isolated to the 5V power domain, so they will only light up and draw power when the board is either being powered by the USB plug, or by a 5V power source connected to the 5V header pin.

[top](#)

MY ORANGE CHARGE LED BLINKS WHEN NO BATTERY IS CONNECTED

That is the correct behaviour! The charge IC can't tell the difference between a full battery and no battery, so it cycles between trying to charge and not charge.

[top](#)

HOW FAST WILL MY BATTERY CHARGE?

The charge rate is fixed and set to 330mA based on the 3K3 charge resistor we used. This is far from the fastest you can charge a battery, but we'd rather err on the side of safety than be too aggressive.

[top](#)

SOMETIMES MY BATTERY DOESN'T START CHARGING WHEN I PLUG IT IN

If you have 5V power connected to your TinyPICO before you plug your battery in, sometimes it might not start charging. Just unplug and re-plug in the 5V source and it should start charging correctly. This is a common issue with PMICs (Power Management ICs).

I'VE SET MY RGB LED TO BLACK/CLEAR, BUT IT'S STILL DRAWING A LOT OF CURRENT

The APA102 has a quiescent current of between 500uA and 1mA - that's a lot! To counter this, we added a high side switch to the power of the APA, connected to GPIO13. So you not only need to set the APA to be black/clear, but you also need to pull GPIO13 low to cut power to the APA.

To make this easy for you, we added a simple helper function to both our MicroPython and Arduino TinyPICO libraries :-)

[top](#)

WHAT WIFI RANGE CAN I EXPECT?

This is really a tough question to answer as there are so many external factors that can affect WiFi performance, including construction of the building you are in, your network layout, power or RF noise in your area and where you are placing your TinyPICO.

All we can say is we spent a great deal of time tuning the matching network and the antenna implementation, to ensure it's the best it can be. The rest is up to you and your environment.

[top](#)

I'VE PLUGGED MY TINYPICO IN BUT I CAN'T SEE IT IN THE ARDUINO IDE OR IN /DEV/

The TinyPICO uses a SiLabs CP2104 to connect the USB to the PICO-D4, so you need to make sure you have the driver installed. You can grab the [latest drivers from from the SiLabs website](#).

Once the driver is installed (you may need a reboot your computer) the TinyPICO should appear in the Arduino IDE and as a device on your computer in the following formats:

macOS: **/dev/tty.SLAB_USBtoUART**

Linux: **/dev/ttyUSBx** where **x** is the index of the USB device

Windows: **COMn** where **n** is the port number assigned by Windows

[top](#)

I GET "FAILED TO CONNECT TO ESP32: TIMED OUT WAITING FOR PACKET HEADER" WHEN I TRY TO FLASH MY TINY PICO

A few folks have seen this from time to time. 95% of the time it's fixed by swapping the USB cable and or trying a different USB port on your computer. If that doesn't fix the problem for you, please contact us and we'll help you out.

[top](#)

MY TINY PICO GETS HOT, ESPECIALLY WHEN RUNNING ON WIFI, IS THAT OK?

The Espressif ESP32-PICO-D4 is a very powerful SIP (System In a Package) for its size, and when running Wi-Fi continuously it gets hot, and because the TinyPICO is small, there is not a lot of place for heat to transfer to.

This is **nothing to be alarmed about** though. Even when running hot, everything is well within spec, so long as you follow some simple rules:

- If you need to keep Wi-Fi active for long periods of time, like for a soft-ap setup, make sure you don't enclose your TinyPICO in a small case or a case that has no air flow, and maybe even consider adding a small heatsink to the board to help with heat transfer.
- Try to keep WiFi use short and sparse, if your project allows. Deep sleep your TinyPICO whenever you can, even when running off 5V as that will allow the ESP32 to cool down more than if it was fully awake.
- If you need to put your TinyPICO in a case, add a small fan to pull the hot air out to keep the ambient temperature around the TinyPICO as low as possible. If a fan is not possible, try to at least have air vents to let heat out.

[top](#)

WHAT VOLTAGE RANGE CAN THE 5V PIN ACCEPT?

The 5V pin on the TinyPICO is an in and out pin. As an input, it will accept a voltage range between 4.8V and 5.2V. Please do not connect any power source over 5.2V to the 5V pin and please do not connect your battery to the 5V pin. It is for a 5V power source only.

When USB power is connected, you will get approximately 4.9V out of the 5V pin. The voltage drop is due to the reverse USB power diode that allows the USB and 5V pin to be power inputs at the same time, and prevents back feeding power into the USB.

The use case for having both connected is when you want to power your TinyPICO from the 5V pin from an external 5V power source, but you also want to connect the USB to flash the board, or to get serial output.

[top](#)



TinyPICO & TinyPICO Nano are designed and manufactured by Seon Rozenblum
aka, Unexpected Maker
Follow more of Seon's maker adventures at <http://unexpectedmaker.com>