



**University of  
Sunderland**

RASPBERRY PI PICO

# SYSTEM GUIDE

LET'S WORK ON BUILDING A  
GAMES CONSOLE OF OUR VERY  
OWN; FUN STARTS HERE!

For this project we are using the low cost, yet powerful, Raspberry Pi Pico microcontroller. This can be used to create a range of exciting applications and projects. Today, it's a games console. Let's show you how!

## PI PICO GAMES CONSOLE

**W**elcome to this Pre-Applicant Day Magazine, which will take you through the process of making your own Handheld Game Console.

Thank you for applying to the University of Sunderland and the School of Computer Science to study one of our dynamic programmes. This magazine and the components you have received in the post will guide you in the building and programming of your very own console. We will show you what software you will need, how to install this and how to put together the hardware.

There will be tips on programming, microcontrollers, such as the raspberry pi pico, and some fantastic follow-on projects you can also try.

During your visit to the School of Computer Science on the 11<sup>th</sup> March we will add some extra features to our handheld games console such as extra buttons and sound!

During the day, you will also design and 3D print your very own custom and bespoke case to package your device.

Exciting times indeed! This is sure to be fun!

**Let's dive in!**



# COMPONENTS

In the pack you received from the University of Sunderland, you will find the components shown here. These components will be used to build your games console.

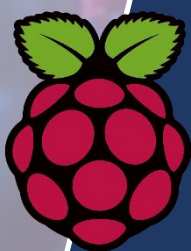
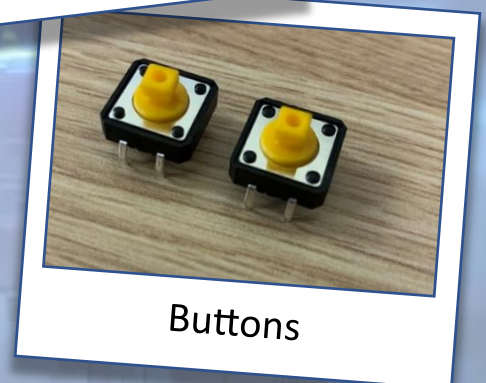
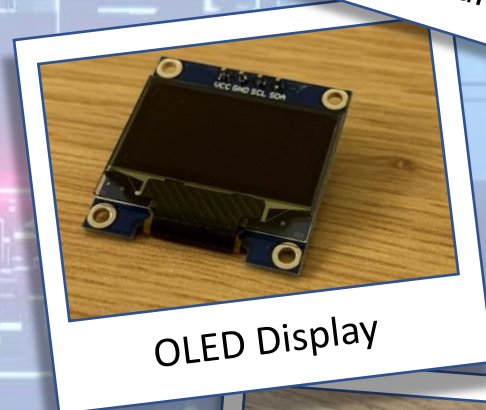
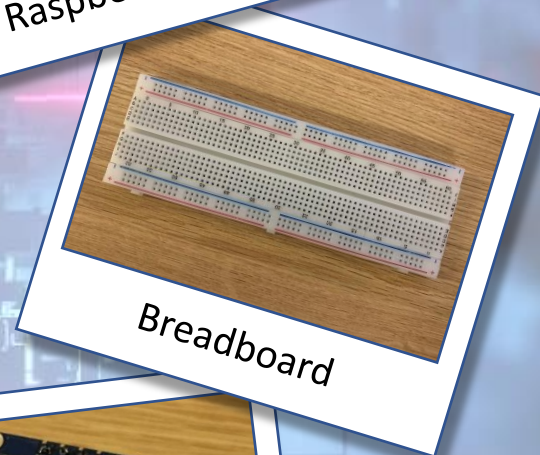
A Description of the components is below:

**Raspberry Pi Pico** – This is a small microcomputer that will act as the brains of your device, this has memory, RAM and inputs and outputs.

**OLED 0.98" Screen** – This is your display and will be used to show your games output so you can see and play your games.

**Breadboard** – This is your building block; this is what all the components will be connected to. This is a plug and play board and does not need soldering.

**Buttons (x2)** – These buttons are used to interface to your device and allow you to control your games that you make!



## MORE DETAIL

If you wish to have a look at more sensors and items that can be used with the Pi PICO or get more information on the components you have, check out the following link:

<https://tinyurl.com/36juawy2>

# SOFTWARE ELEMENTS

Your Raspberry Pi PICO you have received in your packs is ideal for learning to code. In this magazine we'll take you through the basics of programming, to get you ready for your visit to the university's applicant day.

The programming software we use is called Thonny and is free to download from the following link, once downloaded and installed we'll write a simple Python program to test everything.

<https://thonny.org/>

## Help Videos

At any point, if you need help along the way, take a look at our support videos by clicking on the YouTube Icon.

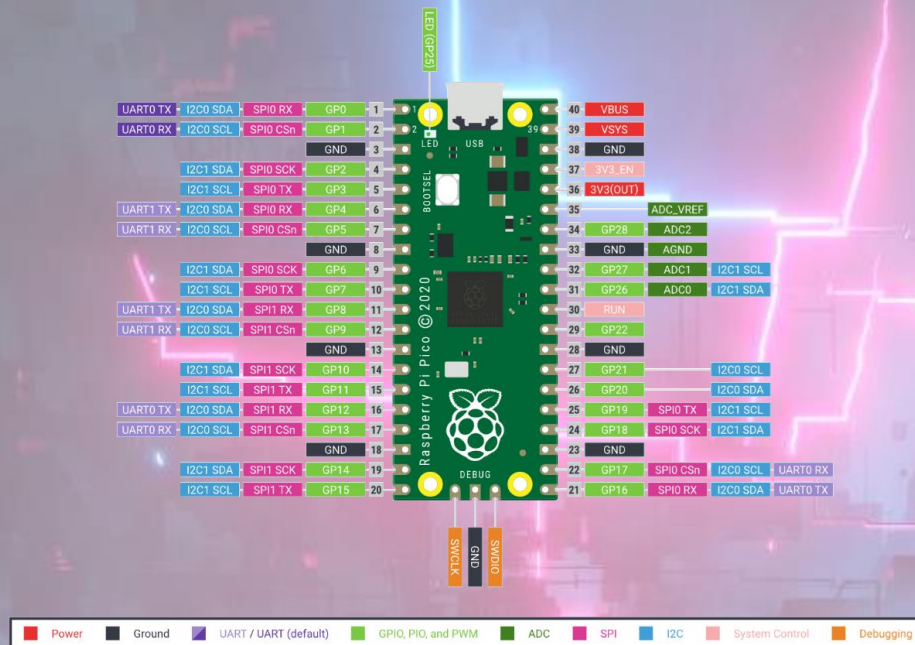


The Raspberry Pi Pico uses a version of the Python Programming Language called micro-Python.

We are going to test our software is installed and everything is working by writing a very simple Python Program to make the LED on the Pi Pico flash.







# PYTHON PROGRAMMING

## 01 Open Thonny & Python Version

Thonny is used to program your PICO and so we need to make sure it is using the correct version of Micro Python for the PICO.

Look at the bottom right of Thonny, click on the version of Python listed and select 'Configure Interpreter'.

## 02 Using the Correct Interpreter

Before using your PICO Pi you need to select the correct interpreter. Therefore, connect your PICO Pi with a USB cable, then in Thonny, on the menu bar select:

Tools > Options... > Interpreter

From the first dropdown box select:

MicroPython (Raspberry Pi Pico)

From the 'Port' dropdown box select:

USB Serial Device (COMX)

Click OK, you're now using the Pi PICO directly to code.

## 03 Entering the Code in Thonny

Writing a program for your Pi PICO is like writing a program for any other device, we can use Thonny to write the code and send it to the Pi PICO.

You can write programs in the Shell area at the bottom of the screen to run immediately, or we can write them at the top and save them as files.

## 04 Hello, World!

Click on the Shell area, next to the >>> symbols, and type:

```
print("Hello, World!")
```

When you press ENTER at the end of the line, you'll see your Pi PICO respond.

## 05 Flashing LED

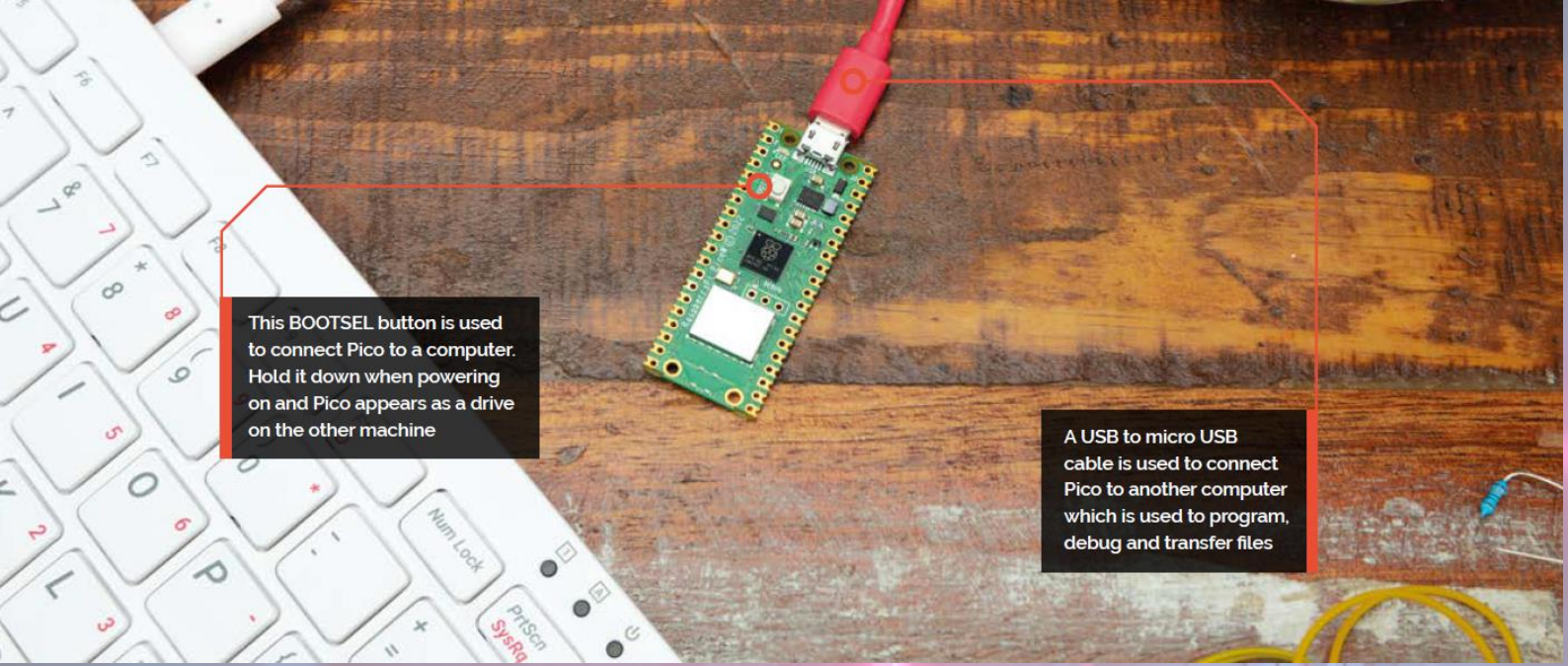
In the Shell area type the code:

```
from machine import Pin
led = Pin(25, Pin.OUT)
led.toggle()
```

This will make the LED on the Pi PICO turn on, Typing led.toggle( ) again, will make it turn off, and again, back on.







This BOOTSEL button is used to connect Pico to a computer. Hold it down when powering on and Pico appears as a drive on the other machine

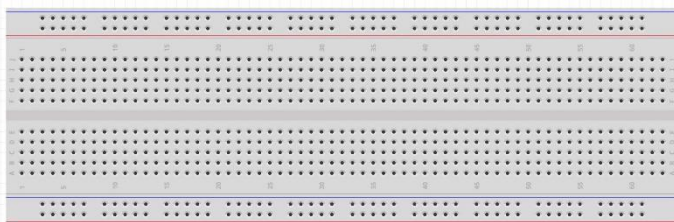
A USB to micro USB cable is used to connect Pico to another computer which is used to program, debug and transfer files



## BUILDING A GAME CONSOLE

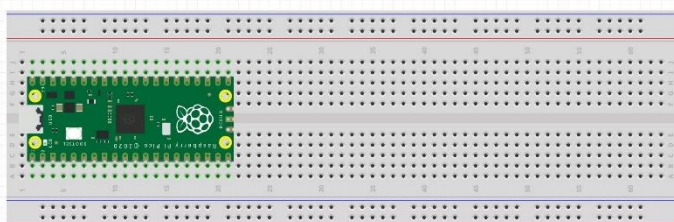
Follow the diagrams below to build your Pi PICO Game Console. Make sure you wire the connections correctly!

### STEP 01



Breadboard – This is used to secure all the components and will act as the frame of the console.

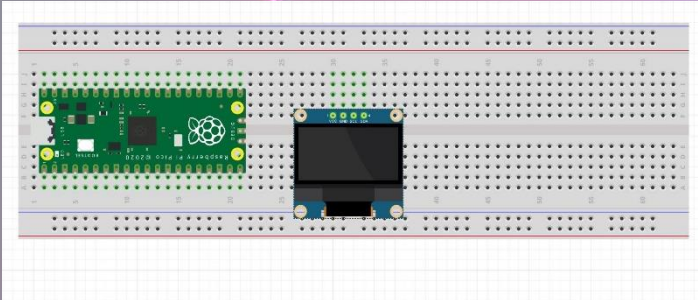
### STEP 02



Pi PICO – Push the PICO into the breadboard in the location shown. Making sure that the USB port is facing the edge.

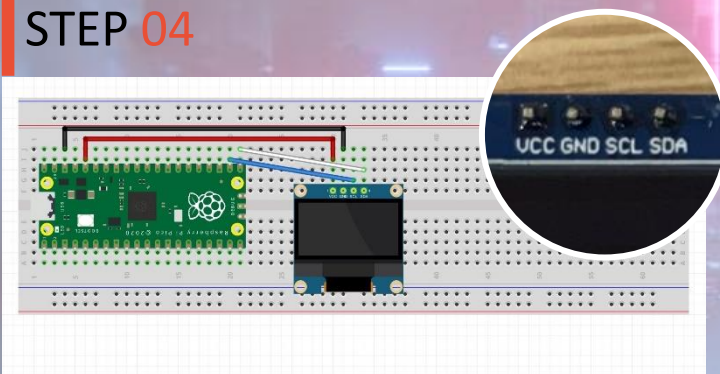


### STEP 03



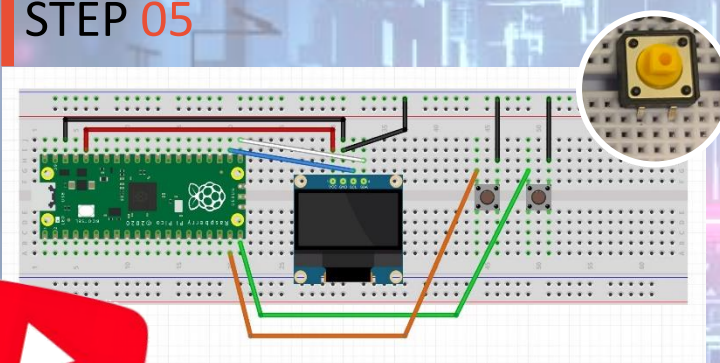
OLED Screen – Push the headers of the Screen into the breadboard. Make sure the Screen sits fully over the breadboard.

### STEP 04



Wire the OLED Screen – Connect the wires to the pins shown. Pay close attention to the labels on the OLED Screen, VCC goes to Pin 5.

### STEP 05



Buttons - Connect the buttons to breadboard, they need to be correctly oriented, see the insert image for a closer photo.



DOWNLOAD  
THE FULL CODE:



[tinyurl.com/UoS-FOT](https://tinyurl.com/UoS-FOT)

