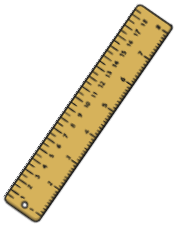
Let's think about building a MARS Rover!

What is a MARS Rover and why do we need to be able to implement obstacle avoidance?



DISTANCE:
Distance is the measure of the space between two points. This can be standardised into a unit of measurement such as centimeters.

What is a MARS Rover?

In robotics MARS is an acronym:

Mobile Autonomous Robotic System

Mobile: The device can move. This usually involves the use of either wheels, tracks or legs.

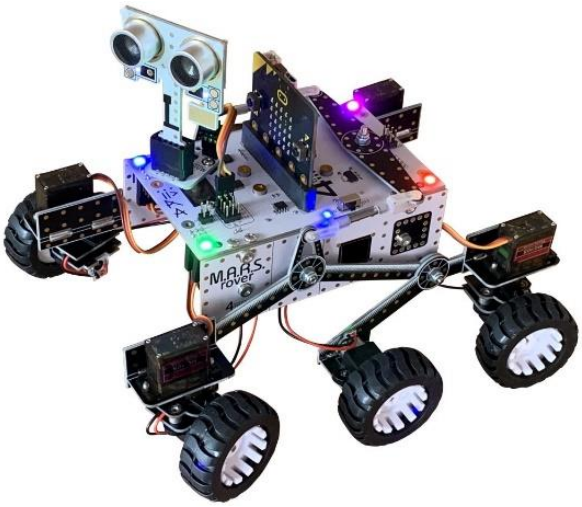
Autonomous: The device/robotic system is capable of some degree of self-sufficiency. No remote control!

Robotic: A programmable machine or device that generally includes some form of actuators and controller.

System: This generally means that the components of the device are integrated to provide a solution to a problem.

What is a Obstacle Avoidance?

Obstacle avoidance is usually about self preservation. It prevents a MARS rover from damaging itself by colliding with obstacles.



University of Sunderland

Let's think about building a MARS Rover!

What are we going to need to investigate distance sensing?

KIT LIST

1 * HC-SR04

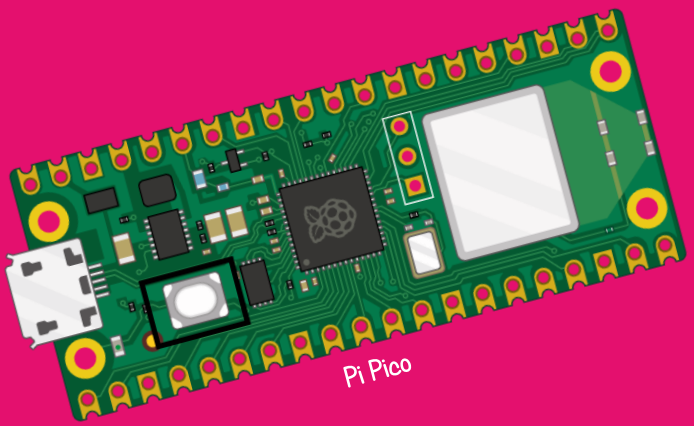
1 * Pi Pico

1 * oled 1306 Display

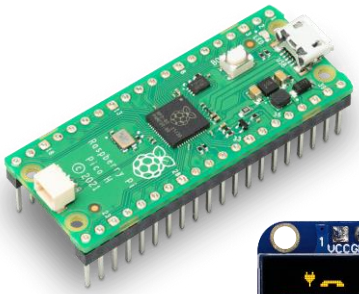
1 * Breadboard

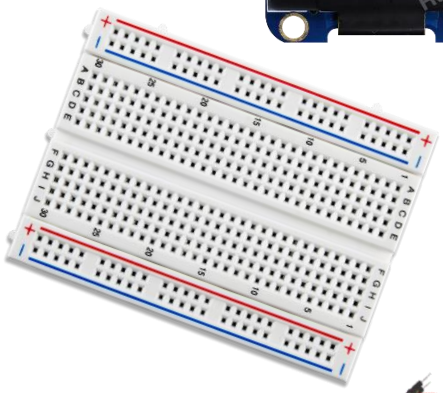
12 * Jumper wires

1 * USB Cable



Kit





University of Sunderland

Let's think about building a MARS Rover!

From design to build what will our distance sensing device end up looking like?

KIT LIST

1 * HC-SR04

1 * Pi Pico

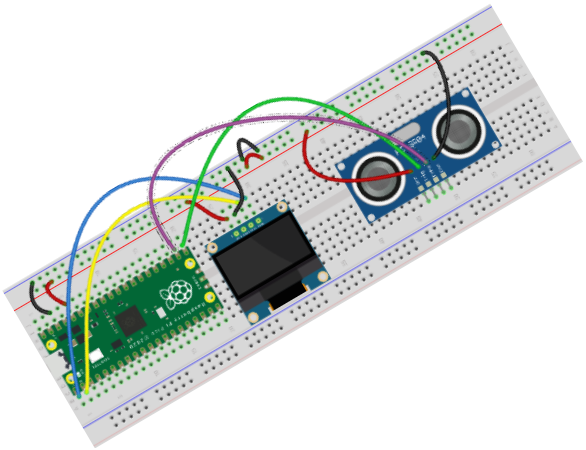
1 * oled 1306 Display

1 * Breadboard

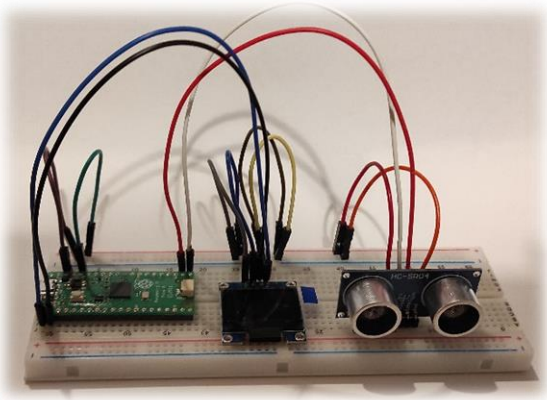
12 * Jumper wires

1 * USB Cable

Design



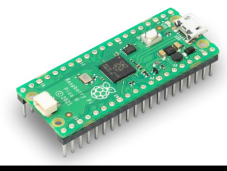
Build



Faculty of Technology

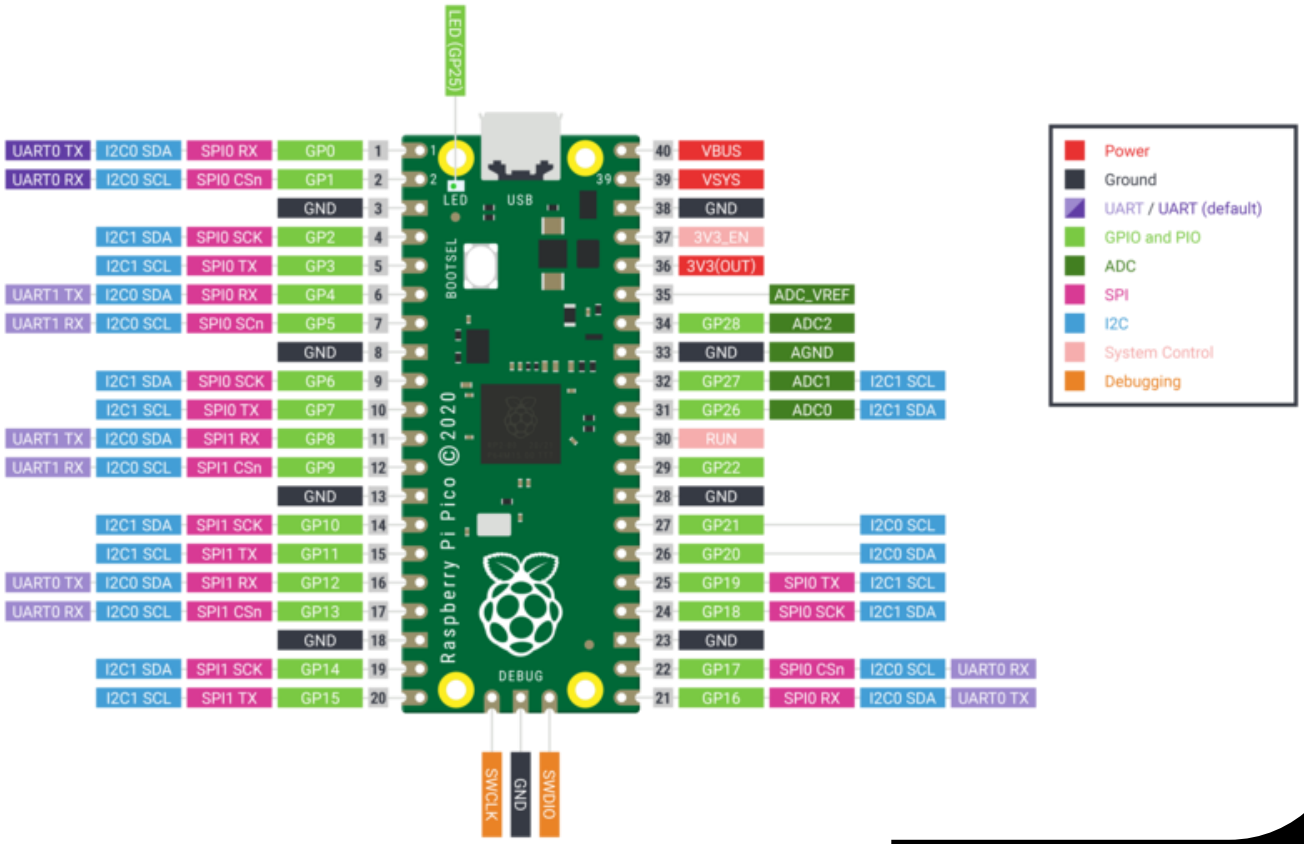
University of Sunderland

Let's think about building a MARS Rover!



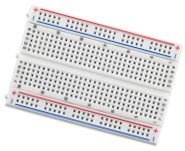
We're using a Raspberry Pi Pico as our microprocessor. So, what exactly is it?

Raspberry Pi Pico is Raspberry Pi's first microcontroller board, designed especially for physical computing. Microcontrollers are a different type of device than Single Board Computers (like the Raspberry Pi 4 and previous generations of Pi). They don't run an operating system and they are typically programmed to do just one task – though that task can be pretty intricate and exciting! They're perfect for experimenting with hardware and using as the brains of custom devices, machines, and inventions.



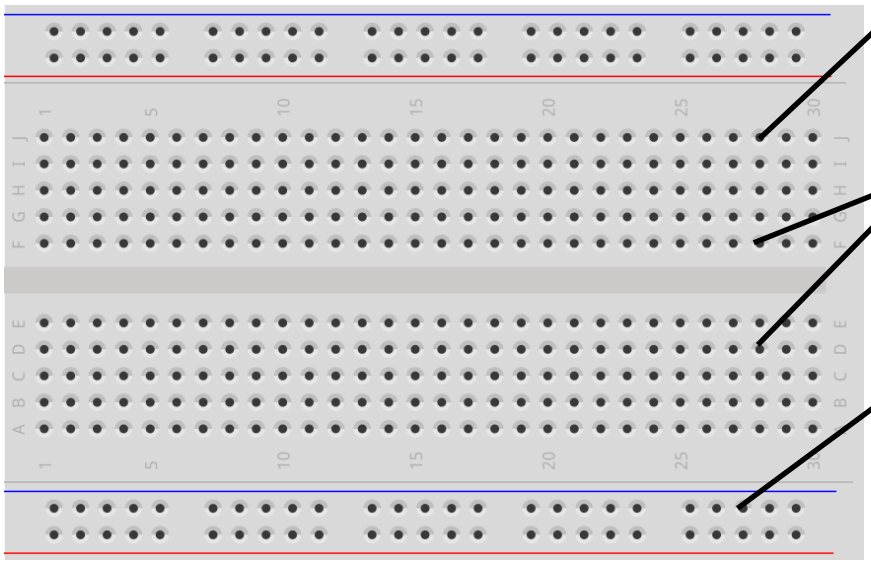
University of Sunderland

Let's think about building a MARS Rover!



We're building our distance sensing device (Obstacle Avoidance?) component as a prototype using a breadboard.
So what exactly is a breadboard?

A breadboard is used as a platform to prototype your circuits on. They provide a reusable environment that allows different ideas to be tested.
Breadboards come in different sizes. but they all have a pattern of holes for jumper wires. The main area has holes connected vertically and the edges (power rails) have the holes connected horizontally.



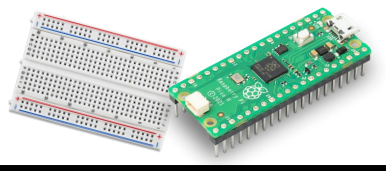
Vertically connected holes to allow patching to other vertical areas..

The two areas are separate to allow more devices to be included.

Horizontally connected rows to distribute power to devices. RED for positive (+). BLUE for negative (-).

University of Sunderland

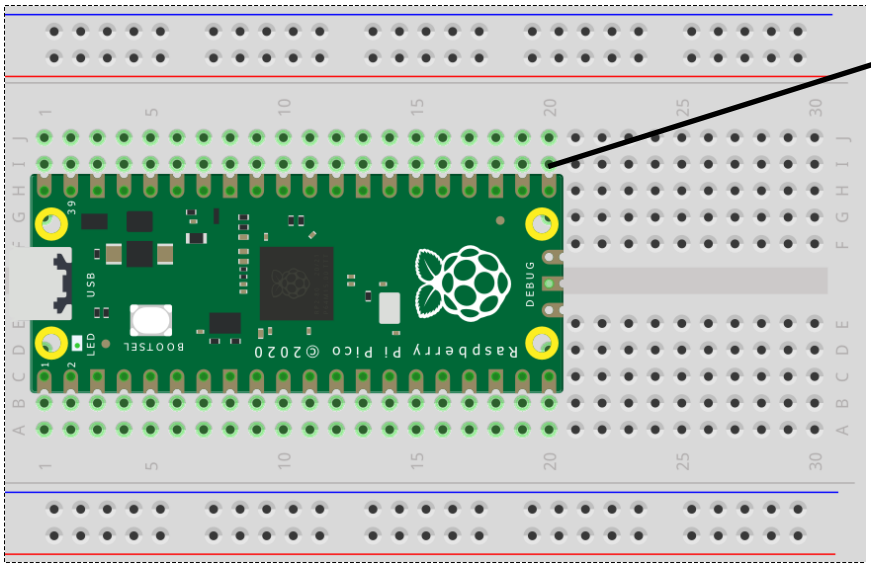
Let's think about building a MARS Rover!



It's time to put the microprocessor onto the breadboard which is where we are going to build our distance sensing prototype. So how do we line them up and fix them together?

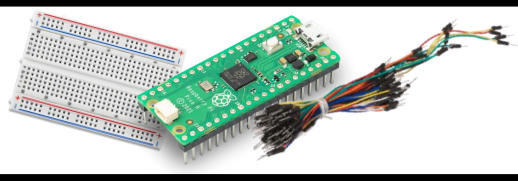
The breadboard has 30/60 columns in two blocks in the centre. The Pico must be located at the very left-hand side with the microprocessor pins in the middle of the two blocks and let left most pins lined up to the very left-hand edge. as shown in the diagram below.

Be very careful pressing the Pico down. don't force it or it may break. Press gently at the edges so it sits flush to the breadboard with the Pico pins all the way in so you cannot see them.



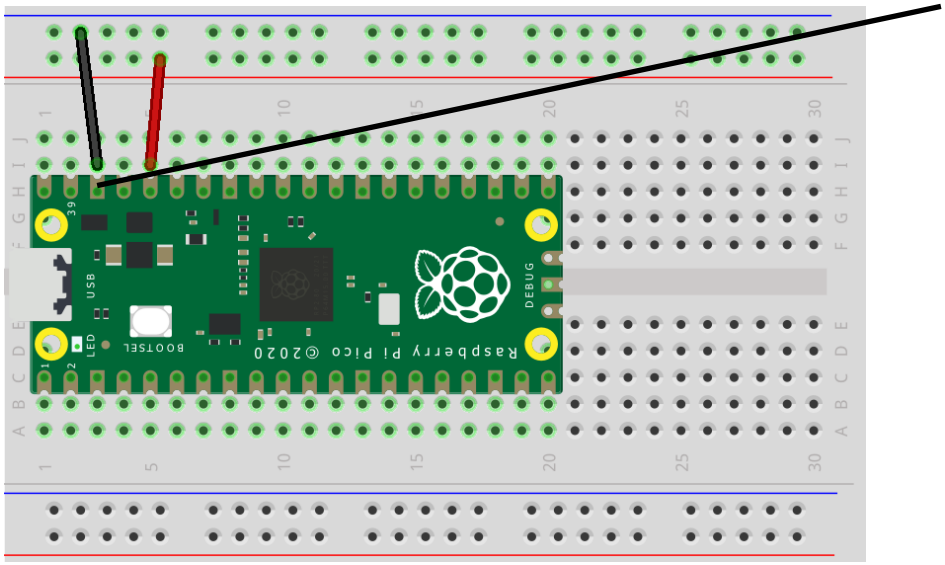
University of Sunderland

Let's think about building a MARS Rover!

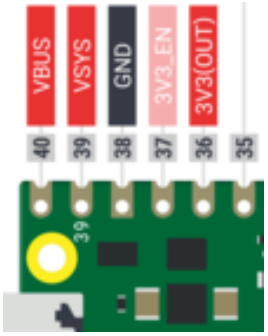


That's the Pico on the breadboard and we can start adding all the components and wires.
Let's start by adding power to the power rails (rows) at the top of the board.

The top row of the breadboard needs to have a 3.3v (+) and ground (GND) added to them.
We can do this by adding a wire from the 3.3v power out on Pico to the RED rail with a wire and a GND to the BLUE rail using a wire. See the diagram on page 4 for the Pins.
Pin 38 - BLUE rail. Pin 36 - RED Rail

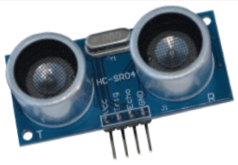


You can tell the GND pin as it has square edges. (There's a few!)



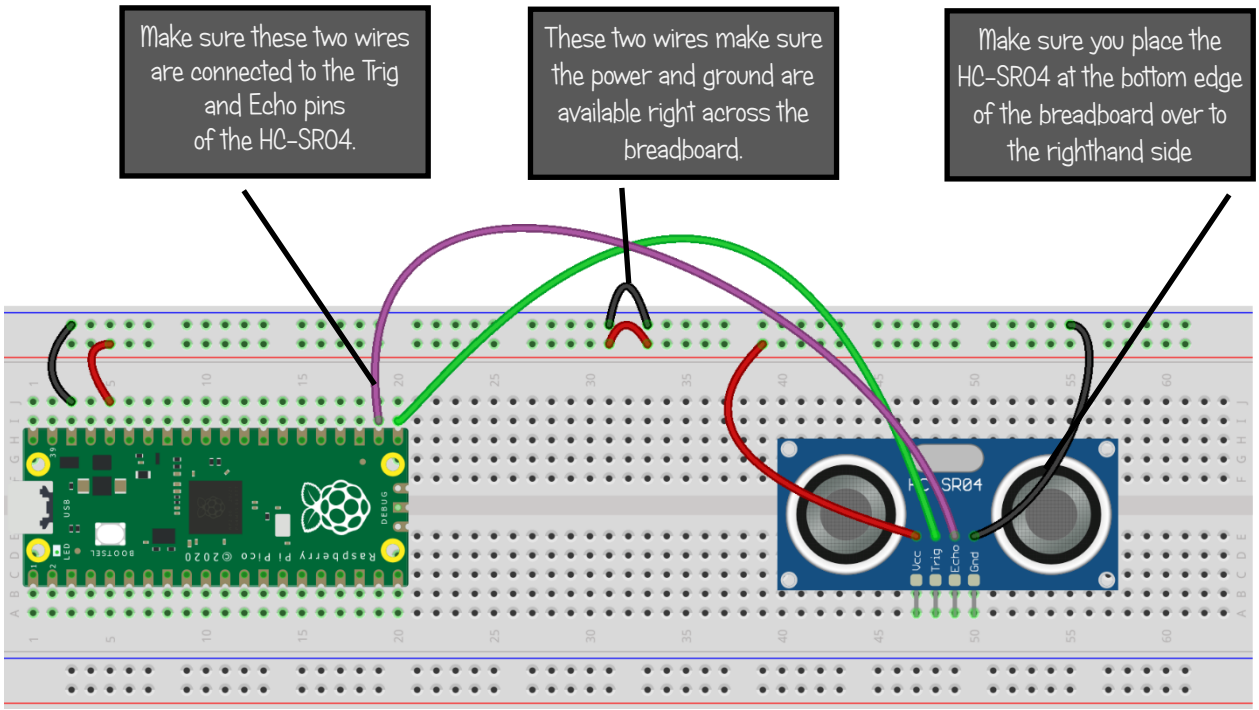
University of Sunderland

Let's think about building a MARS Rover!



Now we have the Pi Pico on the breadboard
and the power distribution is ready.
Now we can site the rest of the components.

The first component we can add is the HC-SR04 sensor.
It has 4 connectors that need to be lined up on the breadboard
The four connectors are 3.3v power.
GND. Trigger (TRIG). and echo (ECHO) which send the pulse (timing) reading to the Pico.



NOTE!
Connect the HC-SR04 wires behind the sensor so the wires aren't in fromnt of the transmitter and recievers

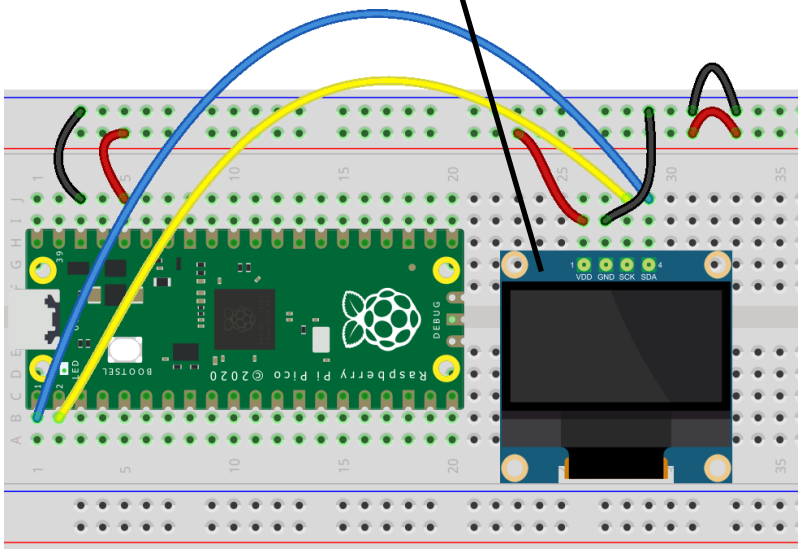
University of Sunderland

Let's think about building a MARS Rover!

Now we have the Pi Pico on the breadboard
and the power distribution is ready we can site the rest of the components.

The next component we can add is the OLED screen.
It has 4 connectors that need to be lined up on the breadboard
The four connectors are 3.3v power (VCC). GND.
and the clock (SCL/SCK) and data signal (SDA) pin which received data from the Pico.

The oled is what we call an i2c device and allows us to have many devices in a chain. In this project we are only using one.

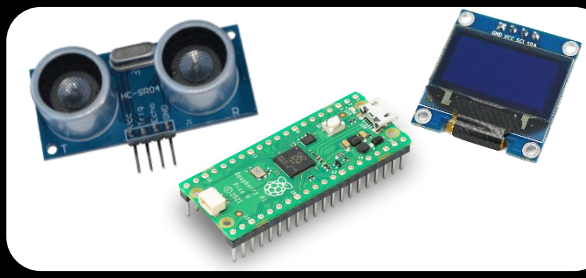


BE AWARE!

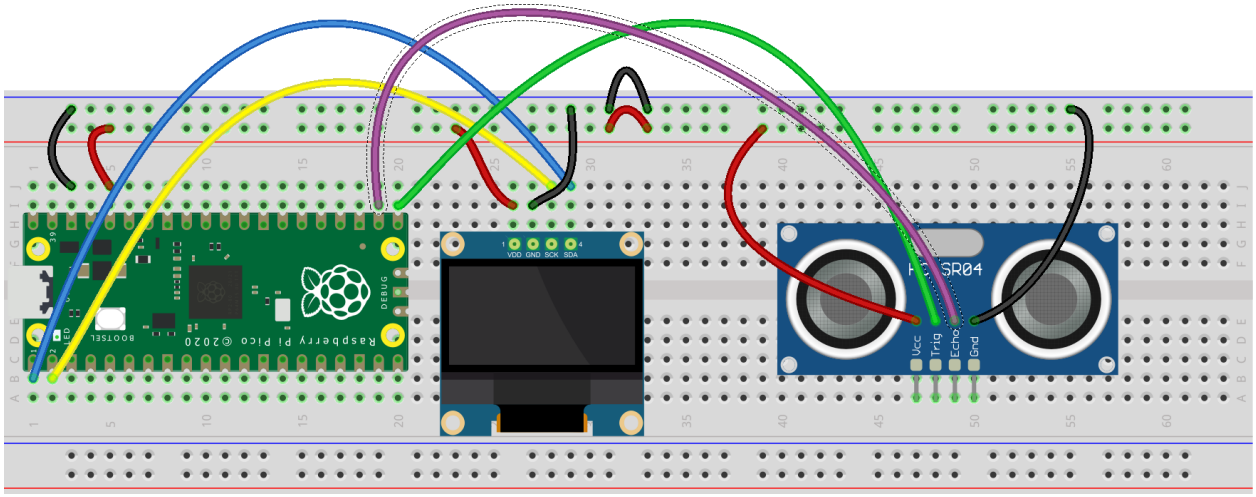
Check the connections carefully some OLED have a different layout!

University of Sunderland

Let's think about building a MARS Rover!



Now we have all the components on the board it should be something like this!

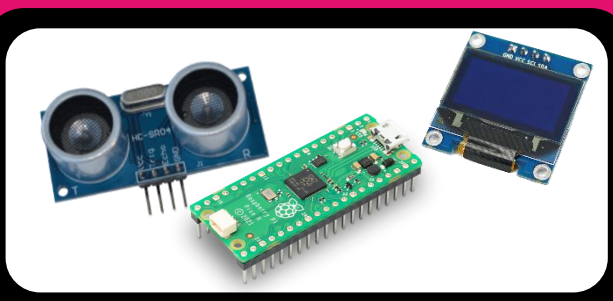


NOTE!

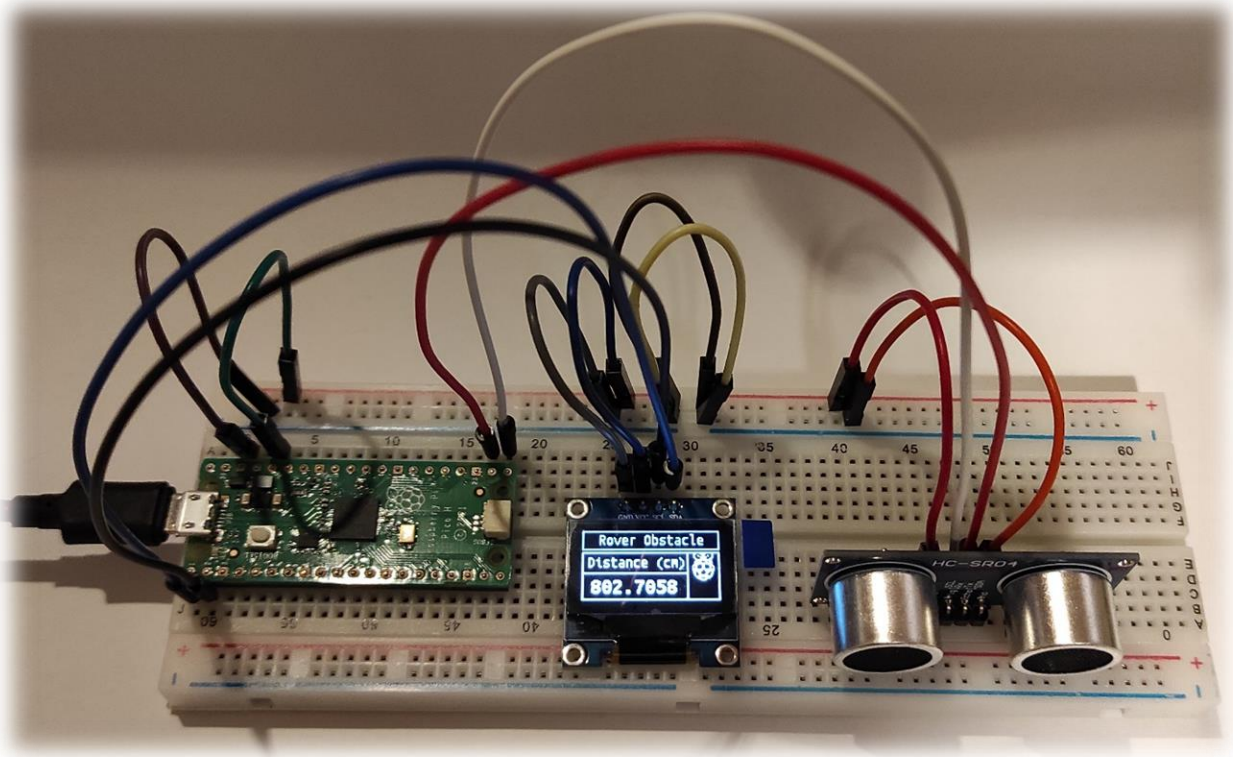
Wire colours are used so you can see the start and end points. However, you can use any colour wire you like!

University of Sunderland

Let's think about building a MARS Rover!



He is a picture of a completed build!
All you need to do now is load the code onto the Pico with Thonny.



Once the software is installed you can try placing your hand in front of it and see what happens!
Now think about a device like this at the front of a MARS rover.
Now you know how to stop it crashing into objects!

University of Sunderland

Let's think about building a MARS Rover!

Follow the links below for the weather station code, how to download, and to get help on how to use Thonny.

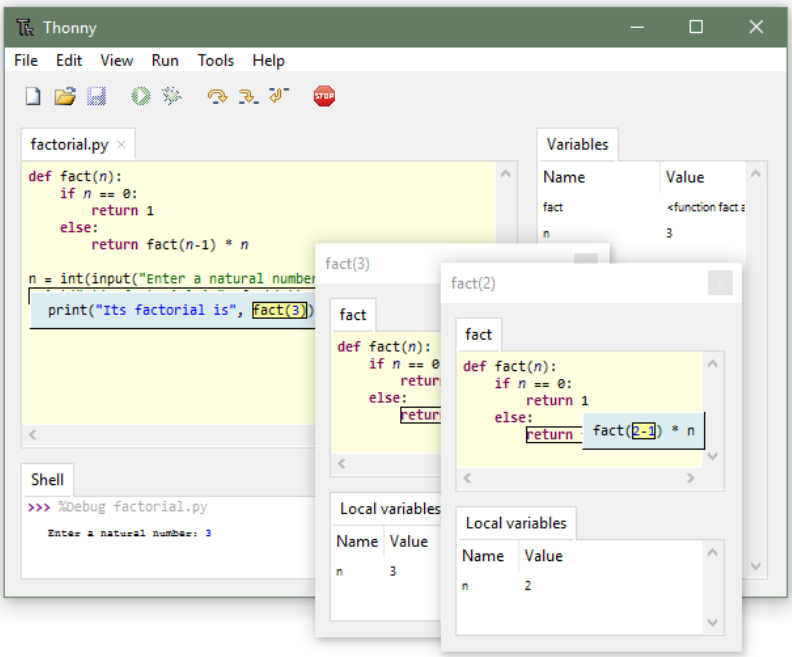
Click HERE to get the code from GitHub!
(https://tinyurl.com/ysx5fppd)

Click HERE for the Thonny download.
(https://tinyurl.com/42zyb2eb)

Click HERE for how to install Micropython as a file.
(https://tinyurl.com/y2sv7ytu)

Click HERE for how to install Micropython from Thonny.
(https://tinyurl.com/3dyypyfr)

Click HERE for help with using Thonny.
(https://tinyurl.com/3uwu9amw)



This page is intentionally left blank.

University of Sunderland