ZUMO WARS!

Zumo Wars Competition Rules

The battle takes place inside a competition standard mini robot sumo ring or "Dohyo". This Dohyo is circular and is painted black with a white border which the robots use to detect the edge of the ring. Two brown lines, known as "Shikiri" lines are located at equal distance from the centre of the Dohyo, these mark the starting positions for each competing robot.

The rule of the battle are as follows:

1. Both competitors place their Zumo robots facing each other on the Shikiri lines.

2. On a signal from a judge, each robots activation button is pressed, after 3 seconds, each robot should begin to move.

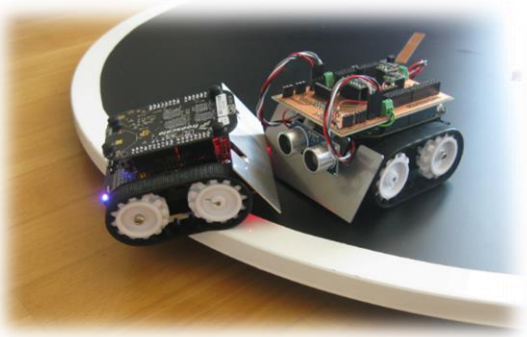
3. FIGHT!!

4. The remaining robot inside the ring is the winner.

What is a ZUMO?

The Zumo robots are small, tracked robots intended for mini sumo robot competitions. They contain two powerful motors and numerous infra-red sensors which can detect changes in colour from black to white, these are used to detect the white edge of the Zumo ring to ensure the robot doesn't fall over the edge.

The purpose of a Zumo robot is simple, to push other Zumo robots out of the Zumo ring. With an Arduino mounted on to the robot, we can control the movement, speed and response of the Zumo thereby determining what it does.



University of Sunderland

ZUMO WARS!

Kit

What are we going to need to build our battle bot?

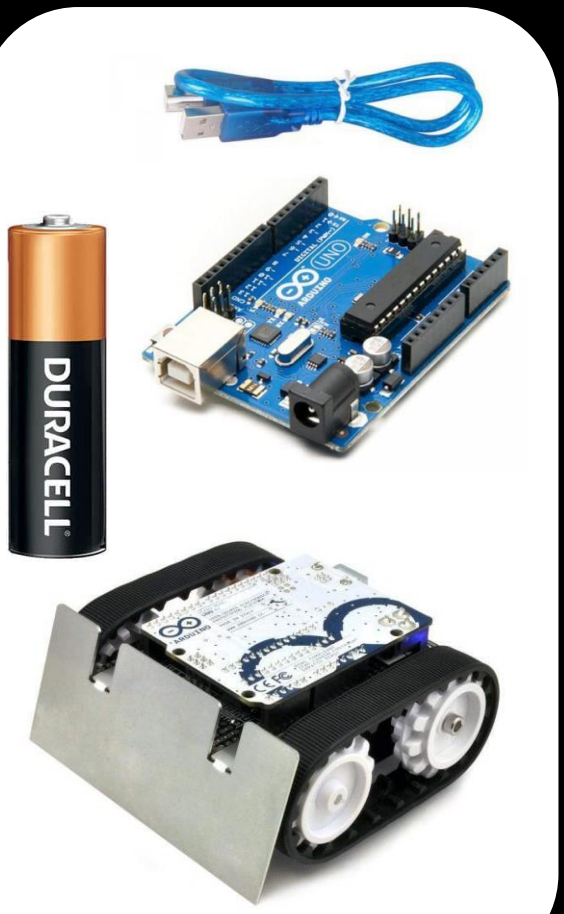
EXISTING KIT

1 * Arduino Uno

1 * Zumo Robot

1 * USB Type B cable

4 * AA Batteries

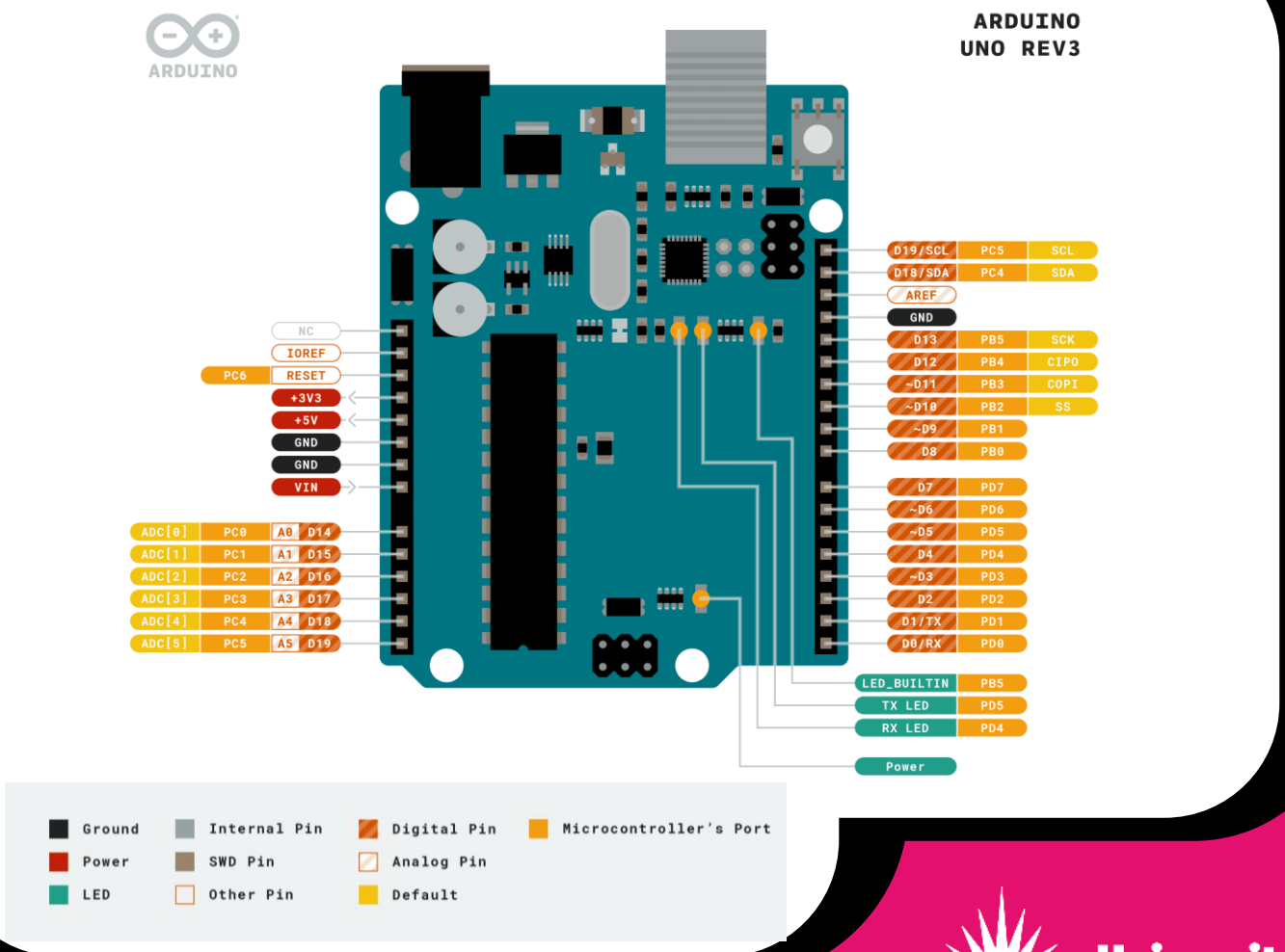


University of Sunderland

ZUMO WARS!

We are using an
Arduino Uno
as our microcontroller.
So what exactly is it?

The Arduino Uno is an open source microcontroller, designed especially for physical computing, also known as embedded systems. 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.

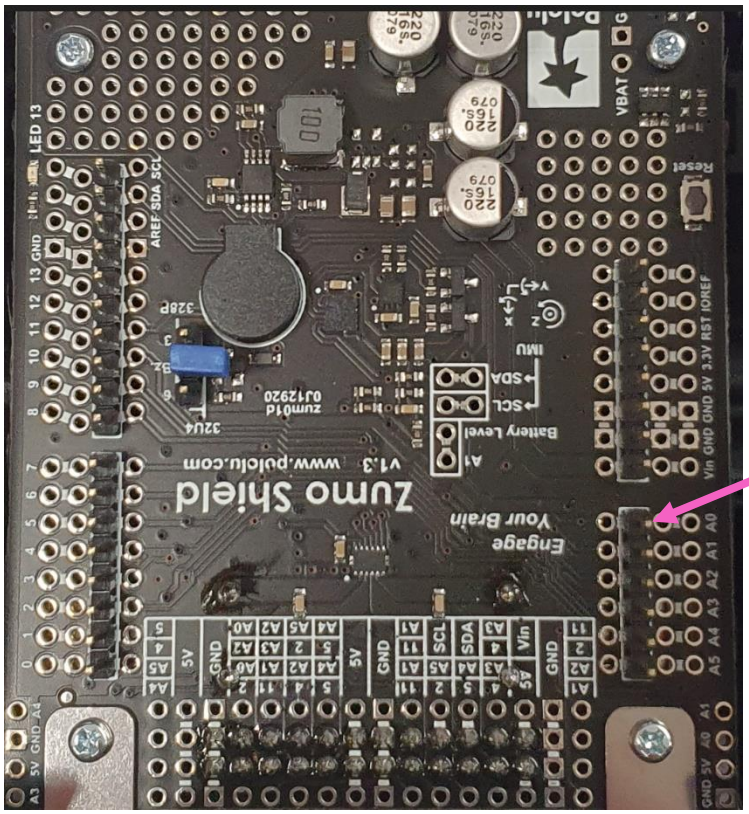


University of Sunderland

ZUMO WARS!

It's time to take a closer look at our Zumo
How is it similar to an Arduino Uno?

The Zumo contains the aptly named Zumo Shield which contains numerous components to make using the Zumo very easy and versatile. some of these components include LED's, buzzers, motor drivers and accelerometers, similar to the ones found in mobile phones. On the Shield you will notice and array of pins. If you look at your Arduino Uno, this contains the same pin layout as the Zumo shield, this is because the Zumo Shield has been designed to be controlled by an Arduino Uno by plugging the Shield and Uno together.

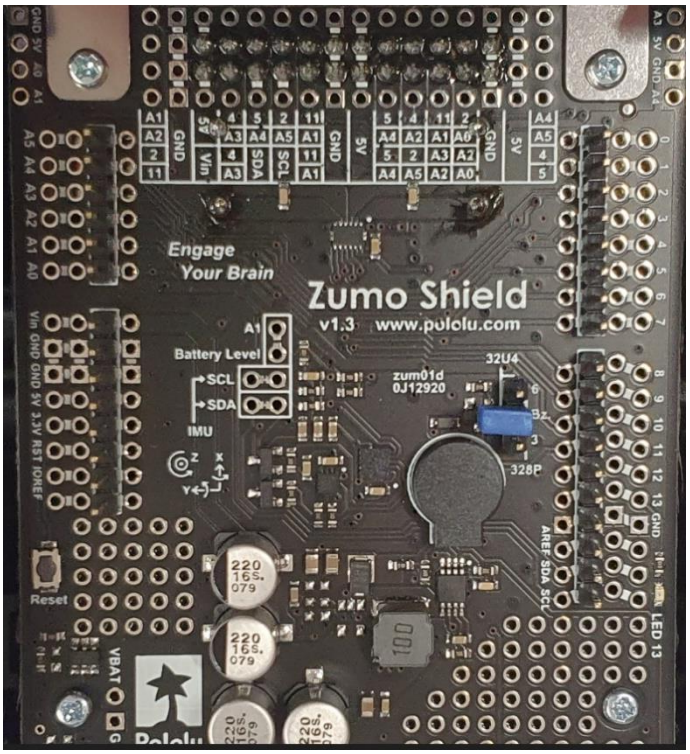


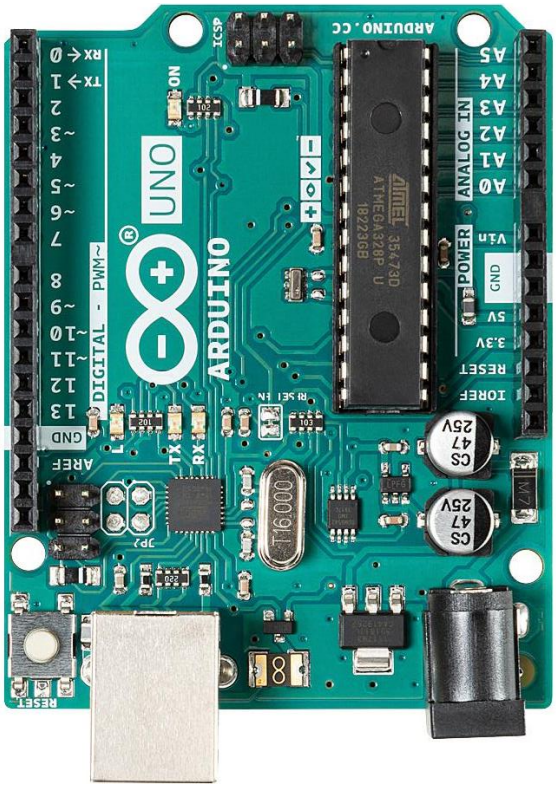
Note how the pins on the Shield match the pins on the Arduino

University of Sunderland

ZUMO WARS!

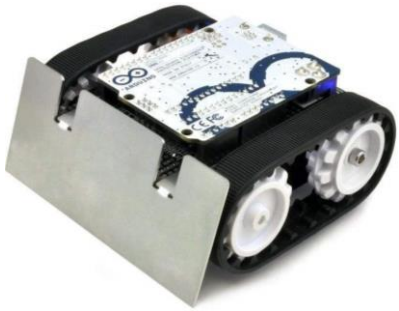
Let's build the Zumo Robot.





With the Zumo Shield and Arduino facing you as shown in the picture. turn the Arduino over and plug the Arduino onto the Shield.
Ensure all the pins are aligned and push in smoothly.

You may now connect the 4x AA batteries. Ensure the correct orientation of the batteries.



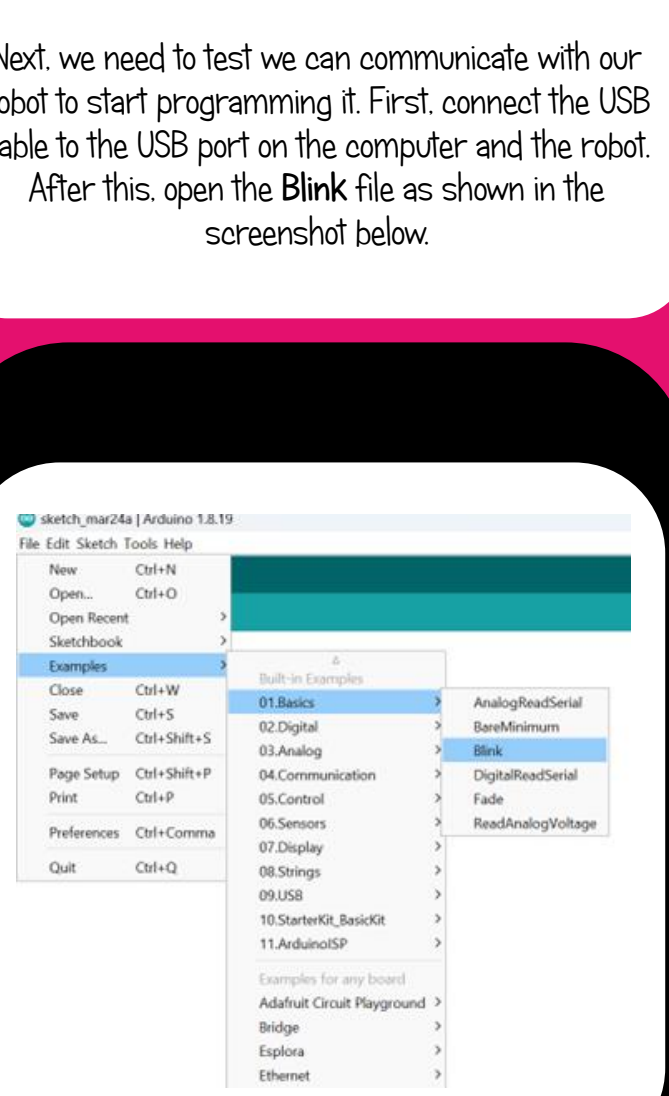
University of Sunderland

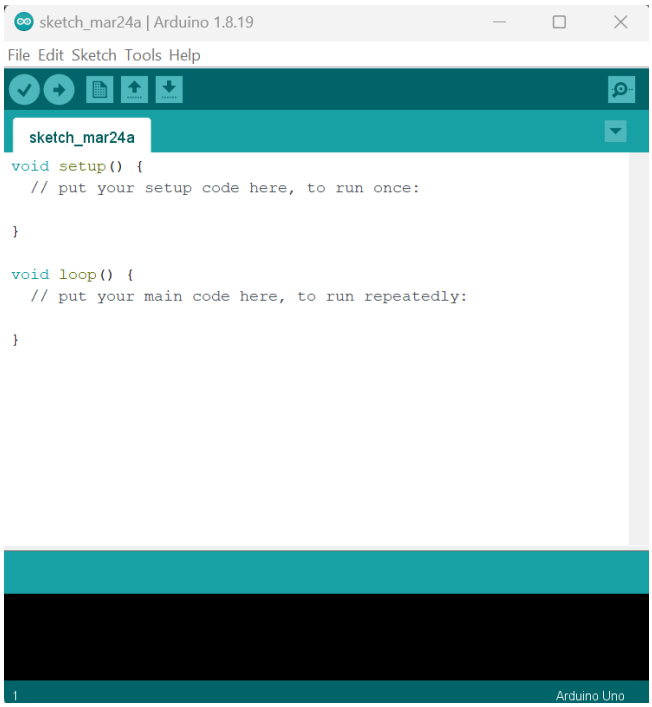
ZUMO WARS

It's now time to connect to the Zumo and start programming!!!

On your computer, search for a program called Arduino and open the program.

Next, we need to test we can communicate with our robot to start programming it. First, connect the USB cable to the USB port on the computer and the robot. After this, open the **Blink** file as shown in the screenshot below.





University of Sunderland

ZUMO WARS

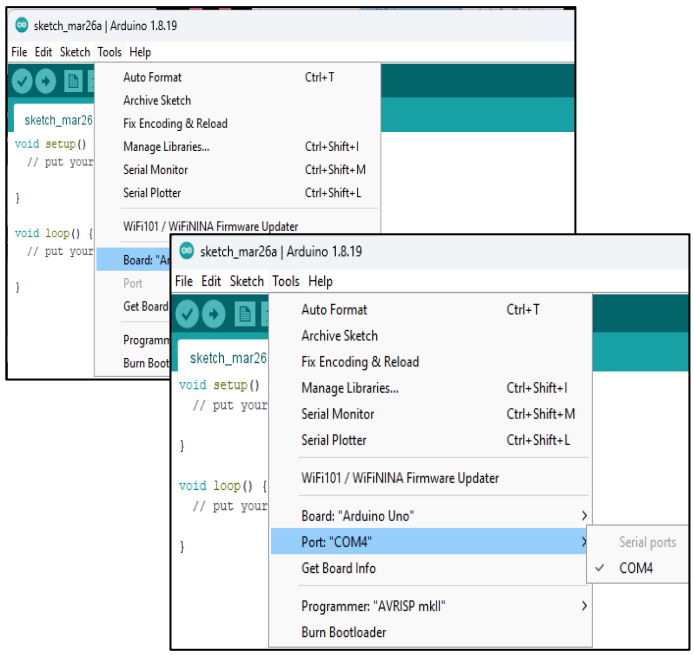
It's now time to connect to the Zumo and start programming!!!

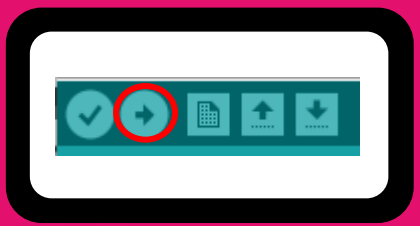
The code on the screen will, once uploaded, will cause the one of the LED's on the Arduino Uno to start blinking. We will use this code to ensure everything is connected correctly.

In order to upload any code to the Arduino, we need to ensure the correct board and port has been selected.

Board: Tools -> Board -> Arduino Uno
Port: Tools -> Port -> [Select the available COM port]

Note: There may be more Com ports showing on your display and so you may not know which one to choose. Remember which Com ports are present and then remove the USB from the Arduino. Reattach the Arduino and determine which is the new port. Select the correct port.





Once you have selected the correct board and port, it is now time to upload your code to the Arduino. The highlighted icon is the Upload button, this will first "Compile" your code and check if there are any errors, next, the code will be uploaded to the Arduino.

Check now to see if there is a blinking LED.

University of Sunderland

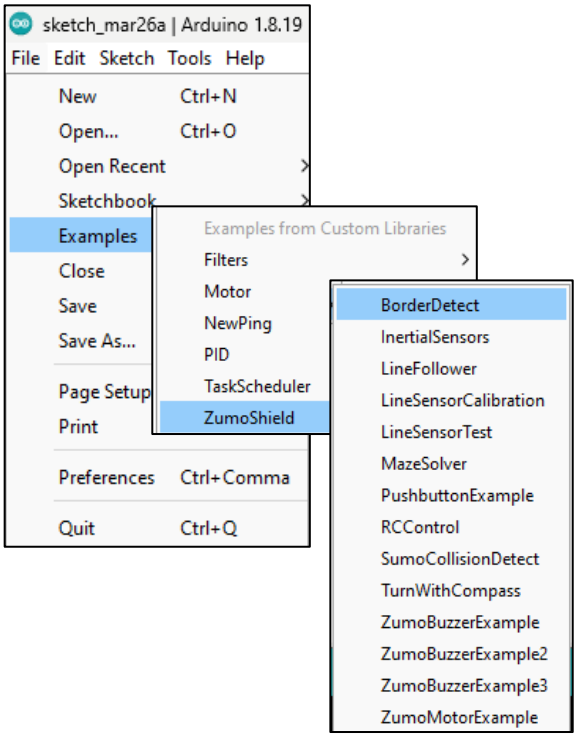
ZUMO WARS

It's now time to connect to the Zumo and start programming!!!

Since we have verified the Arduino is working and we can upload code. it's time to look at how we can get the Zumo robot moving.

Open the example BorderDetect and have a look at the code. As the name implies. this code uses the robots sensors to scan the Zumo ring and detect where the border is so it does not fall off.

While you do not need to understand all the code. there are some sections you should be aware of.




```
// these might need to be tuned for different motor types
#define REVERSE_SPEED     200 // 0 is stopped, 400 is full speed
#define TURN_SPEED        200
#define FORWARD_SPEED     200
#define REVERSE_DURATION  200 // ms
#define TURN_DURATION     300 // ms
```

This section of code describes the speed of the motors. with 400 being full speed. It also shows how long the robot should turn and reverse for expressed in milliseconds (1000 ms to 1s).

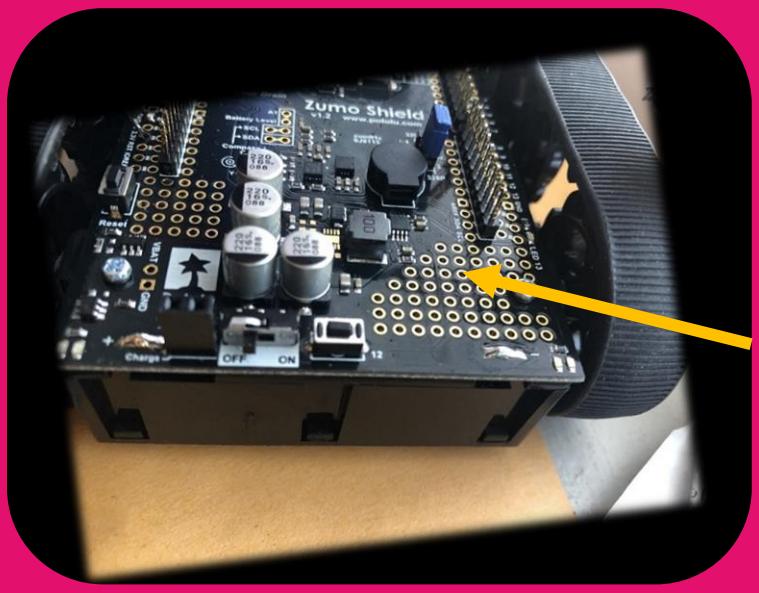
High speed will result in less pushing power. lower speed results in greater pushing power.

Upload this code to the Zumo and take the robot to the Zumo ring to see the robot in action. Experiment with changing these values to see how this changes the robot.

University of Sunderland

ZUMO WARS

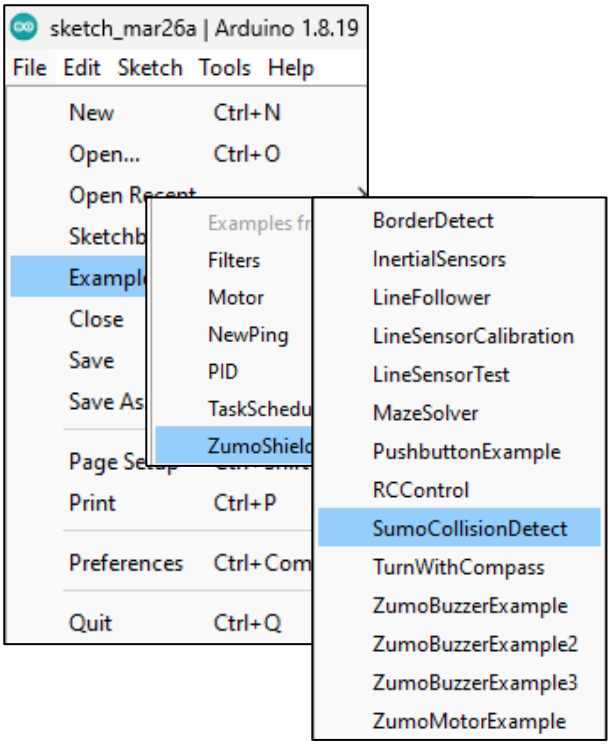
It's now time to connect to the Zumo and start programming!!!



In order to get the robot moving after the code has been uploaded, ensure the robot is turned on. Then set the Zumo down on the starting line of the Zumo ring and press the reset button.

When you are ready to start battleing, open the **SumoCollisionDetect** program, modify the motor values to your preferred setting, and upload your code.

LET BATTLE COMMENCE



University of Sunderland

ZUMO WARS

Championship Rules

- Each Team is required to pick a robot name for the tournament.
- Teams will be chosen at random to face each other in the first round.
- Each round is a "best of three" knockout stage. with the winner progressing onto the next round.
- The Zumo Champion will be the robot that defeats their opponent in the Grand Final.

University of Sunderland

3 2 1 . . .

ACTIVATE

University of Sunderland

This page is intentionally left blank.

University of Sunderland