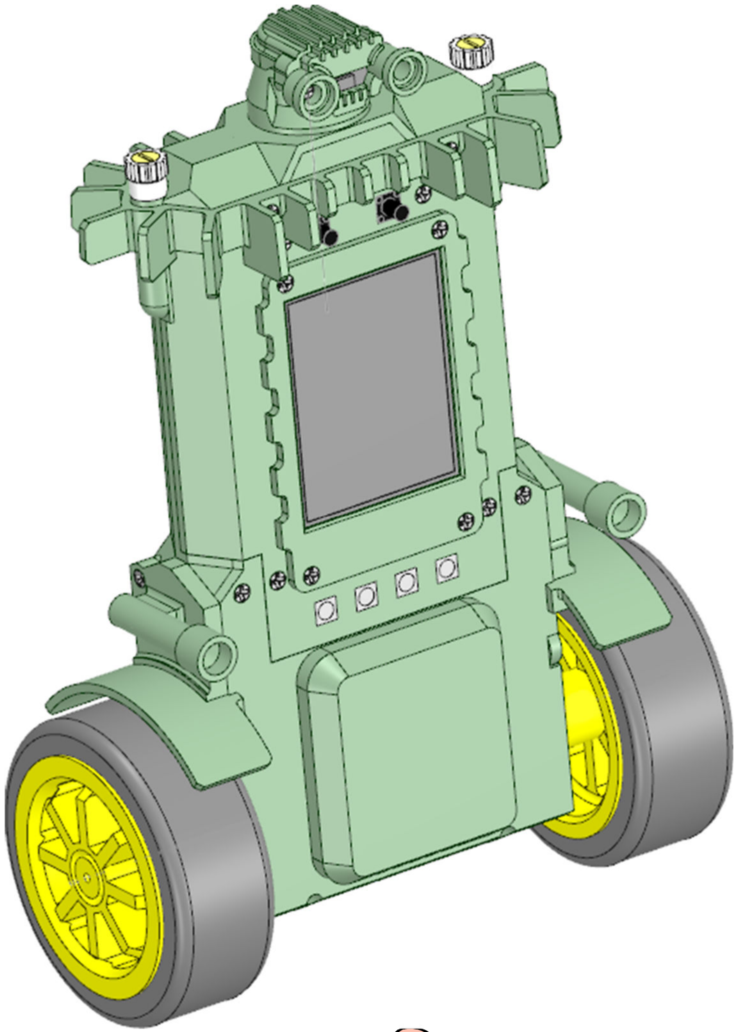
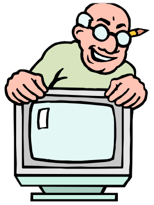
BalanceBot Mk1

Calibration





Calibration of head servo and motors is an essential process.

Issue: 1.0 Released: 30/08/2022 TechKnowTone

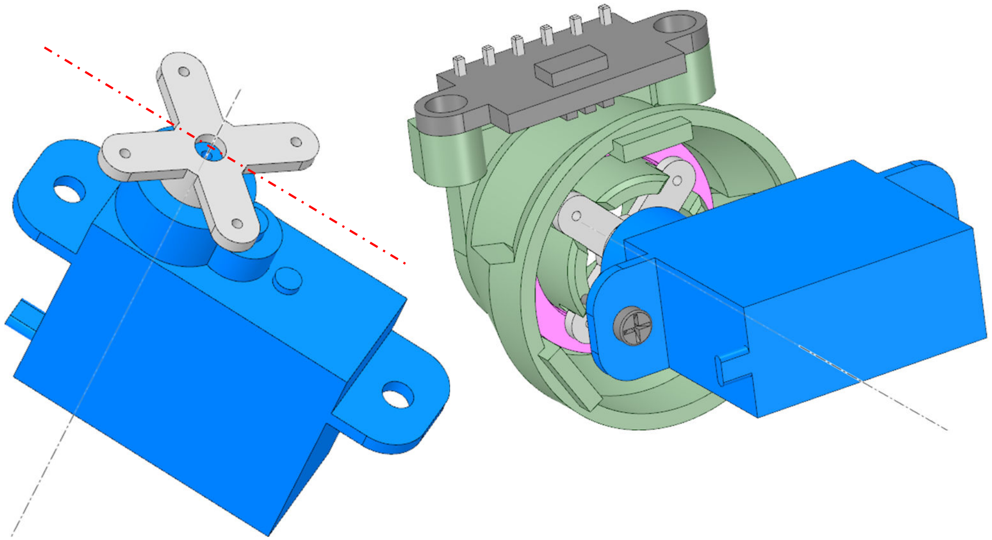
Overview

In this set of procedures we will use a Servo Consistency Tester, and a custom Windows app, written specifically for this project by me.

The servo tester is used to set the centre position, when fitting the cross shaped leaver. This is to ensure that the servo will have sufficient movement range in both directions when mounted in your robot.

The PWM centre position of a servo is 1,500 µs.

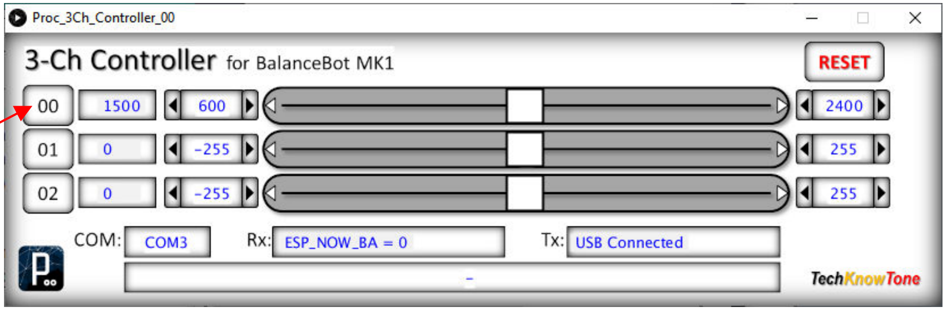
It is simply a matter of powering up the servo tester, plugging in the servo and setting the testers output to 1500. Then fitting the servos cross shaped leaver onto the servos splined drive shaft in the correct position shown here. Given the splined drive shaft teeth positions may prevent you from getting it exactly right, but the nearest position possible will do. This cross shaped leaver mates with recesses in the neck, to drive the head.



Once the head servo is installed in the robot, and the wiring of the chassis is complete, we can then use the 3-channel app to adjust the servo angle, to determine limits to be set in code, and to test the wheel motors to see how much friction there is in their gearboxes.

The channels on this controller correspond to:
00 - head servo angle in microseconds
01 - left-hand motor PWM drive 0 to +/-255
02 - right-hand motor PWM drive 0 to +/-255

The robot needs to be in TEST mode and you need to turn ON a channel, by clicking on the channel button before it will respond to the slider values.



Proc_3Ch_Controller_00

3-Ch Controller for BalanceBot MK1

RESET

00	1500	600		2400
01	0	-255		255
02	0	-255		255

COM: COM3 Rx: ESP_NOW_BA = 0 Tx: USB Connected

-

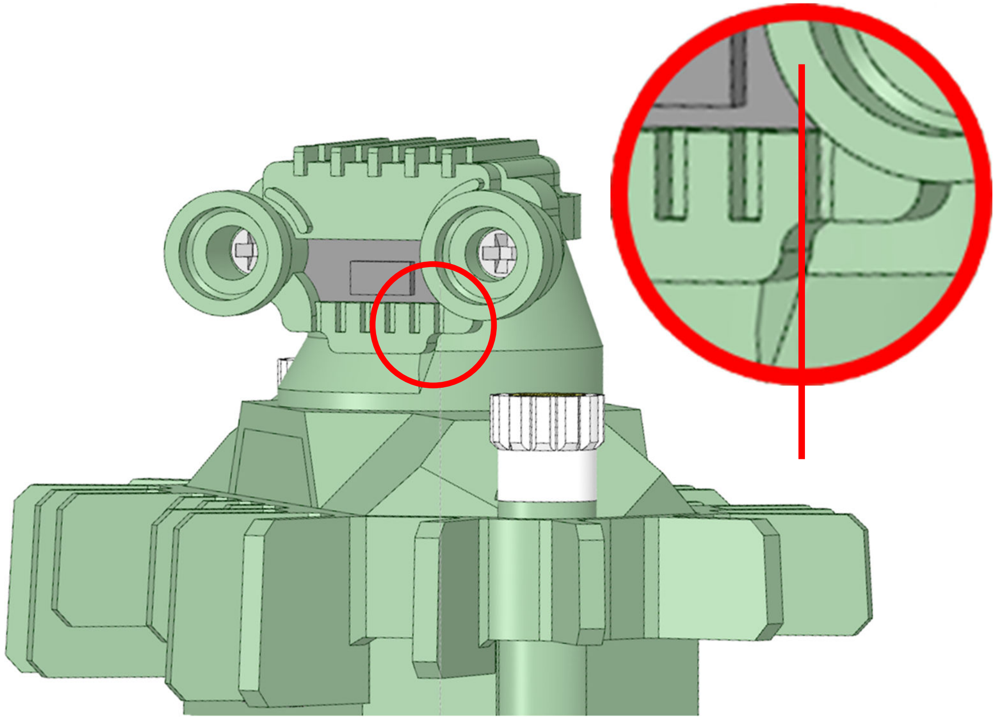
TechKnowTone

Head Servo Calibration

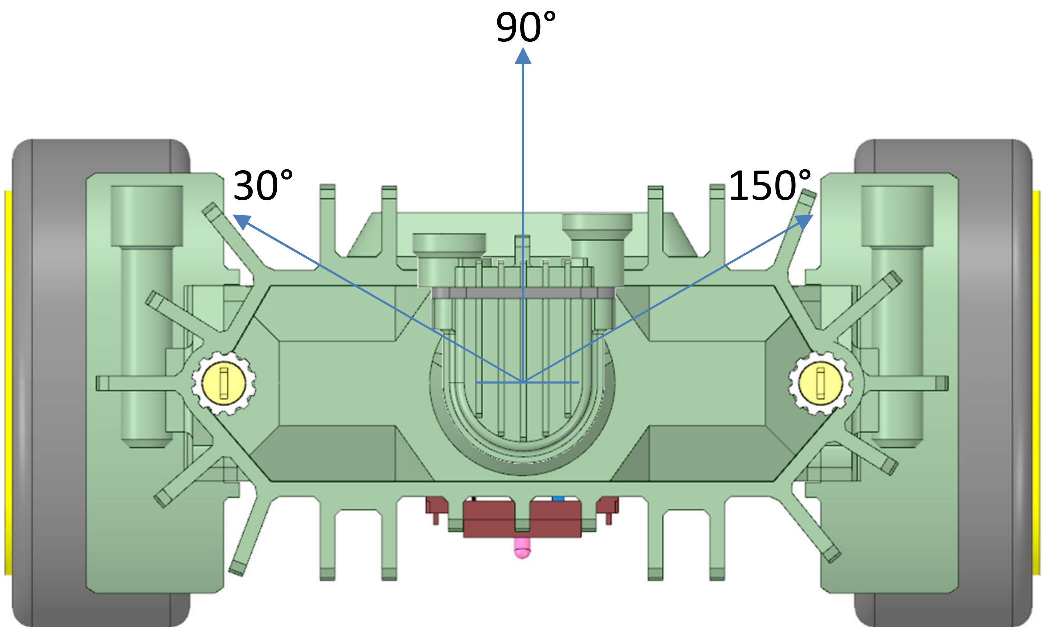
Once the robot is built, set the code into TEST mode and use the 3-channel app to control the position of the head servo, using channel 00.

We want to find the three PWM µs values which set the centre and left/right positions of the head. The left/right positions are approximately +/-60° from centre. Look at the image to the right, which shows the head turned to the left and aligned with the 30° point. The same process applied to the right-hand side.

With the app determine the three values, record them and then enter them into the code in the definition table. Mine we as follows:



```
#define Head_30     885
#define Head_90   1510
#define Head_150 2121
```



Wheel Motor Calibration

With the robot on its stand, set the code into TEST mode and use the 3-channel app to control the motor PWM drive applied through channels 01 and 02.

01 - left-hand motor PWM drive 0 to +/-255
02 - right-hand motor PWM drive 0 to +/-255

Adjust the app sliders to determine the point at which the motors just start to turn. It is likely that this value will be different for each motor, and when driven forward or in reverse. Use the maximum value needed, with which both wheels should start turning, either forward or backwards.

Place this value in your code as follows, under PID controller coefficients, as a floating point variable:

```
float PWM_StartMax = 18.0;        // PWM needed to overcome motor start stiction
```

With that entered into your code, you can now move onto tuning the PID controller using the app I have provided for that. See the separate pdf file for that.