

Appendix-A: The software for calibration

```
#define DOUT 7 //change this pin number to 3 for adjusting lift loadcell and to 7 for adjusting drag loadcell
#define CLK 6 //change this pin number to 2 for adjusting lift loadcell and to 6 for adjusting drag loadcell
HX711 scale;

float cal_factor = -470;

void setup() {
  Serial.begin(9600);
  Serial.println("Press + or - to adjust calibration factor.");
  Serial.println("--Defined pins in this code are for calibrating drag loadcell, lift loadcell make adjustments

  scale.begin(DOUT, CLK);
  scale.set_scale();
  scale.tare();

  long zero_factor = scale.read_average();
}

void loop() {
  scale.set_scale(cal_factor); //Adjust to this cal factor
  Serial.print("Reading: ");
  Serial.print(scale.get_units(), 3);
  Serial.print(" mN");
  Serial.print(" Calibration_factor: ");
  Serial.print(cal_factor);
  Serial.println();

  if(Serial.available())
  {
    char in = Serial.read();
    if(in == '+')
      cal_factor += 1;
    else if(in == '-')
      cal_factor -= 1;
  }
}
```

Appendix-B: The main software

```
#include "HX711.h"
#define DOUT_lift 3
#define CLK_lift 2
#define DOUT_drag 7
#define CLK_drag 6
#define button 8
HX711 scale_lift;
HX711 scale_drag;
float cal_factor_lift = -50;
float cal_factor_drag = -470;
float tot_lift;
float av_lift;
float tot_drag;
float av_drag;
int counter=1;
int counter_2=1;
void setup() {
  Serial.begin(9600);
  while (!Serial) {
```

```

    | ;
    }
    Serial.println("Loading...");
    scale_lift.begin(DOUT_lift, CLK_lift);
    scale_lift.set_scale();
    scale_lift.tare();
    scale_drag.begin(DOUT_drag, CLK_drag);
    scale_drag.set_scale();
    scale_drag.tare(); //Reset the scale to 0
    long zero_factor_lift = scale_lift.read_average();
    long zero_factor_drag = scale_drag.read_average();
}

void loop() {
    scale_lift.set_scale(cal_factor_lift);
    Serial.print("Reading: ");
    Serial.print(scale_lift.get_units(), 5); //printing 5 digits after comma
    Serial.print(" mN_lift");
    tot_lift=tot_lift+scale_lift.get_units();
    av_lift=tot_lift/counter; //average of lift values
    Serial.print(" Average_lift: ");
    Serial.print(av_lift);

    scale_drag.set_scale(cal_factor_drag);
    Serial.print(" Reading: ");
    Serial.print(scale_drag.get_units(), 5);
    Serial.print(" mN_drag ");
    tot_drag=tot_drag+scale_drag.get_units();
    av_drag=tot_drag/counter;
    Serial.print(" Average_drag: ");
    Serial.print(av_drag);

    Serial.print(" Time:"); //running time in seconds
    Serial.print(millis()/1000);

    Serial.print(" Counter:"); //printing number of averaged data
    Serial.println(counter);

    if (digitalRead(button)==1){ //resetting the averaged data and counter
        tot_lift=scale_lift.get_units();
        tot_drag=scale_drag.get_units();
        counter=1;
    }

    counter+=1;
}

```