Realisation of a safety-cage with integrated force sensing for interactive aerial robots

T. (Thomas) van Zonneveld

BSc Report

Committee:
Dr.ir. J.B.C. Engelen
R. Hashem, MSc
Prof.dr. J.C.T. Eijkel

July 2018

015RAM2018
Robotics and Mechatronics
EE-Math-CS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Summary

Aerial robots are getting more advanced every year and their influence in daily life becomes more and more apparent. Most commonly however, these vehicles are not intended for physical interaction with their environment. Yet, the potential of interaction with these vehicles is large. Examples of situations are wind turbines being cleaned by drones or safety operations on hard-to-reach places. Even though safe encased drones do exist, these systems don't allow physical interaction as sensing and control technologies are not implemented. The area of physical interaction between aerial vehicles and their environment is thus a very active and interesting research topic. For these reasons, force sensors have been designed that can be integrated with a safety cage in order to enhance the interaction aspects of drones.

In the presented research, force sensors have been designed by means of strain gauges. A safety cage is realised where these force sensors are integrated, being able to provide useful information regarding interaction forces acting on the entire drone.

By first analysing different layouts of strain gauges and strain gauge characteristics, a final design for a force sensor is proposed. Different designs regarding the placement of these sensors have been analysed in order to propose a final set-up of sensors. With this final set-up in mind, a 3D-design of a safety cage is realised in where these sensors are implemented. Tests with the designed force sensors have been performed and analysed by comparing the results to a commercial 6D force/torque sensor. Results propose a potentially accurate and sensitive system, yet noise, temperature effects and hysteresis are apparent and significantly reduce the accuracy of the sensors.

Contents

1 Introduction

1.1 Context of research

Drones are getting more and more advanced every year. Where drones would hardly be available years ago, they can be bought for a relative small price at any electronics store. Yet, most drones are meant solely for flying purposes, additionally with a cameras attached, and they are not meant for physical interaction with their environment. SPECTORS[1] is an initiative, based upon 31 partners in many different areas, that tries to develop and test new hard- and software solutions, regarding ground based and airborne remote sensing technologies. It comprises different projects, of which the University of Twente is working on physical interactions with drones. This compromises the design and creation of aerial robots that are meant for physical interaction with their environment. Consider for example, a drone with grippers, drones with tilted propellers, or a drone that can exert a force to a surface have already been developed. The purpose of these aerial vehicles is diverse. Instead of sending people up into wind turbines to clean them or to send people for search and rescue operations on hard-to-reach places, airborne vehicles can be used instead. Not only does this decrease the risk of injuries to humans, it could also increase the quality of the work, as humans are restricted in movements due to safety harnesses and other safety equipment. Yet, available aerial vehicles and systems are dangerous and require severe caution when handling. Also, these systems are not implemented with interaction sensing capabilities and thus lack the two main ingredients for an interactive aerial robot, sensing technologies and safety. Although safety cages do commercially exist [2], cages with force integrated sensors do not, as to the author's knowledge.

It is for these reasons that in order to increase the interaction aspects of drones and their environment, force sensors will be implemented in a safety cage to be designed at RAM. Both the sensors as the connection between drone and cage will be designed. The entire system will be designed for a drone that is depicted in figure 1.1

Figure 1.1: Hexacopter to which a system shall be designed

1.2 Problem statement

The research that will be performed during this study includes integration of force sensing into a safety cage for a fully actuated drone, in order to enhance physical interactions with its environment.

This brings rise to multiple research questions; where to place, what type of sensors and how to integrate them into a safety cage. This can be divided into three to be investigated sub parts.

Firstly, in order to measure force, force sensors are needed and thus a first research subject are force sensors themselves. Different parameters of sensors need to be discussed and their influences must be analysed. Secondly, a trivial question is obviously where these sensors must be placed regarding force sensing. What would be the most optimal place to put these sensors and how about orientation? Lastly, these sensors must be placed inside a provided cage, as to enhance safety. What type of connections shall be used and what would be beneficial regarding implementation of force sensors.

Concluding, three research-questions can be identified:

1. What type of sensor shall be used
2. Where to place these sensors
3. How to integrate the force sensors into the cage design

1.3 Goals

In order to evaluate the outcomes of the research, goals must be set.

Must be implemented

- The design of force sensors
- Integrate these sensors into a safety cage

Should be implemented

- Robust cage
- Flexible cage (for impacts)
- Lightweight system
- Force measurements in 3 directions
- Accurate force sensing

Could be implemented

- Torque sensing
- Integrated processing/calibration of forces
- Force measurement in more than 3 directions for extra redundancy
- A physical model of the cage
- Flying demonstration of drone+cage

1.4 Layout of this report

This report is divided in a similar manner as the research has evolved over the time of the project. It starts with chapter 2 in where basic concepts used will be briefly introduced, starting with a quick introduction to strain gauges and wrench transformations. Chapter 3 relates to the first 2 questions proposed in the problem statement. Firstly the sensors are analysed and different parameters are discussed (what type of sensor shall be used). Briefly an amplifier and micro controller are analysed too. Secondly, the second research question is investigated, namely where to place the force sensors efficiently. In chapter 4, the third question is addressed as how to integrate the force sensors with the cage and how to connect the safety cage to the drone. Chapter 5 proposes a set-up to analyse the sensor design made in chapter 3 and results of these experiments are discussed in chapter 6. Chapter 7 contains overall conclusions and recommendations for further research.

2 Background theory

2.1 Strain gauges

Due to the size and ease of implementation of strain gauges, strain gauges will be used as force sensors. This sections tends to explain the basic principles of a single metallic foil strain gauge. Although other types of strain gauges do exist, they will not be analysed. These semiconductor types might be more sensitive, yet they are non-linear [3] and thus introduce more issues than they tend to solve. Additionally, they are not available during the time span of this project and thus not further discussed.

An illustrative image of a foil strain gauge is provided in figure 2.1. A strain gauge is a device that measures strain. Strain is the result of a stress on a material and this stress again is caused by a force. Strain is a "unit-less" (mm/mm or cm/cm) parameter that determines a change in length per unit of length. It is denoted as $\epsilon = \frac{\delta L}{L}$. Stress on the other hand is the force per area and can be described by $\sigma = F/A$. Due to the force applied to a material, the material might be stretched or it might be compressed. For a conducting wire this means a change in the cross sectional area of the wire and in its length. The ratio between the increase/decrease in length and cross sectional area is called the Poisson's ratio and is denoted as ν.
When a wire is stretched, its cross sectional area decreases, its length increases and thus its resistance increases. When it is compressed, the cross sectional area increases, its length decreases and the resistance of the wire goes down. When strain gauges are mounted/glued to a material, forces applied to this material will correspond to changes in the strain gauge. The ratio between the resistance of a wire and length/area decreases/increase is called the resistivity of a wire and has a unit of ohm meter (Ωm). The relation between the resistance of a wire and its length, cross sectional area and the resistivity is shown in equation 2.1

$$R = \rho * \frac{L}{A} \tag{2.1}$$

The ratio between the relative change of resistance and the relative change of length of the wire is known as the gauge factor. It determines the "sensitivity" of a strain gauge and is shown in equation 2.2.

$$K = \frac{\frac{\Delta R}{R}}{\frac{\Delta L}{L}} = \frac{\frac{\Delta R}{R}}{\epsilon} \tag{2.2}$$

Common metallic foil strain gauges have a gauge factor of approximately two, but the exact values differ for each gauge produced. The approximation of the gauge factor is shown in appendix A.1. With this gauge factor, strains can be calculated and they are commonly minute, as shown in appendix A.2. For this reason, a wheatstone bridge shall be used in order to obtain a better/more measurable output.

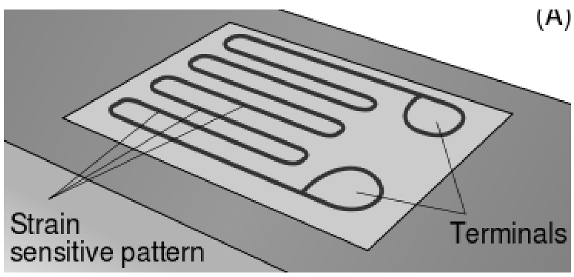


Figure 2.1: Metallic foil strain gauge [4]

2.2 Strain gauge parameters

In order to choose suitable strain gauges, some parameters can be analysed in order to find the best gauges. A few will be introduced [5].

2.2.1 Grid pattern

Gauges exists that are a combination of more than one gauge and measure strain in more than 1 direction. Following the naming convention of HBM (as can be seen on their website as in [6]), single gauges (LY gauges) are gauges that can measure strain in one single direction (as shown in figure 2.1) and XY gauges are gauges that have two single gauges placed perpendicular to each other. XY2 or XY4 gauges are gauges where two gauges are combined in a V-shape, especially designed for measuring torques. Besides those "basic" layouts, full bridge gauges exist in pretty much any combination of the above.

2.2.2 Thermal expansion

When materials heat up, they will expand or contract, depending on the material. When strain gauges are bonded to these materials, they will expand/contract in a similar manner and thus their resistance changes. Not only smart layouts and placing of gauges can cancel these effects (as in section 3.1.1), when it is known how much the material will expand for specific temperature changes (thermal expansion coefficient), gauges are available that are adapted to these changes. Most gauges are only available for common mounting materials, such as steel, aluminium or plastics [3].

2.2.3 Grid resistance

Commonly, most strain gauges available have a resistance of 120Ω, 350Ω or 1000Ω. As mentioned in [7], the change in voltage across a strain gauge depends on the change in resistance, which is linear with the initial resistance value as can be seen in equation 2.2. A larger initial resistance would thus increase the change in resistance and thus output a higher change in voltage. Additionally, larger resistances allow for less heat being generated by the strain gauge, as for the same voltage a smaller current arises and thus the power consumed and generated heat will be decreased [8].

Smaller resistances would be useful in applications where potential electrical interference is apparent and where signal transmissions are of severe importance [8].

2.2.4 Grid length

For very accurate measurements of stress on materials, smaller gauges are better [9]. However, small gauges of a few millimetres are difficult to handle and thus are prone to errors regarding correct alignment on the material. According to [9], when working with homogeneous materials like glass and metal, smaller gauges can be used compared to working with inhomogeneous materials such as carbon or epoxy. When inhomogeneous materials are used, the strain might not be equally divided between the materials and thus a longer gauge is needed. According to [9], [10], the gauge must be at least 5 times larger than the inhomogeneities of the material. Additionally, the larger the gauges are, the better they can dissipate generated heat.

2.3 Wheatstone bridge circuits

A Wheatstone bridge is a circuit consisting of four resistors and an output voltage that is in between those resistors. Two sets of two resistors in series are placed in parallel with each other. Wheatstone bridges allow for accurate measurement of small changes in resistance. An image of a Wheatstone bridge is shown in figure 2.2

It should be noted that the numbering of the resistors inside the wheatstone bridge is viable to different numberings, together with the +/- output of the bridge. The set-up and numbering as shown in figure 2.2 is used throughout the entire report.

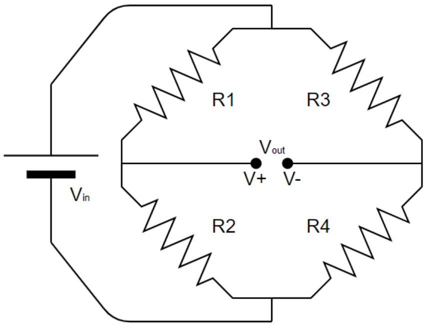


Figure 2.2: Wheatstone bridge

The output voltage of this bridge circuit is 0V when all resistors are of the exact same value. However, when resistor values change, the output voltage does too. This means that when the resistors are replaced by strain gauges, the output voltage will change as the resistances of the strain gauges changes. Since the resistance of the strain gauges is related to a force/strain acting upon it, a Wheatstone bridge thus transforms the force/strain into a measurable voltage difference.

The output voltage of a balanced Wheatstone bridge can be derived by making use of voltage dividers.

$$V_{out} = (\frac{R2}{R2 + R1} - \frac{R4}{R4 + R3}) * V_{in} \tag{2.3}$$

The expression in 2.3 can be rewritten into 2.4, which allows for easier interpretation later on.

$$V_{out} = \frac{R2R3 - R4R1}{(R2 + R1)(R3 + R4)} * V_{in} \tag{2.4}$$

In order to actively use the bridge circuit and to measure strains, resistors in the Wheatstone bridge have to be replaced by strain gauges. This can be done by using 1 gauge, 2 gauges or 4 gauges, depending of the direction of forces to be measured. Looking at figure 2.3, a parallel strain is measured in the X direction while a bending strain is measured in Y or Z direction. By making use of expression 2.4, some basic bridge circuits can be analysed in a bit more detail.

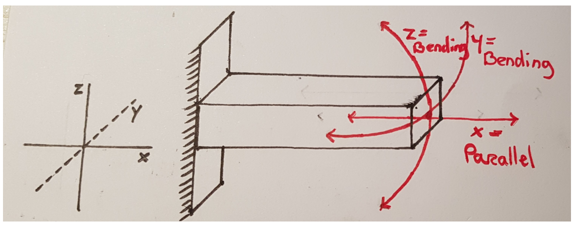


Figure 2.3: Orientation of bending/parallel strains

2.3.1 Single gauge

Using a single strain gauge (quarter bridge) of which the resistance increases or decreases with ΔR and by placing this strain gauge as resistor 1 in figure 2.2 while keeping all the other resistances equal (having a value of R_0), the output of the bridge can be derived, as shown in appendix B.1, where $V_r = \frac{V_{out}}{V_{in}}$.

$$V_{out} = \pm\frac{1}{4}V_{in}\frac{\Delta R}{R_0} \tag{2.5}$$

2.3.2 Two gauges

Instead of using a single gauge, two gauges can be used in order to increase the sensitivity (half bridge). Firstly (case 1), the gauges can be placed as resistors R1 and R4 (or R2 and R3) and measure a parallel strain. Here they should have the same sign and thus both contract or expand. If not, no output change will occur. Case 1: $R1 = R4 = R_0 + \Delta R$ and $R2 = R3 = R_0$ (or vice versa).

Secondly (case 2), the gauges can be placed as R1 and R2 (or R3 and R4) and they should have opposite signs. Case 2: $R1 = R_0 + \Delta R$, $R2 = R_0 - \Delta R$ and $R3 = R4 = R_0$ (or vice versa). They thus measure a bending force with both gauges placed on opposite sides.

For case 2 it is clear that the gauges must be at different sides of the rod and only a bending force will cause a change in output voltage. For case 1 it is trickier. For case 1, the gauges can both be placed on the same side of the bar or they can be placed on opposite sides. In both cases, a parallel strain will be measured by the gauges. However, when both gauges are placed on the opposite sides of the bar, a bending force will cause one of the two (e.g. R1) gauges to expand and the other (R4) to contract. A change in output voltage will occur, compared to no output change when both are placed on the same side.

For both options, the input/output voltage relation can be derived and is shown in appendix B.2. In both cases, the output voltage is equal to 2.6

$$V_{out} = \pm\frac{1}{2}V_{in}\frac{\Delta R}{R_0} \tag{2.6}$$

2.3.3 Four gauges

To enhance an even higher sensitivity, all four resistors could be replaced with strain gauges (full bridge). As can be seen in equation 2.4, this only gives a single option placing gauges. R2R3 should increase while R4R1 decreases. Thus, R2=R3 is contracting while R1=R4 is expanding. Since now both pairs of gauges have a different sign, this would be viable only when a bending force is applied and both couples of gauges are on different sides of the bar, implying $R2 = R3 = R_0 + \Delta R$ and $R1 = R4 = R_0 - \Delta R$ (or vice versa). The derivation is shown in appendix B.3

$$V_{out} = \frac{\Delta R}{R_0}V_{in} \tag{2.7}$$

2.4 Wrench transformations

As multiple sensors measure forces in their own reference frame, they must be converted to a common frame of reference. Since each sensor measures relative to its own frame of reference, each individual result must be transformed individually.

Since forces and torques are the main interest, a wrench is of most importance. A wrench is shown in 2.8. Here, T stands for a torque and F for a force. It should be noted, this wrench is written as a row vector, therefore the r in superscript.

$$W_r = \begin{pmatrix} T_x & T_y & T_z & F_x & F_y & F_z \end{pmatrix} \tag{2.8}$$

Additionally, a twist is defined as rotational velocity and linear velocity, as shown in equation 2.9. Here the twist is defined as a column vector.

$$T_c = \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_y \\ v_x \\ v_y \\ v_z \end{pmatrix} \tag{2.9}$$

In order to transform the wrench from one frame to another, the power continuity property is used as total power exerted in frame A must be equal to the power exerted in frame B.

$$^{A}W^{A}T = {}^{A}P = {}^{B}P = {}^{B}W^{B}T$$

This ultimately boils down to the expression in 2.10

$$^{A}W_c = Ad^{T}_{H^B_A}\,{}^{B}W_c \quad where \quad Ad_{H^B_A} \equiv \begin{pmatrix} R^B_A & 0_{3\times 3} \\ {}^{B}\tilde{o}_A R^B_A & R^B_A \end{pmatrix} \tag{2.10}$$

In order to obtain the wrench in frame A, the wrench in frame B must be multiplied with the transpose of the adjoint matrix $Ad_{H^B_A}$. This adjoint matrix is build from the homogeneous H-matrix, 2.11,

$$H^B_A = \begin{pmatrix} R^B_A & {}^{B}o_A \\ 0 & 1 \end{pmatrix} \tag{2.11}$$

In the adjoint matrix, R^B_A is the rotation matrix from frame A to frame B, $0_{3\times 3}$ is an empty matrix and ${}^{B}\tilde{o}_A$ is the tilde form of the distance vector from the origin of frame A to the origin of frame B, expressed in frame B. The tilde form is defined as in 2.12.

$$^{B}o_A = \begin{pmatrix} o_x \\ o_y \\ o_z \end{pmatrix} \Leftrightarrow {}^{B}\tilde{o}_A = \begin{pmatrix} 0 & -o_z & o_y \\ o_z & 0 & -o_x \\ -o_y & o_x & 0 \end{pmatrix} \tag{2.12}$$

When multiple wrenches must be transformed in order to obtain a single wrench in the new frame, they can simply be added, as shown in in equation 2.13. Note, here the wrench is written as a column vector.

$$^{A}W_c = \sum_{i=1}^{n} Ad^{T}_{H^i_A}\,{}^{i}W_c \tag{2.13}$$

The complete derivation and a slightly more extensive introduction is given in appendix C.

3 Analysis

This chapter is mainly to analyse different options regarding force measurement. Firstly, the layout of the sensor will be discussed. Secondly, amplifiers and micro-controllers are quickly analysed. Thirdly, the placement of sensors is discussed.

3.1 Sensor lay-out

3.1.1 Dummy strain gauges

As with strain gauges and depicted in chapter 2.3, two types of forces can be measured. Bending forces are perpendicular to the strain gauges while a parallel force is in the same direction as the strain gauges, which is illustrated in figure 2.3.

As in chapter 2, it can be concluded that a full bridge circuit is 4 times more sensitive than a quarter bridge circuit. However, this full bridge circuit can only measure bending forces and is not sensitive to a parallel force. Instead, to measure a parallel force, only the half bridge circuits with two strain gauges will work. Two cases with 2 gauges have been discussed:

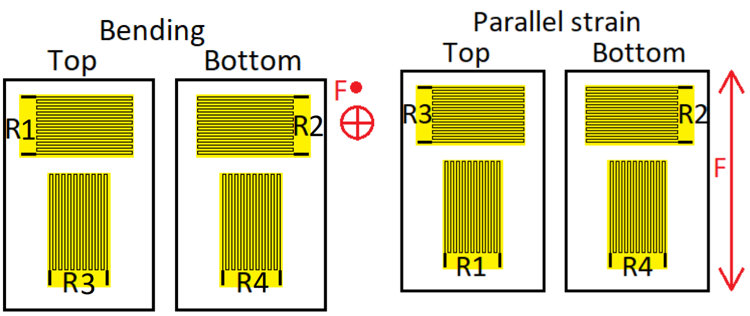
case 1 :$R1 = R4 = R_0 + \Delta R$ and $R2 = R3 = R_0$
case 2: $R1 = R_0 + \Delta R$, $R2 = R_0 - \Delta R$ and $R3 = R4 = R_0$

There is however an issue regarding temperature changes. When a temperature change occurs on one side or a specific strain gauge, these gauges will expand/contract while this is not compensated for by the other gauges. This thus results in false outputs.

Assuming the full bridge for bending as in 2.3.3, R2 and R3 are placed on one side while R1 and R4 are placed on the other. When a temperature change occurs on one side of the bar and the resistance of R2 and R3 increases, this will not be compensated for and the output voltage will change, as can be seen in equation 2.4

$$V_{out} = \frac{R2R3 - R4R1}{(R2 + R1)(R3 + R4)} * V_{in} \tag{2.4}$$

To compensate for this temperature effects, both bending and parallel strains need a full bridge layout with dummy gauges implemented as shown in figures 3.1a and 3.1b. Not only do these layouts in perfect circumstances eliminate temperature changes, for a parallel strain it also increases the sensitivity compared to using 2 gauges only [3], [7], [11], [12].



(a) layout for a bending strain **(b)** layout for a parallel strain

Figure 3.1: Temperature independent full bridge circuits

Looking at figures 3.1a and 3.1b, it can be seen that for a bending force R4 will increase while R3 will decrease. The difference compared to case 2 is that now also R1 will increase and R2 will decrease with the poissans ratio. Looking at equation 2.4, it can be seen that this indeed results in a higher output than only using two gauges. It can also be observed that when a temperature change occurs on a single side of the bar, the resistance R1 and also R3 increases (or decreases). Now, due to the opposite signs in the output voltage equation, no change will be measured (assuming all resistances are of equal value). For a parallel strain, both R1 and R4 measure the same strain and have the same sign. When these two increase due to a force, R3 and R2 will decrease and thus a higher output is obtained compared to using only two gauges. When temperature changes on one side, R1 and R3 might increase but the output will still be stable.

With respect to temperature compensation, it can be seen that this only works when all resistances are of exactly equal value. If so, the numerator equals to zero and no output is measured. However, if the resistances are not all equal and thus the numerator is not equal to zero the denominator can not be ignored and will cause output changes.

3.1.2 Final sensor lay-out

As the final cage must measure forces for interaction purposes, both torques(bending) and parallel forces could be measured. It is chosen to start implementing sensors for parallel forces first, whereas torques could be implemented later if time permits. Since it is chosen to start with the implementation of parallel forces first, the full bridge circuit for parallel forces as shown in figure 3.1b shall be used. This circuit has, for parallel strains, the highest sensitivity, included is temperature compensation and no bending forces will be measured.

Linear (LY) gauges will be used with a resistance of 120Ω. In order to increase preciseness, a XY gauge with a 90 degree angle between two gauges would prove useful, but, due to delivery times and the scare availability of these gauges, single linear gauges are chosen instead. It is also for this reason that strain gauges with an resistance of 120Ω are picked although 350Ω gauges are preferred. Similarly, available gauges are designed for materials of which the expansion coefficient is positive while that of the to be used carbon fibre rods is almost zero, some even negative (it depends on the type of resin, fibres and the ratio between those) [13], [14]. Due to the lack in availability of gauges, this choice can not be made and instead gauges must be picked that are optimized for other materials. A choice that can be made however is the length of the gauges. Since carbon is used and since this is an inhomogeneous material, a long gauge is preferred. By a quick measurement of the carbon material, fibres can be seen with a width of 2mm, implying a grid length of 10 mm.

Concluding, a layout for a parallel strain as shown in figure 3.1b will be implemented. Strain gauges with a resistance of 120Ω and a grid length of 10mm will be used, resulting in a total length of 13mm for the total gauge. The gauges used will have lead wires pre-soldered and its data-sheet is shown in appendix D.

3.2 Amplifier + micro controller

As the output voltage across the Wheatstone bridge is still small, an amplifier is used in order to amplify this signal. In order to convert these voltages into a digital value, an analog to digital converter (ADC) is necessary. Lastly, a micro controller is used in this project in order to process the obtained data.

Although potential other options exist, for the microprocessor the choice is made between an Arduino and a Raspberry Pi. Both boards would be able to perform well, however, previous experience has learned that it is easier to work with an Arduino rather than a

Raspberry if no external hardware is needed. It is therefore chosen to work with an Arduino Uno.

The choice and potential options for an amplifier however, are bound to constraints. Since the entire system is mounted to a drone, it should be light, not to big in size and preferably robust. Unfortunately, strain gauge bridge amplifiers sold by big strain gauge manufactures like HBM and Omega are often too big in size and their weight is not suitable for a drone.

Instead it is chosen to use the HX711 analog to digital converter [15]. The main reason for this decision is due to the available software and its size/weight. This ADC has a programmable gain of 32, 64 or 128, an adjustable sampling rate of 10Hz or 80Hz, a differential input range of 20mV or 40Mv (depending on the gain) and an output of 24bits in 2's complement. The exact specifications can be found in the datasheet, as in appendix E

3.3 Sensor placement

In order to analyse where to place the sensors relative to the drone its frame of reference, the total wrench of all sensor is combined and analysed for a few different layouts. This is done by using the condition matrix. Different cases of sensor layouts are analysed, where-after different condition numbers will be derived and compared.

3.3.1 The condition matrix

In order to obtain the total wrench in frame A, wrenches in all other frames must be multiplied with their adjoint matrix, of which the results are summed, as was shown in equation 2.13.

$$^{A}W_c = \sum_{i=1}^{n} Ad^T_{H^i_A} \, {}^{i}W_c \tag{2.13}$$

Regarding transformations, the sensor's frame of reference can be rotated with respect to the original frame of the drone, so that it matches with the sensors orientation. This thus implies that there will be no transformation between the wrench and the sensors frame of reference, but only in the rotation matrix between the two reference frames. The wrench will simply be the wrench measured in the new frame while the rotation matrix consists of sine and cosine functions. To make this clearer, figure 3.2 is added. The wrench of sensor 1 and 2 is measured directly along the frames axes, while the wrench measured by sensor 3 will need a transformation to its own frame by sine and cosine functions, before being transformed to the base frame.

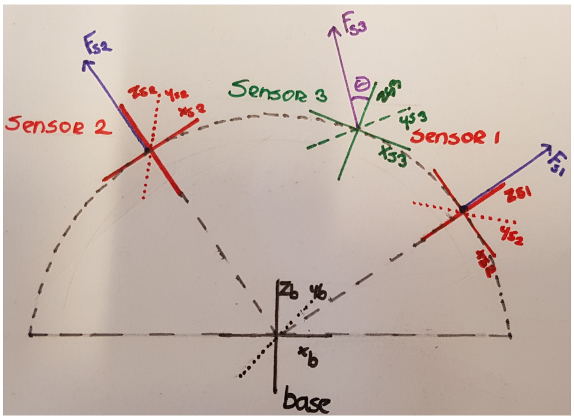


Figure 3.2: Wrench transformations

Following this approach, all adjoint matrices contain a rotation matrix that is a genuine 3D rotation matrix filled with sines and cosines. The wrench is just a force measured in one single direction since linear gauges are used and torques are not implemented. To make an easier calculation later on, the force direction of each sensor is set equal for all sensors. All sensors will measure in the same Z direction of their own personal frame of reference. All the transformations and rotations are compensated for inside the rotation matrix.
This thus implies that the equation as in 2.13 can be transformed to a form, similar as in [16].

$$W_A = C_{6\times n} S_{6\times 1} \tag{3.1}$$

Where C is a $6\times n$ matrix, where n is the number of sensors and where S is the strain vector containing all measured strains by all sensors. Of this matrix C the condition number can be determined [17]. This condition number of a matrix determines how "sensitive" the output of a linear equation is with respect to a small change in the input. The output error is thus equal to the input error times the condition number, implying this number should be as close to 1 as possible [18]. It is calculated using the cond() function in Matlab, returning the ratio of the largest and smallest singular value of the matrix C.

With this number and the above mentioned C matrix, different sensor placements can be tested for their sensitivity to small changes in orientation. When strain gauges are glued on the carbon rods, 100% precise orientation might not be achieved due to inaccuracy of the human hand. Strain gauges might thus not be placed exactly as planned and errors in alignment might arise. Different cases for sensor placement are analysed and their condition number is compared.

3.3.2 Cases

Different layouts will be analysed using a provided Matlab script. This script is based upon a few steps and the entire script is provided in G.2. The first step is to define the sensors by defining their reference frame and their distance from the origin. From these definitions, the H-matrix is determined and this H-matrix is then in a second step used to determine the adjoint matrix. Since that all sensors measure in their Z axis, relative to their own frame of reference, the entire adjoint matrix can be discarded except for the last column of this matrix. From these vectors, a new matrix can be constructed that is equal to the C matrix as in equation 3.1. Since no torques are of importance, only the last three rows of the $6 \times n$ matrix are important. These rows are than combined into a new matrix of which the condition number is determined. Additionally, a sphere is plotted where frame of reference of each sensor is shown.

Although in principal many different layouts could be tested, as stated in [17] only a cube/rectangular design, a tetrahedron- and an octagonal-design prove viable. A fourth case is implemented for more redundancy. In order to ensure the most strain to be measured, the force sensors shall be placed as close to the outer cage as possible.

Tetrahedron(1)

This first design is not mentioned in [17] but is merely meant for redundancy and getting acquainted with the script. Two sensors are placed directly below and atop the origin while the other three are placed 120 degrees from each other in the same plane. Totally the force is measured on 5 locations. The Matlab sketch is shown in figure 3.3a and 3.3b. The condition number for this design is 1.1547.

Tetrahedron(2)

This design is similar to the first design, except that this design is mentioned in [17] and the sensors are placed slightly different. All angles between all frames are equal and are 120°. It is shown in figures 3.4a and 3.4b. The condition number is 1.2472.

Cube/rectangular

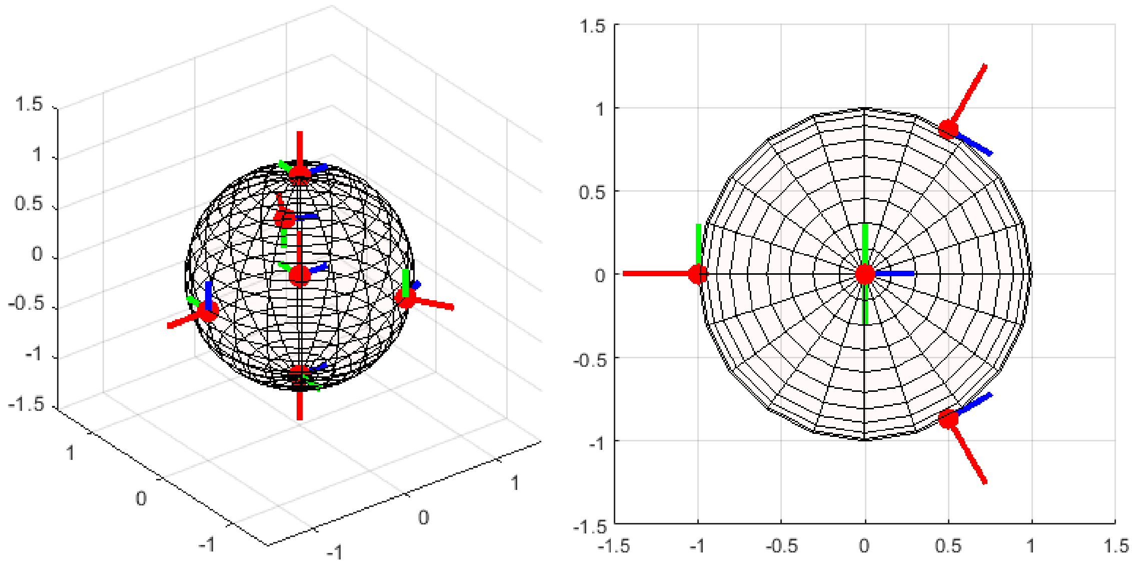
All angles between all sensors are equal and are 90°. It is an elongation of the reference frame in the origin as shown in figures 3.5a and 3.5b. The condition number is 1.

Octagonal

For this layout, eight sensors are used. Four in the upper half and 4 in the lower half. In both halves, the angles between sensors is 45°, implying that both halves are equally divided by the four sensors. It is shown in figures 3.6a and 3.6b. The condition number is 1.4142.

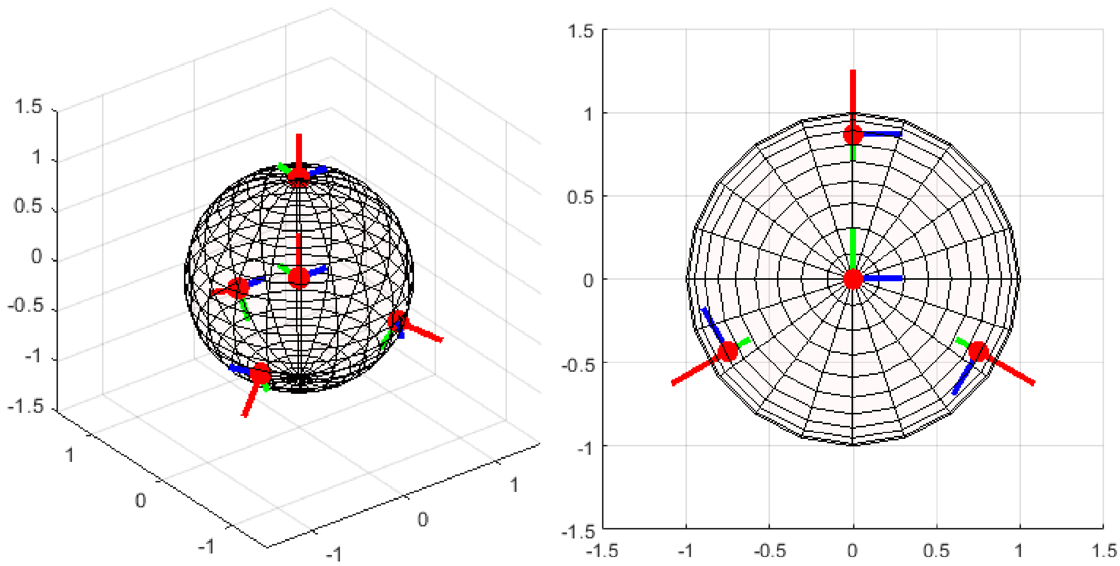
3.3.3 Final sensor placement

As mentioned, the lower the condition number, the less prone the output would be to errors. It is therefore decided to pick the rectangular design. Also, due to the fact that this design implies right angles, the difficulty designing an outer sphere and connecting elements to this sphere is decreased, as will become clear in 4.1.



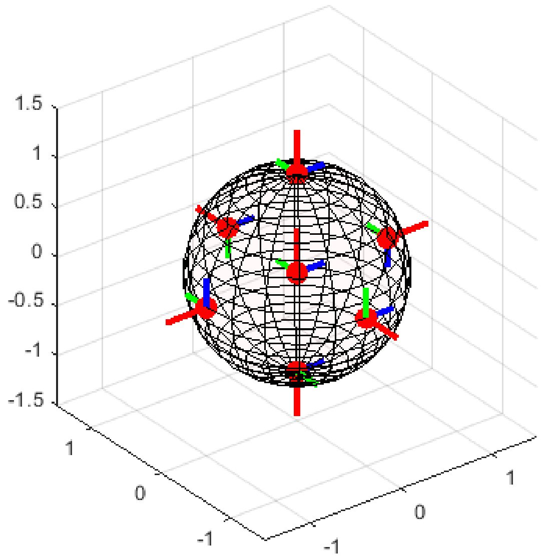
(a) Side view **(b)** top view

Figure 3.3: Tetrahedron design 1 (Condition = 1.1547)

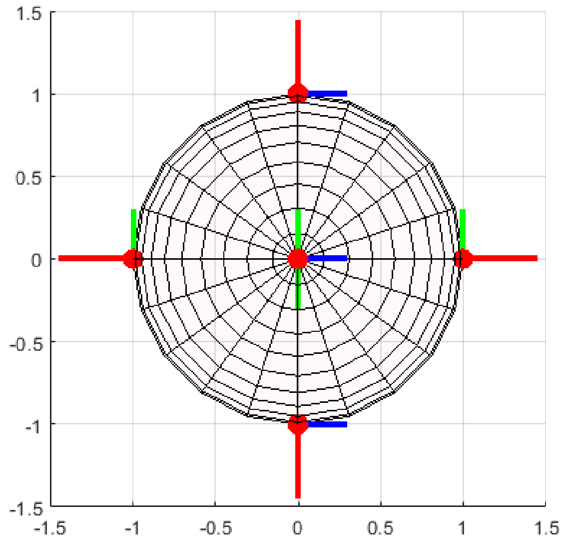


(a) Side view **(b)** top view

Figure 3.4: Tetrahedron design 2 (Condition = 1.2472)

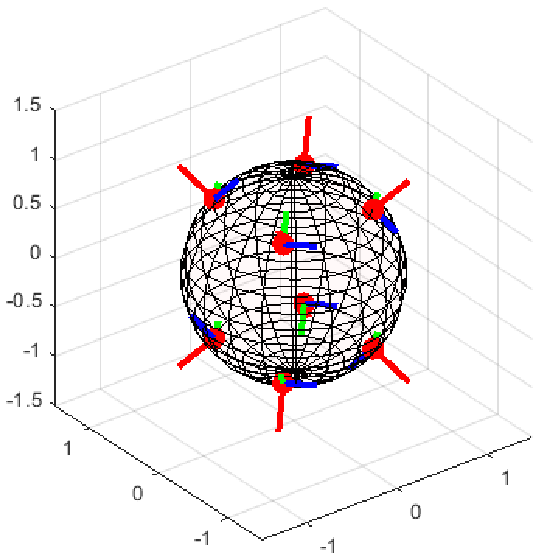


(a) Side view

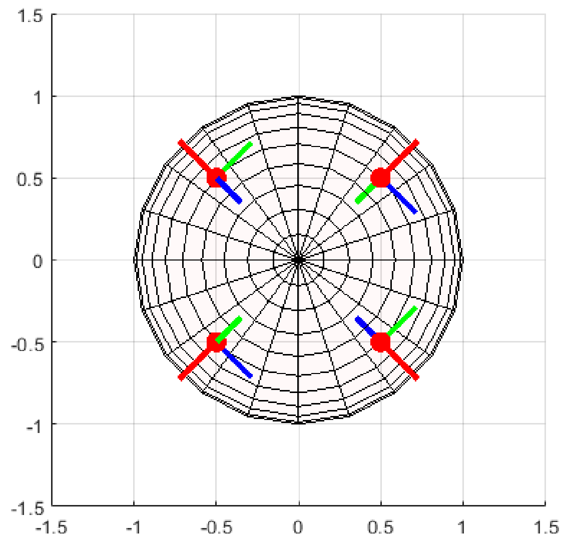


(b) top view

Figure 3.5: Cube/rectangular design (Condition = 1)



(a) Side view



(b) top view

Figure 3.6: Octagonal design (Condition = 1.4142)

4 Sensor implementation

As the placement of the sensors has been identified, they have to be implemented into a cage. The total system consists of an outer cage that is provided, that via a to be designed "inner structure" or frame is connected to the drone itself. The designs are made with Solidworks [19]. As for the materials, carbon and ABS will be used. Carbon is lightweight and flexible, yet it is extremely strong. This will be used for the frame and the outer cage of the drone. ABS will be used for the connectors needed for the drone and the cage. Although other plastics do exist, the available 3D-printer prints in ABS and thus all final 3D printed parts will be made out of ABS. Dimensions of parts and the system are provided in appendix H.

4.1 Outer sphere

For the outer sphere, a design of an icosahedron shaped outer sphere of thin circular carbon fibre bars is provided and is shown in figure 4.1. This icosahedron design allows for easy assembly and connection to the frame, as all bars inside this icosahedron are of equal size and on all sensor positions the bars are horizontal or vertical. This is clearer to see in figure 4.2.

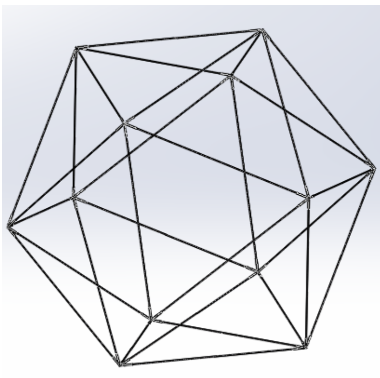


Figure 4.1: Provided 3D-model of cage design

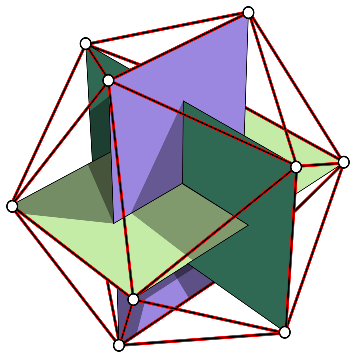


Figure 4.2: Illustration of icosahedron cage

4.2 Inner structure

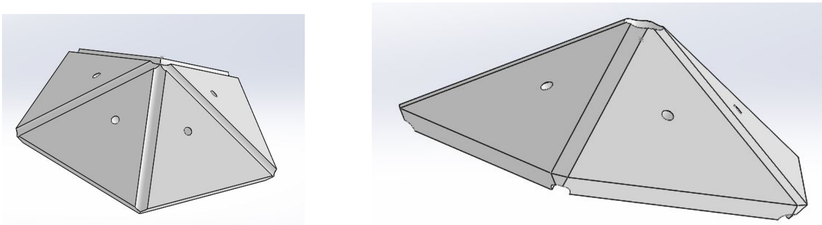
For the inner structure, it is designed to use (six) rectangular carbon fibre rods to support the drone, each sensor being placed upon its own rod. They are rectangular in order to fit the strain gauges used. All bars, except for the bar upwards, are coming together below the base plate of the drone. The drone is thus supported by the square bars below and does not rests on the cage itself. The bar going upwards is set on top of the drone and is not connected in the same point as the other bars. The main reason for this design is due to physical assembly and adjustments being easier accessible from the outside. The frame can be seen in the final design, as shown in figure 4.7

4.3 Connections

In order to keep the entire system together, 3 connecting elements are necessary. First are the connectors between the corners of the cage (cage to cage connectors). These need to hold five small rods of which the frame is constructed. Second is the connector between the square (supporting) rods and the drone (drone to frame connectors). Thirdly are the connectors between the square rods/frame and the cage itself (drone+frame to cage connectors).

4.3.1 Cage to cage connectors

The cage to cage connector has been predesigned previously and is made up from two parts. It consists of a bottom part and an upper part that are screwed together. In between the carbon rods are locked by friction. It is shown in figure 4.3. An earlier design has been revisited as this design proved not to be strong enough (it can be seen in figure 4.1)



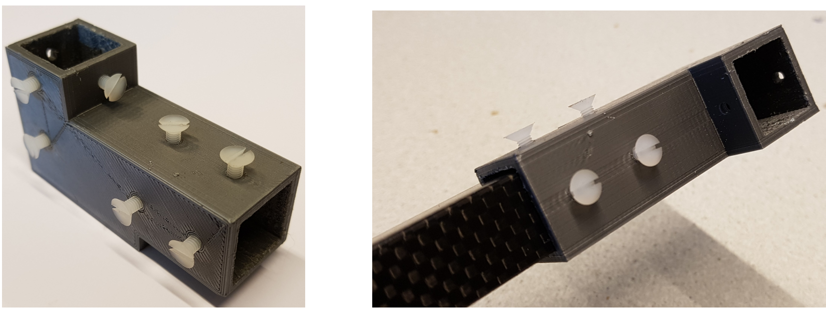
(a) Bottom part **(b)** Top part

Figure 4.3: Cage to cage connector

4.3.2 Drone to frame connectors

To connect the drone to the frame, a connector had to be designed. A design with two separate connectors was created. The first connector is placed below the drone and supports five rods for the frame. The second connector is placed on top of the drone and holds a single rod.

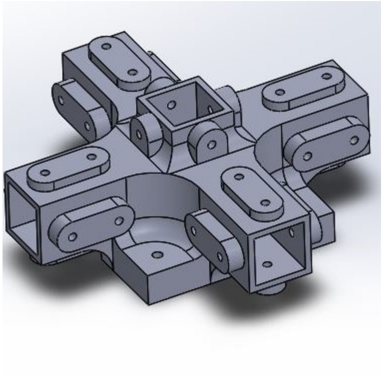
For the bottom connector some designs have been considered, but the basic idea was similar: a square block with on 5 sleeves on the sides for the bars. A first design had sleeves in where the rods would perfectly fit. Unfortunately, as it turned out that all the rods were slightly different in dimensions (they were square and supposed to be 15mm wide, but they had a slight deviation ranging between 14.8mm and 15.4mm), this precise fit could not be designed. To keep the rods in place, instead, a connection with screws has been designed. Screws can be used to exert a force on the rods to push them in place in one corner of the slot. A part of this design was printed and is shown in figure 4.4. As became clear from the print, screws did not have enough screw threat to exert enough



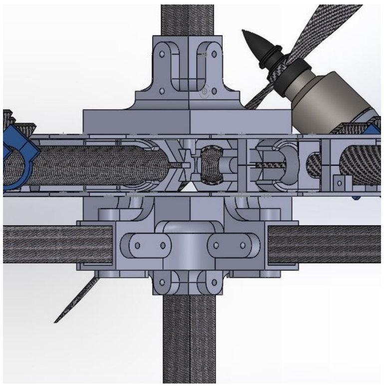
(a) Side view **(b)** One square rod connected

Figure 4.4: First print

force without breaking and thus material had to be thickened. Additionally, screws made scratches on the carbon material and thus damaged the carbon. In order to deal with this, two small and thin pieces of spring steel are placed between the connector/screws and the carbon rods. This not only reduces damaging the carbon, it also divides the force more equally over the carbon rod. These changes have been incorporated in a new design where additional material for screws is added. An almost identical design, meant for a single rod instead of five, is chosen for the top connector, as shown in figures 4.5a and 4.5b.



(a) Connector



(b) Side view including top connector

Figure 4.5: Drone to frame connection

4.3.3 Drone+frame to cage connectors

Due to the fact that all square supporting bars meet the cage at a horizontal or vertical rod, this connection is rather simple compared to a case where the cage bars would be under an angle. To connect the cage to the frame, a similar concept as for the cage to cage connectors is used, consisting of two parts in between the rods are kept on place using friction. The first part is connected directly to the square bars and is almost similar to the top drone to frame connector. The second is screwed upon the first part. In between are the circular cage bars. This design is shown in figure 4.6 where, for clarification, a small gap is kept in between the two parts.

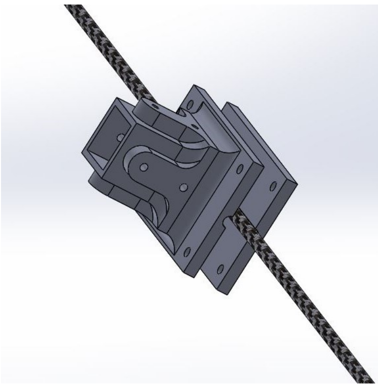


Figure 4.6: Cage to drone connector

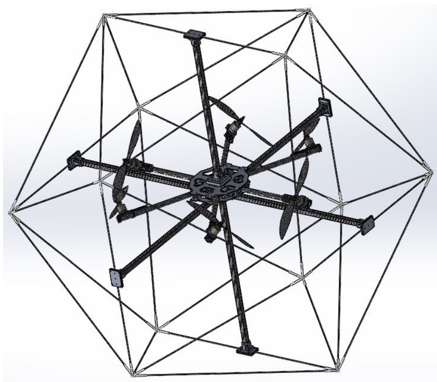


Figure 4.7: Final design (with old connectors)

4.4 Final sensor implementation

A final design to implement sensors into the cage is shown in figure 4.7. Due to "over-constraints" in Solidworks, the cage to cage connectors could not be added to the cage design. Instead, the unrealisable connectors are shown. Dimensions of parts can be found in appendix H

A physical model of this design has been created using 3D-printed parts (ABS) and carbon fibre rods and this final model is shown in figure 4.8. No rotors and electronics have been added to the drone, only the frame itself is connected. The weight of all individual parts is provided in table 4.1. With the physical model, it can be concluded that almost all parts provide a nice fit. Unfortunately, the sleeves of the cage to cage connectors had to be slightly increased in order to create an easy fit.

Table 4.1: Final weights (screws/bolts included if applicable)

Part	Weight [g]	Quantity	Total weight [g]
Drone to frame connector (bottom)	101	1	101
Drone to frame connector (top)	44	1	44
Cage to cage connector	52	12	624
Drone+frame to cage connector	38	6	228
Square rods incl. Hx711/strain gauges (frame)	66	6	396
Circular rods (cage)	7	30	210
Spring steel	32	-	32
Total			1635

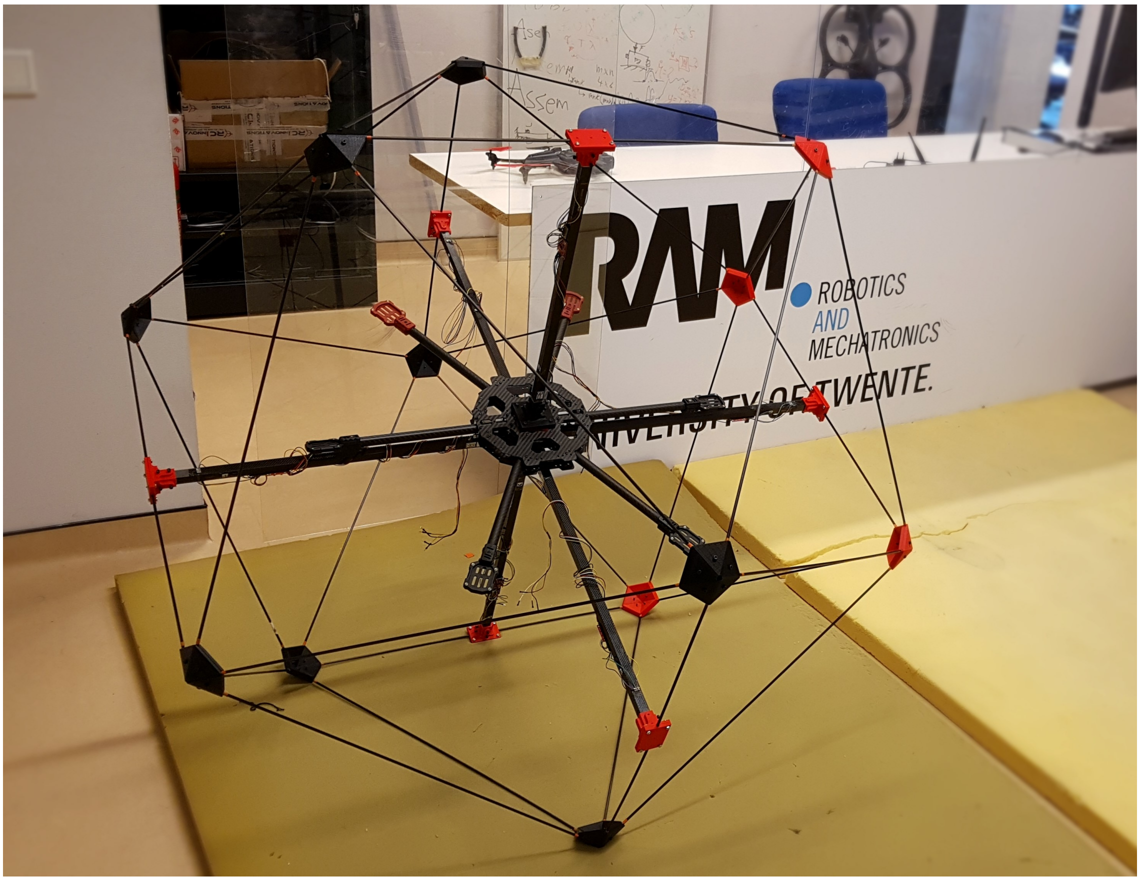


Figure 4.8: Physical final cage including drone frame (without rotors/electronics)

5 Experimental setup

5.1 Overall setup

A setup has been created in order to analyse the strain gauge system and to test its capabilities for measuring strains. This set-up consists, of a fully ready/functional rod with sensors placed on it as analysed in section 3 - four strain gauges placed for parallel strain as in figure 3.1b, the HX711 amplifier and a wheatstone bridge wiring connector to connect the strain gauges in the correct way. The strain gauges are glued by hand using cyanoacrylate (Loctite) and the wheatstone connector is a piece of PCB where strain gauges and inputs/outputs are soldered to. The gauges are connected via a small strain reliever to ensure no broken gauges during soldering. The entire setup is wired as depicted in appendix F.

An Arduino Uno is used to read out data from the HX711 and this data is stored using software called Tera Term [20]. The Arduino will be powered by an external 12V/1A adapter. The setup is placed inside a drone flight lab at RAM. Temperature is consistent at around room temperature. Furthermore, although six bars are ready to be tested, the main analysis will be performed with rod 2 only.

In order to analyse the output of the strain gauge sensor and to compare it to known forces, a fully calibrated force/torque sensor is used, namely the Mini40 from ATI [21]. It is mounted to measure in de negative Z direction. In order to connect the Mini40, a connector has been designed that is almost similar to the top part of the drone to frame connector as in figure 4.5b. A similar part can be placed on top of the rod in order to apply loads to the setup. These loads can be exerted by a bucket that can be filled with sand, while hanging down the setup with a nylon wire. The Mini40 will give known forces while the strain gauges will provide an un-calibrated ADC output. The final setup without bucket is shown in figure 5.1.

For another experiment, a ventilator is connected to the rod. This is connected in such a way that the air flow is not directly on the gauges, but it is directed to the empty side of the bar, to ensure both strain gauges are affected similarly. By taping a ventilator to a stick of aluminium that is screwed into the top part of the drone to frame connector as in figure 4.5b. The ventilator is connected to a 7V supply and placed 9 cm away from the gauges, it is shown in figure 5.2.

5.2 Synchronising sensor data

Unfortunately, the HX711 and the Mini40 have a different sampling rate. The Mini40 samples at a rate of 500Hz while the HX711 has a standard sampling rate of 10Hz. In order to calibrate and to measure the force acting on both sensors at the exact same time, the samples must be aligned and clocked together. This is possible by implementing a Matlab script that reads out both values at exact same times and frequencies, thus aligning both results perfectly. Using Simulink and external Matlab packages specifically for Arduino, this would be a simple programming task. Unfortunately, the HX711 does not uses a common communication protocol and uses a custom I2C protocol that is not supported by Matlab. To get both measurements aligned, a calibration point will be set in the data samples, a point that is equal in both measurements that allows for a soft-coded shift in one of the two data sets. This implies in measurements related to calibration, a significant large force must be exerted first. This peak will cause a high value in both data sets of which the samples can be manually aligned. To do so, a few steps are followed.

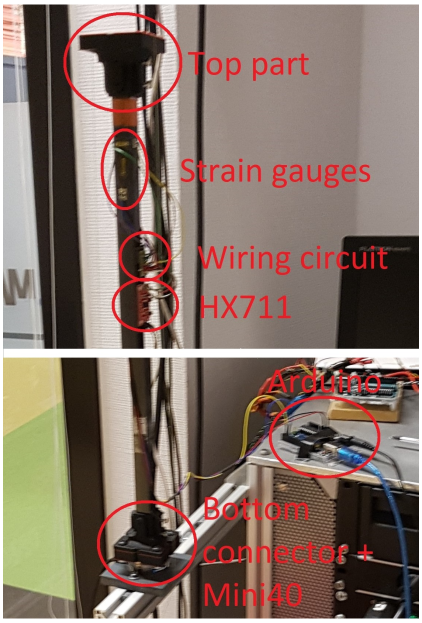


Figure 5.1: Test-setup

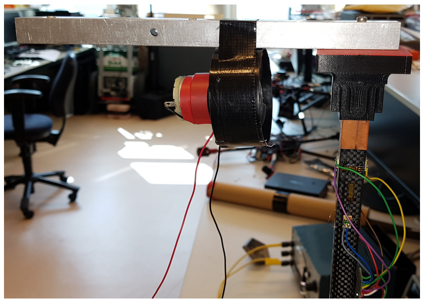


Figure 5.2: Ventilator connection to rod

- Import data files into Matlab
- Convert data to array's instead of tables using table2array()
- Divide the Mini40 data by 1000000 (1N = 1000000).
- Downsample the Mini40 data with downsample() or resample().
- Align both sets on the calibration point by removing samples from the data-set which has its calibration sample further in time than the other.
- Align both data-sets in length, removing samples from the longest remaining data set.

5.3 Arduino code

In order to perform experiments, an Arduino code is used that only reads the raw values from the ADC. This data is not processed any further and stored in a file on a computer. This allows for data manipulation in software and does not need additional measurements or processing power from the Arduino. The code is provided in G.1

The code firstly includes a library called HX711-master. This library can be found online and is written especially for communication with the HX711 [22]. After a class object has been declared, a call is made for the read function from the HX711-master library. This code itself has not been altered and thus a gain of 128 is used. The read function shifts the 24 bits coming out of the HX711 to a 32 bits long that can be easily read by a computer. This value is then forwarded with a Serial.print() statement to the serial output port of the Arduino.

6 Experimental results

6.1 Temperature effects

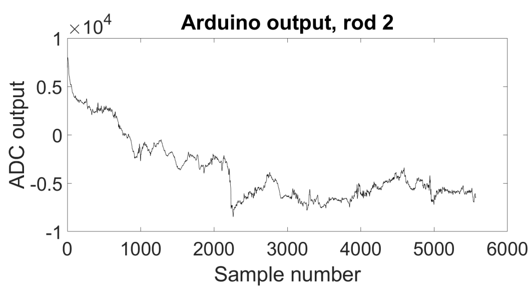
A first experiment is performed to analyse the effects of self-heating in the strain gauges. As mentioned in section 3.1.1, there should be no changes in the output as a result of temperature changes. It is also implied in section 2.2.3 that as soon as a current starts to flow, gauges will heat up due to power dissipation. As a first measurement, the ADC output from the Arduino is analysed for about 9 minutes. A Matlab plot of the result is shown in figure 6.1a. From this figure, it can be seen that the output of the ADC changes rapidly the first 2500 samples after which it goes down and sort of stabilises.

A second experiment is performed where a ventilator is connected to the rod. This experiment is performed to simulate the turbulent wind caused by the rotors and the effects of this on the strain gauges. The measurement starts without any additional wind for 1 minute. After this first minute the ventilator is turned on for 90 seconds and switched of again. It must be noted that the connector does add weight to the system and thus the mean value is not equal as in other experiments with rod 2. The result is shown in figure 6.1b.

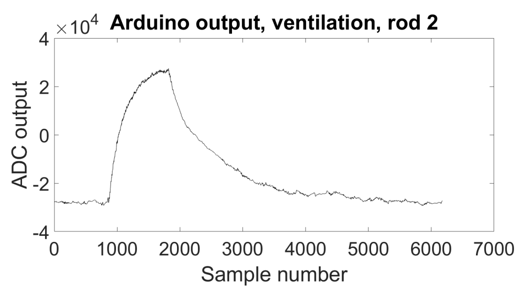
6.2 Weights and calibrating forces

In order to calibrate the ADC output, both measurements from the ATI sensor and the strain gauge sensor must be aligned. Each sample of the ATI sensor must correspond to the correct gauge sample and thus, as mentioned in 5, a peak in the measurement is added by applying a high pressure to the bar. After this peak, slowly, sand is added in the bucket. When the bucket has been fully filled it was removed from the set-up after a few minutes. Two data sets have been obtained, one from the Arduino and the other from the Mini40. The raw ADC and Mini40 data (no scaling, shifting or re-sampling) are shown in figure 6.2a and figure 6.2b.

As mentioned in 5.2, this data will be manipulated to be aligned with each other. Assumed is a frequency of 500Hz for the Mini40 (provided during measurements by the Mini40 itself) and a frequency of 10Hz for the HX711 (as provided in the data-sheet), resulting in a down-sampling ratio of 50. From the aligned results (where the down peak is synchronised for both datasets and where they have the same length) the Arduino output is plotted against the known forces of the Mini40, shown in figure 6.3a and 6.3b where in the second figure the first 500 samples (the down peak) of both data sets is removed. These figures thus depict the difference between the Mini40 data and the Arduino data.

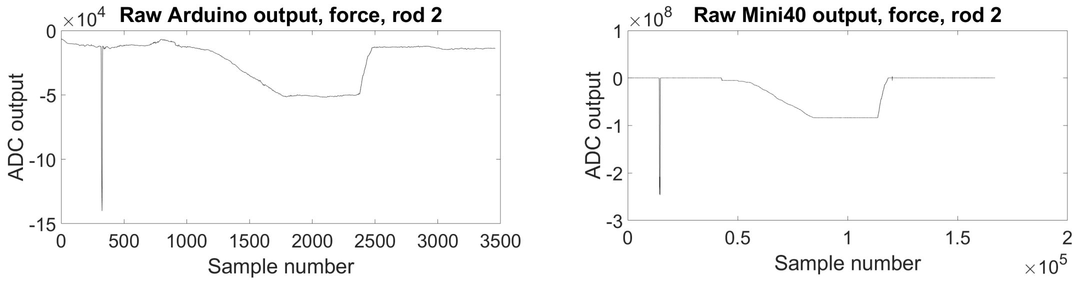


(a) Self-heating of strain gauges



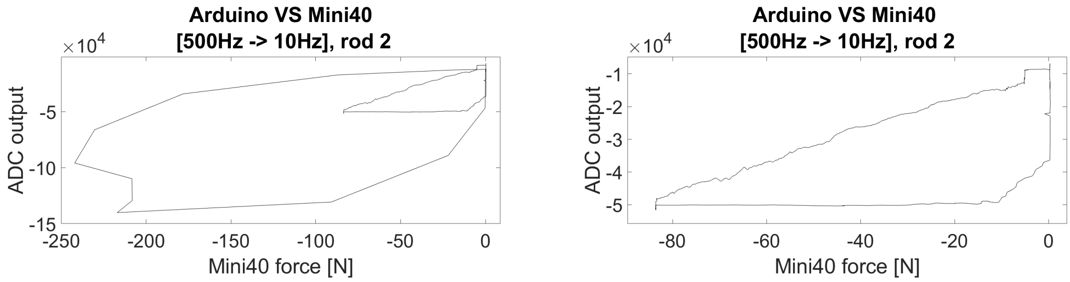
(b) Ventilator in front of strain gauges

Figure 6.1: Temperature effects



(a) Raw Arduino output **(b)** Raw Mini40 output

Figure 6.2: Raw Arduino and Mini40 data



(a) Down peak not removed **(b)** Down peak (500 samples) removed

Figure 6.3: Synchronised Arduino and Mini40 data

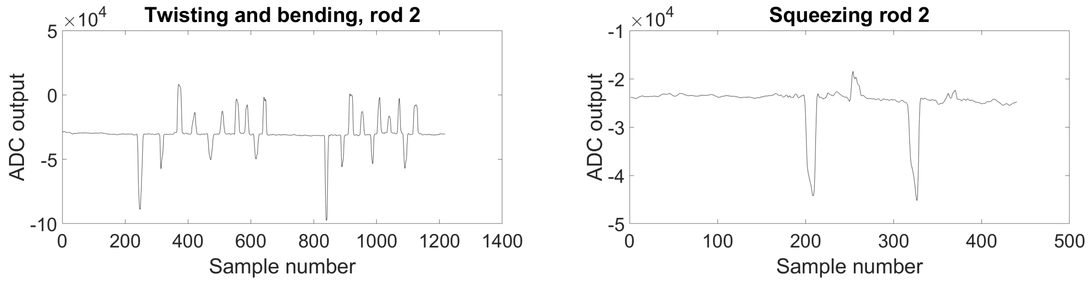
6.3 Bending and additional forces

In order to analyse the effects of bending, an experiment is performed where rod 2 is bend and twisted. The twist is performed by rotating the upper part of the rod. Bending is performed by blocking the middle of the bar by hand, while simultaneously pushing away the top part. Forces up and down are exerted by pulling and pushing the top part up or downwards. The orientation of the rod is similar as in figure 5.1
The forces exerted in the experiment are as follows: 1: Force down, 2: twist counter-clockwise, 3: twist clockwise,4: force up, 5: bending right, 6: bending left, 7: force up, 8: bending backwards, 9: bending forwards, 10: force up. This sequence is repeated twice and the result is shown in figure 6.4a.
Additionally, while screwing the connectors to the rod, it was noticed that tightening the connectors by screwing the bolts in changed the output of the ADC. This gave rise to question whether squeezing the material would influence the results. Therefore, exactly this is experimented by measuring the output while pressing together the two opposite sides of the square bar on the top part of the rod. Firstly the sides with strain gauges were squeezed together, after which the remaining sides followed. This has been repeated twice. The result is shown in figure 6.4b.

6.4 Intermediate interpretation of results

By looking at the results in the figures, observations that relate to faults in the setup can be made.

Firstly, looking at figure 6.2a it can be observed that the HX711 is wired "incorrectly". Firstly, this chip is originally meant for a weight scale and an increasing force should increase the output of the ADC [23]. Yet, it is exactly the opposite. Similarly, observing figures 6.1a and 6.1b, it can be seen that temperature effects are exactly opposite of what is expected. Increasing temperature (self-heating of strain gauges) should result in an

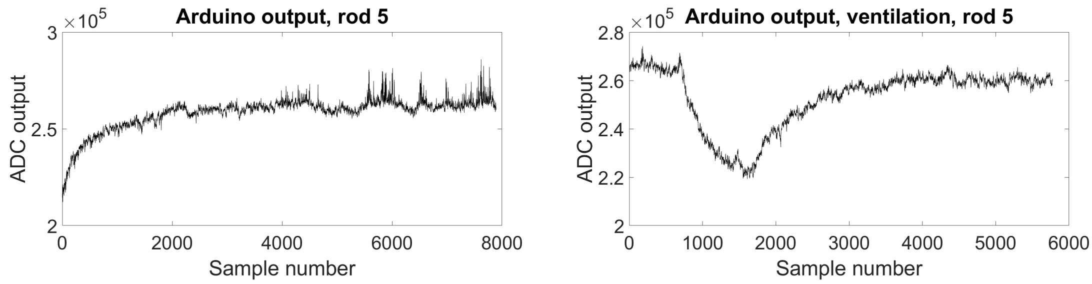


(a) Twisting and bending, rod 2 **(b)** Squeezing rod 2

Figure 6.4: Twisting, bending and squeezing rod 2

increase in resistance (and thus, normally, a higher ADC output) while ventilation should cool the gauges down and thus result in a lower resistance. Normally resulting in a lower ADC output. Indeed, referring to [23], it can be seen that the orientation of the output voltage of the wheatstone bridge used in this example is reversed compared to the setup used in this report. Switching the O+ and O- wire of the HX711 should thus resolve this issue.

Similar measurements have been performed, although not with the same rod and not in the flight lab but its adjacent lab to which it has an open connection. Rod 5 has been used and here the O+ and O- wire are re-soldered. Implying the O+ from the wheatstone bridge connector (white wire) to be connected to the GRN input of the HX711 and vice versa. The result is shown in figures 6.5a and 6.5b. Other experiments have not been redone. Firstly due to the fact that the output of the ADC will simply be reversed (positive difference → negative difference and vice versa) and due to time constrains.



(a) Self-heating of strain gauges rod 5 **(b)** Ventilator in front of strain gauges rod 5

Figure 6.5: Temperature effects

Secondly, looking at figures 6.3a and 6.3b, it can be observed that the peak value causes a lot of additional hysteresis in the plot. Removing this peak however does not fully remove these additional lines. Instead, during the time that weight is removed, the Arduino still outputs -5000 (horizontal line in figure 6.3b. It seems that the later samples are not aligned, even though the first peak is perfectly aligned (same sample number in the data sets). This gives rise to question the frequencies used for measuring. Assuming the frequency of the Mini40 to be correct (since it is provided by the Mini40 during measurements), the HX711 frequency is measured by making a small change in the Arduino code used. No output will be generated, but only after 5000 samples being read the total runtime of the Arduino code is provided. Performing this experiment, a runtime of 483323ms is obtained, implying a frequency of 10.34Hz. This calls for down sampling by 48,332, which is performed using the resample() function from Matlab. The other steps remain equal and the same data is used. After both data sets are aligned, the first 500 samples are removed to remove the down peak. The result is shown in figure 6.6.

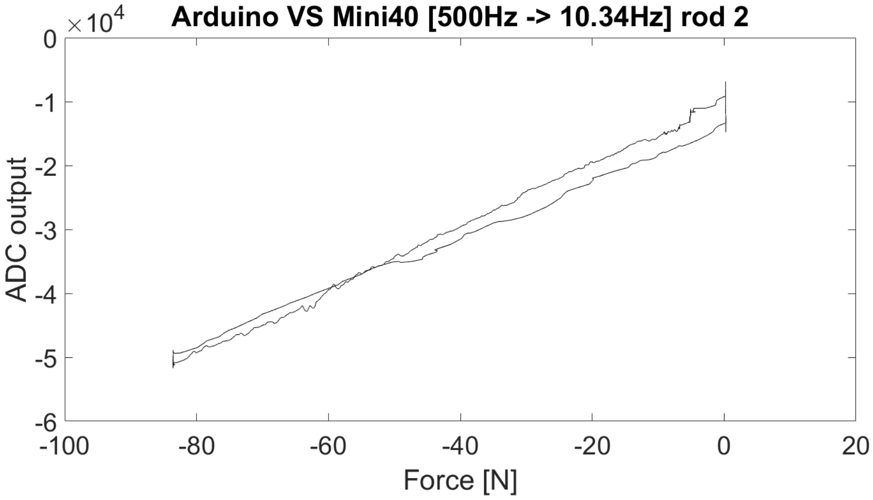


Figure 6.6: Mini40 VS Arduino data, down peak (500 samples) removed, resampled (500Hz $\rightarrow$ 10.34Hz)

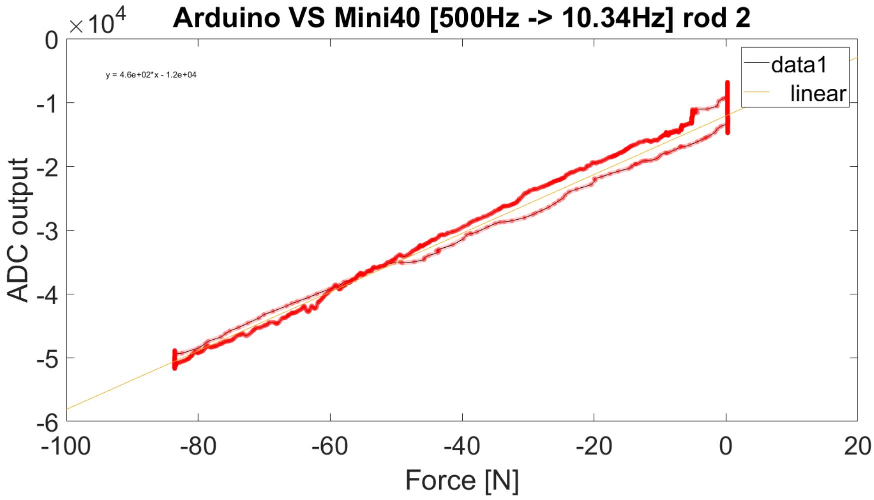
6.5 Analysis of results

6.5.1 Calibration + sensitivity

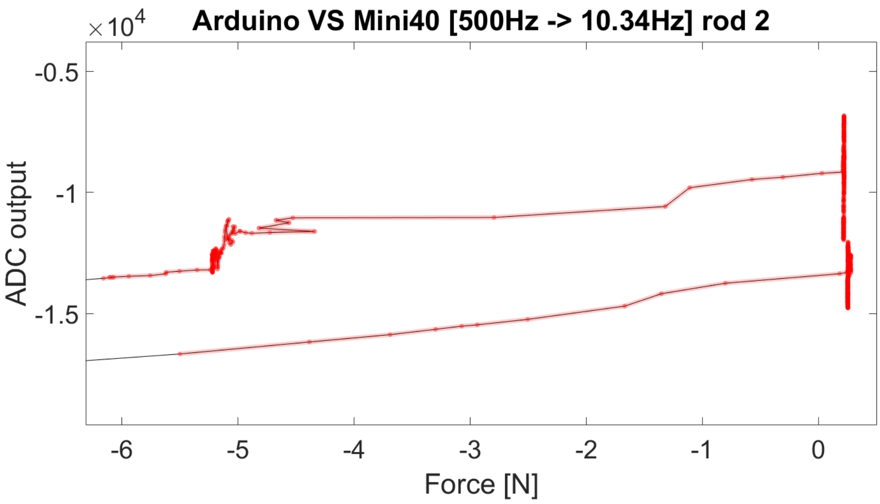
The plot as it is depicted in figure 6.6 gives insight regarding the characteristics of rod 2 and the sensor. This figure is plotted again in figure 6.7a where a linear fit is added to the plot, together with all data points marked red. In figure 6.7b the most right vertical line is magnified. In figure 6.7c, again the same as 6.7a is plotted, although the data samples after and during the weight being removed are discarded. The linear fit in figure 6.7a has a slope of 460 and an offset of -12000. The linear fit in figure 6.7c has a slope of 490 and and offset of -10000.

Without any forces applied, the sensor is relative inconsistent. For the same force of approximately 0.2N, obtained by the Mini40, the sensor has an output ranging between -7000 and -15000, implied by the right vertical line that is enlarged in figure 6.7b. Here also a difference can be seen between the zero force moments before and after the weight is applied. Before weight is added, the values range from -7000 till -12000 while afterwards the values range from -12000 till -15000 As soon as a weight is applied, the red most dense line is followed. Since weight is slowly increased, more samples are available compared to when the weight is removed. This results in a difference between sample density as can be seen in figure 6.7a. It can be seen that ADC values when the weight is increasing are different than ADC values when the weight is removed, thus implying hysteresis. The biggest difference can be seen at around -5N, also contained in figure 6.7b. Here the weight of bucket is added to the setup and the the difference between the Mini40 the ADC is increased, as the strain sensors apparently do not measure the force induced by this empty bucket. However, the constant weight does cause an output range of strain gauge values from the ADC, ranging between -11000 till -13000. As soon as constantly weight is being added, an approximate linear line is followed until the moment the force is constant again. This constant force again causes an uncertainty in the ADC values, ranging from -49000 till -52000.

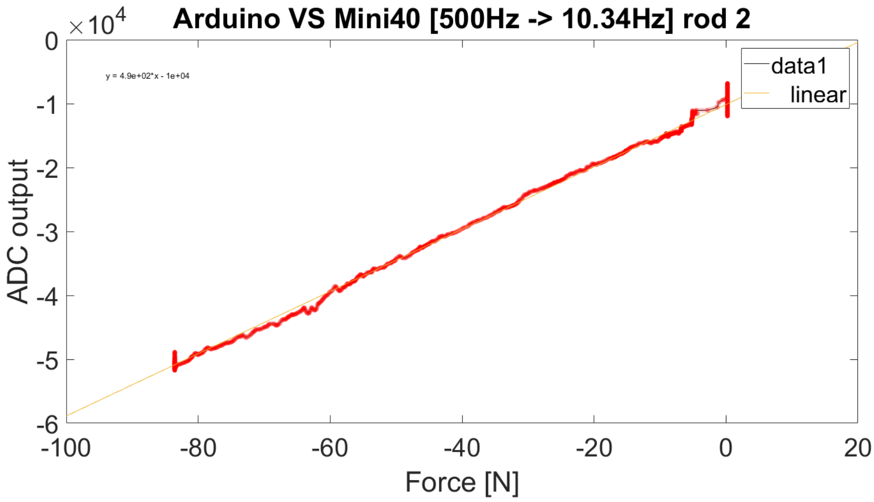
On average, the output values thus have a maximum range of $\frac{8000+2000+3000}{3} \approx 4333$, including hysteresis. Here only the constant force measurements are used. Dividing this



(a) Data points marked + linear fit (y = 460x - 12000



(b) Zoomed in to figure 6.7a



(c) Data points marked + linear fit (y = 490x - 10000), only weight being added

Figure 6.7: Mini40 VS Arduino data, down peak (500 samples) removed, re-sampled (500Hz → 10.34Hz)

with the sensitivity of the sensor as calculated by Matlab (460 ADC units per N), this results in uncertainty of 9.4N that can not be accurately measured in the used setup. If there would have been no hysteresis, the setup would be more sensitive. By removing samples when the weight is (being) removed, figure, 6.7c is obtained that depicts an almost similar line as in figure 6.7a, yet, without the less dense line and the lower data-points in figure 6.7b are gone. An average maximum range of $\frac{5000+2000+3000}{3} \approx 3333$ is obtained. A new linear fit shows a sensitivity of 490 and thus the uncertainty of measurements is brought down to 6.8N.

A potential explanation for the hysteresis seen comes as a result of the two different sensors. Although both sets of data are aligned, they are not synchronised/clocked together. Samples are provided at the same time interval and frequencies, they still might not be measured at the same time. Assuming correct frequencies, removing the first 500 values reduces the hysteresis caused by the calibration pulse, but the hysteresis is not fully removed, as is can be seen in figure 6.6. However, looking more closely at figure 6.2a, it can be seen that the Arduino output changes around sample number 3000, although no additional forces have been applied. This would suggest that not (only) the sampling of both sensors is causing hysteresis, but the carbon rod itself.

Also the calibration offset can be obtained, as this is equal to the offset of the linear fit, being -12000 or -10000 for the setup with and without hysteresis. Yet, as can be seen comparing rod 2 to rod 5 (figure 6.1a and 6.5a) the offset is severely different for each rod. This can be explained by the fact that no strain gauges have the exact same resistance and thus not all rods result in the same bridge output voltage.

6.5.2 Temperature effects + additional forces

From figures 6.1a, 6.5a, 6.1b, 6.5b, 6.4a and figure 6.4b, it is clear to see that the system is not immune for additional forces like bending and squeezing. Also the output changes as a result of temperature changes.

A potential explanation for the bending and twisting effects as shown in figures 6.4a and 6.4b, might come from strain gauges that are not placed at the exact correct angles relative to each other. It is possible that not well placed gauges are contracting less than the expanding gauges are expending, and thus implying a difference in the output of the bridge, resulting in an amplified output signal from the ADC. Since the way the strain gauges are applied (by hand and tweezers), and due to the size of the gauges, a small misalignment can easily be made without any consent.

Looking at figure 6.1b and, 6.5b, a ventilator close to the gauges changes the output of the ADC even more than does the entire bucket of sand. Similarly, self-heating of the gauges, as shown in figures 6.1a and 6.5a, does have influence on the ADC output, even though the setup had been designed not to be influenced by temperature nor bending effects. As the strain gauges are glued, it might be possible that the glue beneath the strain gauges is not equally divided. When the gauge heats up and wants to expand, some gauges might expand more easily than others, due to the non-homogeneous division of glue between the carbon rod (with an almost zero linear expansion coefficient as mentioned in 3.1.2) and the gauge. This difference might cause unwanted temperature effects. Also, as no strain gauges will have the exact same resistance, the bridge will not be exactly balanced, as discussed in 3.1.1.

7 Conclusion and recommendations

7.1 Conclusion

The goal of this thesis has been to realise a safety cage with integrated force sensing, in order to enhance more safe interactive capabilities of aerial vehicles. To do so, two main objectives had to be met; force sensors had to be designed and had to be integrated into a 3D design of a safety cage.

Firstly, force sensors have been created with the use of strain gauges and these sensors have a minimal accuracy of 9.4N. Secondly, these sensors are mounted on carbon rods that serve the purpose of a frame for a provided safety cage. They are thus inherently implemented in the cage design and it can be concluded that both objectives have been achieved.

As a rectangular design is chosen for placing the sensors, force is being measured in 3 different directions, increasing redundancy by measuring each axis twice. Unfortunately, the sensors seem not to be of high accuracy and thus accurate force sensing is not accomplished. A physical model has been realised and as the parts for the cage are made out of carbon and ABS, a potential lightweight yet flexible and robust cage is realised. As a lot of material is used, the weight of the system is quite large. It can thus not be concluded that the system is lightweight.

7.2 Recommendations

Mentioned in 6.5.1, hysteresis occurs in the output of the ADC together with noise, temperature drifts, and disturbances caused by bending, squeezing and/or twisting. All of these reduce the potential accuracy of the sensor in a negative way as a noisier signal is obtained.

A first improvement would be to figure out where the noise is coming from. By hooking the output of the ADC and the wheatstone bridge to an oscilloscope, an answer could be obtained whether it are the gauges or the HX711 where noise originates from. To cope with the temperature issues, temperature controlled calibration should be implemented by using additional temperature sensors. Additionally, strain gauges with temperature compensation for carbon could be analysed. Bending, additional forces and temperature effects might be dealt with by increasing the accuracy of placing the sensors (using XY gauges for example). Yet however, these effects should be further investigated to find their origin. Hysteresis in the calibration might be reduced by a software controlled input, for both the Arduino/ADC and the Mini40. A code ensuring both sensors to be clocked to each other and thus providing data at the exact same times could potentially reduce the hysteresis. Also, hysteresis in carbon rods themselves might cause hysteresis in the final system. This should be investigated in more detail, as it needs to be known whether or not this can be excluded. Additionally, a complete calibration could be performed, where a first set of samples is ignored (startup phase), an average is calculated together with the variance where after a potential more accurate measurement can be obtained. Al these sensors could be combined by a micro-controller in order to fully create an interactive system.

From the physical cage model and the weight of components, it is recommended to decrease the weight of the cage to cage connectors, the drone+frame to cage connectors and the square rods. The connectors could be significantly decreased in weight by making them smaller. The rods could be decreased by connecting them to the same rods as the propellers. This would also allow to make the entire cage elliptical, making the drone to frame connectors redundant.

A Strain gauges

A.1 Derivation of gauge factor

The gauge factor is the ratio between the relative change of resistance and the relative change of length of the wire.

It is known that the resistance of a wire can be calculated using equation 2.1. In order to obtain the gauge factor, the derivative of R with respect to L must be found. However, in this equation this proves cumbersome. When L changes, A changes proportionally due to the Poisson's ratio. Therefore, instead, the equation is multiplied with L/L to obtain a volume. Since mercury is used, this volume is now considered a constant (a very low poisons ratio) and the derivative can be obtained.

$$R = \frac{\rho L}{A} = \frac{\rho L}{A} = \frac{\rho L^2}{V} \tag{A.1}$$

$$\frac{\delta R}{\delta L} = \frac{2\rho L}{V} = \frac{2\rho L}{A} = \frac{2\rho L^2}{V} \tag{A.2}$$

Multiplying this result again with L/L and by looking at the result of equation A.1.

$$\frac{\delta R}{\delta L} = \frac{2\rho L}{V} * \frac{L}{L} = \frac{2\rho L^2}{VL} = \frac{2R}{L} \tag{A.3}$$

In order to obtain the gauge factor, as defined in 2.2, a simple division by R and L results in a final gauge factor of 2 [24]. However, this gauge factor does not only depend on L and A and the resistivity ρ is not a constant. Instead, it changes with specific forces applied and the resistivity is temperature dependent. This piëzo-resistive effect can be seen in a different equation for the gauge factor as shown in A.4. For semiconducting strain gauges, it is the second part of equation A.4 that dominates [25].

$$K = \frac{\frac{\Delta R}{R}}{\frac{\Delta L}{L}} = (1 + 2\nu)\epsilon + \frac{\Delta\rho}{\rho} \tag{A.4}$$

A.2 Calculating strains

Using the elasticity modulus of a material, also known as Young's Modulus or Young's constant, strains can be calculated.

Stress is equal to the force per area ($\sigma = F/A$), but it is also known that the stress is equal to Young's constant multiplied with the strain, $\sigma = E * \epsilon$. Both equations can be combined to obtain equation A.5

$$\epsilon = \frac{F}{A * E} \tag{A.5}$$

It can be calculated that a force of 1N on a copper wire with a Young's modulus of 110GPa and a diamater of 1mm, results in a strain occurs 0.001. This means, for each mm of wire, it will stretch or compress 0.001mm. Obviously, such a small change will only cause a tiny change in resistance. This change in resistance, however, depends on the gauge factor.

In order to obtain the change in resistance as a result from a force applied, the relation as in equation 2.2 can be rewritten into equation A.6.

$$K = \frac{\frac{\Delta R}{R}}{\epsilon} \rightarrow \frac{\Delta R}{R} = K\epsilon \tag{A.6}$$

Thus, the 1N force applied to the small wire would cause a strain of 0.001mm/mm. Using a gauge factor of two and only this specific wire, this would result in a ratio of between R and ΔR of 0.002. This implies that the force of 1N applied would only cause a change in resistance of 0.2%.

Thus, strains are commonly minute and thus only a very tiny change in resistance will be obtained. In order to measure such a very small change in resistance, a very accurate measuring device must be used. For this purpose, commonly a wheatstone bridge is used.

B Wheatstone bridges

B.1 Single gauge

$$V_{out} = \frac{R2R3 - R4R1}{(R2 + R1)(R3 + R4)} * V_{in}$$
$$V_{out} = \frac{R_0^2 - R_0(R_0 \pm \Delta R)}{(R_0 + (R_0 + \Delta R))(R_0 + R_0)} * V_{in}$$
$$V_{out} = \frac{R_0^2 - R_0^2 \pm \Delta R R_0}{R_0^2 + R_0^2 \pm 2\Delta R R_0 + 2R_0^2} * V_{in}$$
$$V_{out} = \frac{\pm \Delta R R_0}{4R_0^2 \pm 2\Delta R R_0} * V_{in}$$
$$V_{out} = \frac{\pm \frac{\Delta R}{R_0}}{4 + \frac{\pm 2\Delta R}{R_0}} * V_{in}$$

Assuming Δ is small relative to the initial value of R (as is shown earlier), we can ignore the second part in de denominator and the expression as in B.1 can be obtained [26].

$$V_{out} = \pm \frac{1}{4} V_{in} \frac{\Delta R}{R_0} \tag{B.1}$$

Thus, the output voltage is equal to the change in resistance times a quart of the input voltage.

B.2 Two gauges

Case 1

$$V_{out} = \frac{R2R3 - R4R1}{(R2 + R1)(R3 + R4)} * V_{in}$$
$$V_{out} = \frac{R_0^2 - (R_0 + \Delta R)^2}{((R_0 + \Delta R) + R_0)^2} * V_{in}$$
$$V_{out} = \frac{R_0^2 - R_0^2 - 2\Delta R R_0 - (\Delta R^2)}{R_0^2 + \Delta R R_0 + R_0^2 + \Delta R R_0 + (\Delta R)^2 + \Delta R R_0 + R_0^2 + (\Delta R)^2 + R_0^2} * V_{in}$$
$$V_{out} = -\frac{2\Delta R R_0 + (\Delta R)^2}{4R_0^2 + 4\Delta R R_0 + (\Delta R)^2} * V_{in}$$
$$V_{out} = \frac{-\frac{2\Delta R}{R_0} + \frac{(\Delta R)^2}{R_0}}{4 + \frac{4\Delta R}{R_0} + \frac{(\Delta R)^2}{R_0}} * V_{in}$$

Again assuming ΔR is small relative to the initial value of R, we can ignore these terms, including its square, to obtain a final expression as in B.2

$$V_{out} = -\frac{1}{2} V_{in} \frac{\Delta R}{R_0} \tag{B.2}$$

Case 2

$$
\begin{gathered}
V_{out} = \frac{R2R3 - R4R1}{(R2 + R1)(R3 + R4)} * V_{in} \\
V_{out} = \frac{(R_0 + \Delta R)R_0 - (R_0 - \Delta R)R_0}{[(R_0 - \Delta R) + (R_0 + \Delta R)](R_0 + R_0)} * Vin \\
V_{out} = \frac{R_0^2 + \Delta R R_0 - R_0^2 + \Delta R R_0}{(R_0 - \Delta R + R_0 + \Delta R)(2R_0)} * V_{in} \\
V_{out} = \frac{2\Delta R R_0}{2R_0^2 - 2\Delta R R_0 + 2R_0^2 + 2\Delta R R_0} * V_{in} \\
V_{out} = \frac{2\Delta R R_0}{4R_0^2} * V_{in}
\end{gathered}
$$

By dividing both the numerator and the denumerator with R_0, obtained is expression B.3

$$V_{out} = \frac{1}{2} V_{in} \frac{\Delta R}{R_0} \tag{B.3}$$

B.3 Four gauges

$$
\begin{gathered}
V_{out} = \frac{R2R3 - R4R1}{(R2 + R1)(R3 + R4)} * V_{in} \\
V_{out} = \frac{(R_0 + \Delta R)^2 - (R_0 - \Delta R)^2}{[(R_0 + \Delta R) + (R_0 - \Delta R)][(R_0 + \Delta R) + (R_0 - \Delta R)]} * V_{in} \\
V_{out} = \frac{R_0^2 + 2\Delta R R_0 + (\Delta R)^2 - R_0^2 + 2\Delta R R_0 - (\Delta R)^2}{(2R_0)(2R_0)} * V_{in} \\
V_{out} = \frac{4\Delta R R_0}{4R_0^2} * V_{in} \\
V_{out} = \frac{\Delta R R_0}{R_0^2} * V_{in}
\end{gathered}
$$

$$V_{out} = \frac{\Delta R}{R_0} V_{in} \tag{B.4}$$

C Wrench transformations

Most equations and information in this section are taken from [27]

From physics it is known that power can be calculated as a force times a velocity.

$$P = vF \tag{C.1}$$

As shown in [27] this implies the wrench to be multiplied with the twist. However, only if all are relative to the same frame.

$$P^A = W_r^A T_c^A$$

$$P = \begin{pmatrix} T_x & T_y & T_z & F_x & F_y & F_z \end{pmatrix} \bullet \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_y \\ v_x \\ v_y \\ v_z \end{pmatrix}$$

In order to transform the twist of a body A relative to frame B and expressed in frame A to a twist of body A relative to frame B but expressed in frame B, ${}^A T_A^B \to {}^B T_A^B$, the first twist must be multiplied with the adjoint matrix of body A relative to frame B.

$${}^B T_A^B = Ad_{H_A^B} {}^A T_A^B; \qquad Ad_{H_A^B} \equiv \begin{pmatrix} R_A^B & 0_{3x3} \\ {}^B\tilde{o}_A R_A^B & R_A^B \end{pmatrix} \tag{C.2}$$

The adjoint matrix contains elements from the homogeneous transformation matrix H_A^B, as explained more in depth in [27]. Firstly it contains the 3d rotation matrix R_A^B which depicts a transformation from frame A to frame B and 0_{3x3}, a 3x3 matrix filled with zero's. The term ${}^B\tilde{o}_A$ is the distance vector of frame A relative to frame B, put in tilde form, as shown in C.3. The full derivation of the adjoint matrix (similar as the H matrix) are shown in [27]

$${}^B o_A = \begin{pmatrix} o_x \\ o_y \\ o_z \end{pmatrix} \Leftrightarrow {}^B\tilde{o}_A = \begin{pmatrix} 0 & -o_z & o_y \\ o_z & 0 & -o_x \\ -o_y & o_x & 0 \end{pmatrix} \tag{C.3}$$

In order to transform the wrench from one frame to another, the power continuity property is used, since a wrench in one frame should exert the same power in another frame.

$$\begin{aligned} {}^A W {}^A T = {}^A P = {}^B P = {}^B W {}^B T \\ {}^A W (Ad_{H_B^A} {}^B T) = {}^B W {}^B T \\ ({}^A W Ad_{H_B^A}) {}^B T = {}^B W {}^B T \end{aligned}$$

Both sides now can be divided by ${}^B T$ resulting in C.4

$${}^A W Ad_{H_B^A} = {}^B W \tag{C.4}$$

By rewriting expression C.4 by means of taking the transpose, a similar expression can be obtained for a wrench as being a column vector.

$$\begin{aligned} {}^A W_r Ad_{H_B^A} = {}^B W_r \\ ({}^A W_r Ad_{H_B^A})^T = ({}^B W_r)^T \\ Ad_{H_B^A}^T {}^A W_c = {}^B W_c; \end{aligned}$$

Now A and B can simply be renamed, A $\rightarrow$ B and B $\rightarrow$ A, resulting in equation C.5

$$Ad^T_{H^B_A} {}^B W_c = {}^A W_c; \tag{C.5}$$

In order to obtain the wrench in frame A, the wrench in frame B must be multiplied with the adjoint matrix transposed. When multiple wrenches must be transformed in order to obtain a single wrench in the new frame, due to linearity, they can simply be added.

$$^A W_c = \sum_{i=1}^{n} Ad^T_{H^i_A} {}^i W_c \tag{2.13}$$

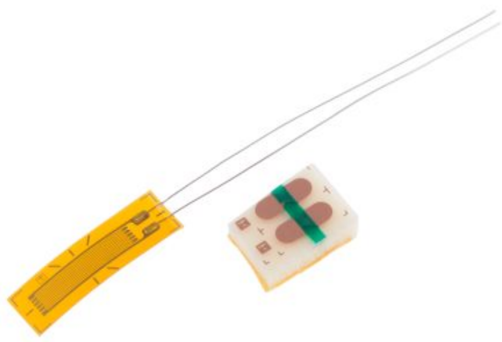
D Strain gauge datasheet

ENGLISH

Datasheet

RS Pro Wire Lead Strain Gauge 4mm, 120Ω -30°C +180°C

RS Stock No: **865-6235**



Product Details

RS Pro wire lead strain gauge has a gauge resistance of 120 Ω and 4 mm wide with the operating temperature range from -30 to +180°C. It has foil type polyester backed gauge with integral 30 mm flying lead. It is supplied with 2 self adhesive terminal pads.

Features and Benefits

- Fatigue life >10^6 reversals at 1000 µ strain
- Temperature compensation for steel with red backing
- Temperature compensation for aluminium with blue backing
- Gauge resistance 120 Ω
- 13 mm long

ENGLISH

Specifications:

Dimensions	13 x 4 mm
Gauge Factor	2
Gauge Length	8 mm
Gauge Resistance	120 Ω
Length	13 mm
Maximum Operating Temperature	+180°C
Minimum Operating Temperature	-30°C
Terminal Type	Wire Lead
Width	4 mm

E HX711 data-sheet

海芯科技 AVIA Semiconductor

HX711

24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales

DESCRIPTION

Based on Avia Semiconductor's patented technology, HX711 is a precision 24-bit analog-to-digital converter (ADC) designed for weigh scales and industrial control applications to interface directly with a bridge sensor.

The input multiplexer selects either Channel A or B differential input to the low-noise programmable gain amplifier (PGA). Channel A can be programmed with a gain of 128 or 64, corresponding to a full-scale differential input voltage of ±20mV or ±40mV respectively, when a 5V supply is connected to AVDD analog power supply pin. Channel B has a fixed gain of 32. On-chip power supply regulator eliminates the need for an external supply regulator to provide analog power for the ADC and the sensor. Clock input is flexible. It can be from an external clock source, a crystal, or the on-chip oscillator that does not require any external component. On-chip power-on-reset circuitry simplifies digital interface initialization.

There is no programming needed for the internal registers. All controls to the HX711 are through the pins.

FEATURES

- **Two selectable differential input channels**
- **On-chip active low noise PGA with selectable gain of 32, 64 and 128**
- **On-chip power supply regulator for load-cell and ADC analog power supply**
- **On-chip oscillator requiring no external component with optional external crystal**
- **On-chip power-on-reset**
- **Simple digital control and serial interface: pin-driven controls, no programming needed**
- **Selectable 10SPS or 80SPS output data rate**
- **Simultaneous 50 and 60Hz supply rejection**
- **Current consumption including on-chip analog power supply regulator: normal operation < 1.5mA, power down < 1uA**
- **Operation supply voltage range: 2.6 ~ 5.5V**
- **Operation temperature range: -40 ~ +85℃**
- **16 pin SOP-16 package**

APPLICATIONS

- **Weigh Scales**
- **Industrial Process Control**

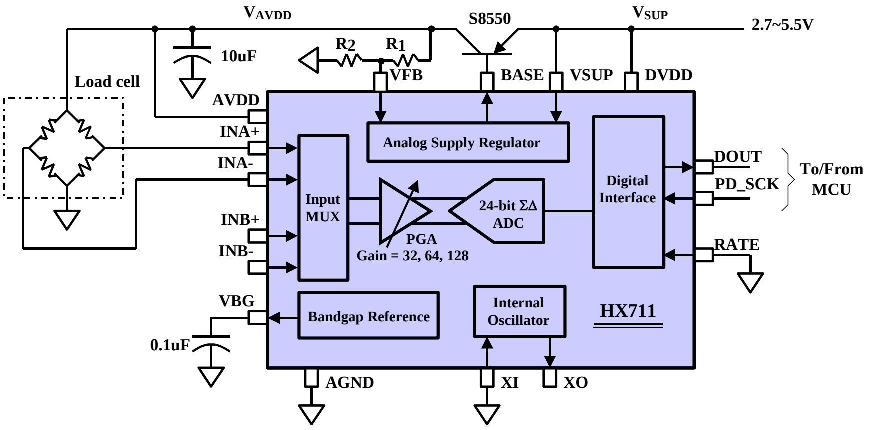
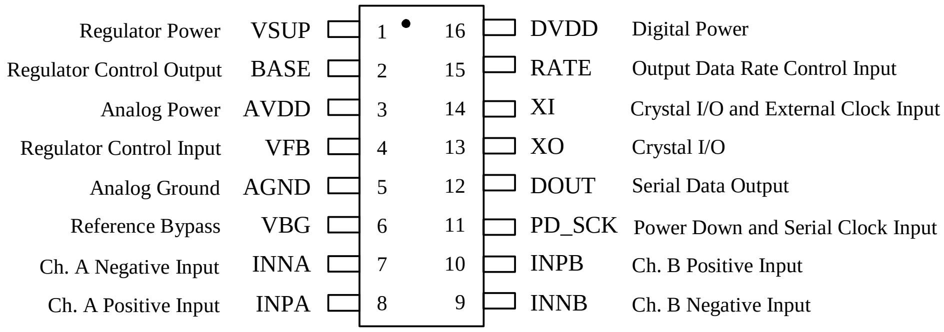


Fig. 1 Typical weigh scale application block diagram

TEL: (592) 252-9530 (P. R. China)
EMAIL: market@aviaic.com

AVIA SEMICONDUCTOR

Pin Description



SOP-16L Package

Pin #	Name	Function	Description
1	VSUP	Power	Regulator supply: 2.7 ~ 5.5V
2	BASE	Analog Output	Regulator control output（NC when not used）
3	AVDD	Power	Analog supply: 2.6 ~ 5.5V
4	VFB	Analog Input	Regulator control input（connect to AGND when not used）
5	AGND	Ground	Analog Ground
6	VBG	Analog Output	Reference bypass output
7	INA-	Analog Input	Channel A negative input
8	INA+	Analog Input	Channel A positive input
9	INB-	Analog Input	Channel B negative input
10	INB+	Analog Input	Channel B positive input
11	PD_SCK	Digital Input	Power down control (high active) and serial clock input
12	DOUT	Digital Output	Serial data output
13	XO	Digital I/O	Crystal I/O (NC when not used）
14	XI	Digital Input	Crystal I/O or external clock input, 0: use on-chip oscillator
15	RATE	Digital Input	Output data rate control, 0: 10Hz; 1: 80Hz
16	DVDD	Power	Digital supply: 2.6 ~ 5.5V

Table 1 Pin Description

KEY ELECTRICAL CHARACTERISTICS

Parameter	Notes	MIN	TYP	MAX	UNIT
Full scale differential input range	V(inp)-V(inn)		±0.5(AVDD/GAIN)		V
Common mode input		AGND+1.2		AVDD-1.3	V
Output data rate	Internal Oscillator, RATE = 0		10		Hz
	Internal Oscillator, RATE = DVDD		80		
	Crystal or external clock, RATE = 0		f_{clk}/1,105,920		
	Crystal or external clock, RATE = DVDD		f_{clk}/138,240		
Output data coding	2's complement	800000		7FFFFF	HEX
Output settling time (1)	RATE = 0		400		ms
	RATE = DVDD		50		
Input offset drift	Gain = 128		0.2		mV
	Gain = 64		0.4		
Input noise	Gain = 128， RATE = 0		50		nV(rms)
	Gain = 128， RATE = DVDD		90		
Temperature drift	Input offset（Gain = 128）		±6		nV/℃
	Gain（Gain = 128）		±5		ppm/℃
Input common mode rejection	Gain = 128， RATE = 0		100		dB
Power supply rejection	Gain = 128， RATE = 0		100		dB
Reference bypass（V_{BG}）			1.25		V
Crystal or external clock frequency		1	11.0592	20	MHz
Power supply voltage	DVDD	2.6		5.5	V
	AVDD，VSUP	2.6		5.5	
Analog supply current (including regulator)	Normal		1400		µA
	Power down		0.3		
Digital supply current	Normal		100		µA
	Power down		0.2		

（1）Settling time refers to the time from power up, reset, input channel change and gain change to valid stable output data.

Table 2 Key Electrical Characteristics

Analog Inputs

Channel A differential input is designed to interface directly with a bridge sensor's differential output. It can be programmed with a gain of 128 or 64. The large gains are needed to accommodate the small output signal from the sensor. When 5V supply is used at the AVDD pin, these gains correspond to a full-scale differential input voltage of ±20mV or ±40mV respectively.

Channel B differential input has a fixed gain of 32. The full-scale input voltage range is ±80mV, when 5V supply is used at the AVDD pin.

Power Supply Options

Digital power supply (DVDD) should be the same power supply as the MCU power supply.

When using internal analog supply regulator, the dropout voltage of the regulator depends on the external transistor used. The output voltage is equal to $V_{AVDD}=V_{BG}*(R1+R2)/R1$ (Fig. 1). This voltage should be designed with a minimum of 100mV below VSUP voltage.

If the on-chip analog supply regulator is not used, the VSUP pin should be connected to either AVDD or DVDD, depending on which voltage is higher. Pin VFB should be connected to Ground and pin BASE becomes NC. The external 0.1uF bypass capacitor shown on Fig. 1 at the VBG output pin is then not needed.

Clock Source Options

By connecting pin XI to Ground, the on-chip oscillator is activated. The nominal output data rate when using the internal oscillator is 10 (RATE=0) or 80SPS (RATE=1).

If accurate output data rate is needed, crystal or external reference clock can be used. A crystal can be directly connected across XI and XO pins. An external clock can be connected to XI pin, through a 20pF ac coupled capacitor. This external clock is not required to be a square wave. It can come directly from the crystal output pin of the MCU chip, with amplitude as low as 150 mV.

When using a crystal or an external clock, the internal oscillator is automatically powered down.

Output Data Rate and Format

When using the on-chip oscillator, output data rate is typically 10 (RATE=0) or 80SPS (RATE=1).

When using external clock or crystal, output data rate is directly proportional to the clock or crystal frequency. Using 11.0592MHz clock or crystal results in an accurate 10 (RTE=0) or 80SPS (RATE=1) output data rate.

The output 24 bits of data is in 2's complement format. When input differential signal goes out of the 24 bit range, the output data will be saturated at 800000h (MIN) or 7FFFFFh (MAX), until the input signal comes back to the input range.

Serial Interface

Pin PD_SCK and DOUT are used for data retrieval, input selection, gain selection and power down controls.

When output data is not ready for retrieval, digital output pin DOUT is high. Serial clock input PD_SCK should be low. When DOUT goes to low, it indicates data is ready for retrieval. By applying 25~27 positive clock pulses at the PD_SCK pin, data is shifted out from the DOUT output pin. Each PD_SCK pulse shifts out one bit, starting with the MSB bit first, until all 24 bits are shifted out. The 25^{th} pulse at PD_SCK input will pull DOUT pin back to high (Fig.2).

Input and gain selection is controlled by the number of the input PD_SCK pulses (Table 3). PD_SCK clock pulses should not be less than 25 or more than 27 within one conversion period, to avoid causing serial communication error.

PD_SCK Pulses	Input channel	Gain
25	A	128
26	B	32
27	A	64

Table 3 Input Channel and Gain Selection

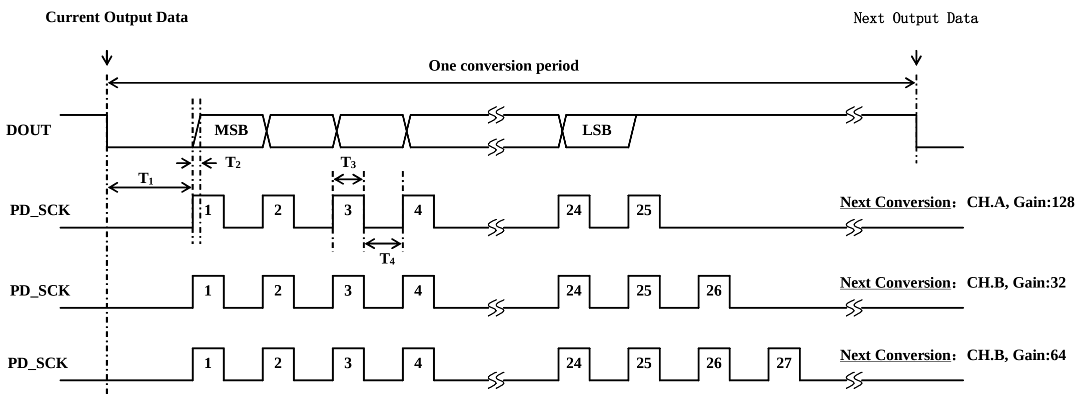


Fig.2 Data output, input and gain selection timing and control

Symbol	Note	MIN	TYP	MAX	Unit
T_1	DOUT falling edge to PD_SCK rising edge	0.1			μs
T_2	PD_SCK rising edge to DOUT data ready			0.1	μs
T_3	PD_SCK high time	0.2	1	50	μs
T_4	PD_SCK low time	0.2	1		μs

Reset and Power-Down

When chip is powered up, on-chip power on rest circuitry will reset the chip.

Pin PD_SCK input is used to power down the HX711. When PD_SCK Input is low, chip is in normal working mode.

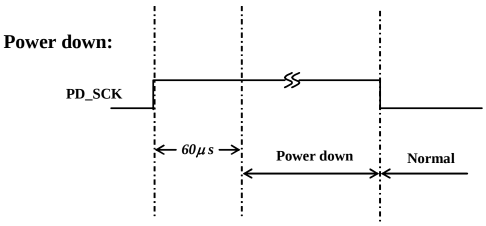


Fig.3 Power down control

When PD_SCK pin changes from low to high and stays at high for longer than 60μs, HX711 enters power down mode (Fig.3). When internal regulator is used for HX711 and the external transducer, both HX711 and the transducer will be powered down. When PD_SCK returns to low, chip will reset and enter normal operation mode.

After a reset or power-down event, input selection is default to Channel A with a gain of 128.

Application Example

Fig.1 is a typical weigh scale application using HX711. It uses on-chip oscillator (XI=0), 10Hz output data rate (RATE=0). A Single power supply （2.7～5.5V） comes directly from MCU power supply. Channel B can be used for battery level detection. The related circuitry is not shown on Fig. 1.

Reference PCB Board (Single Layer)

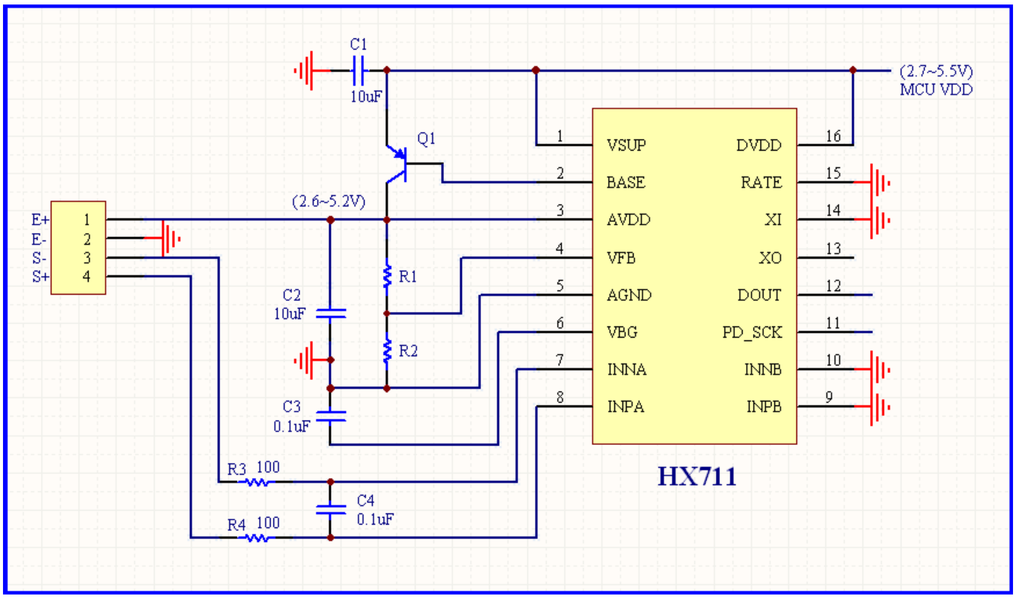


Fig.4 Reference PCB board schematic

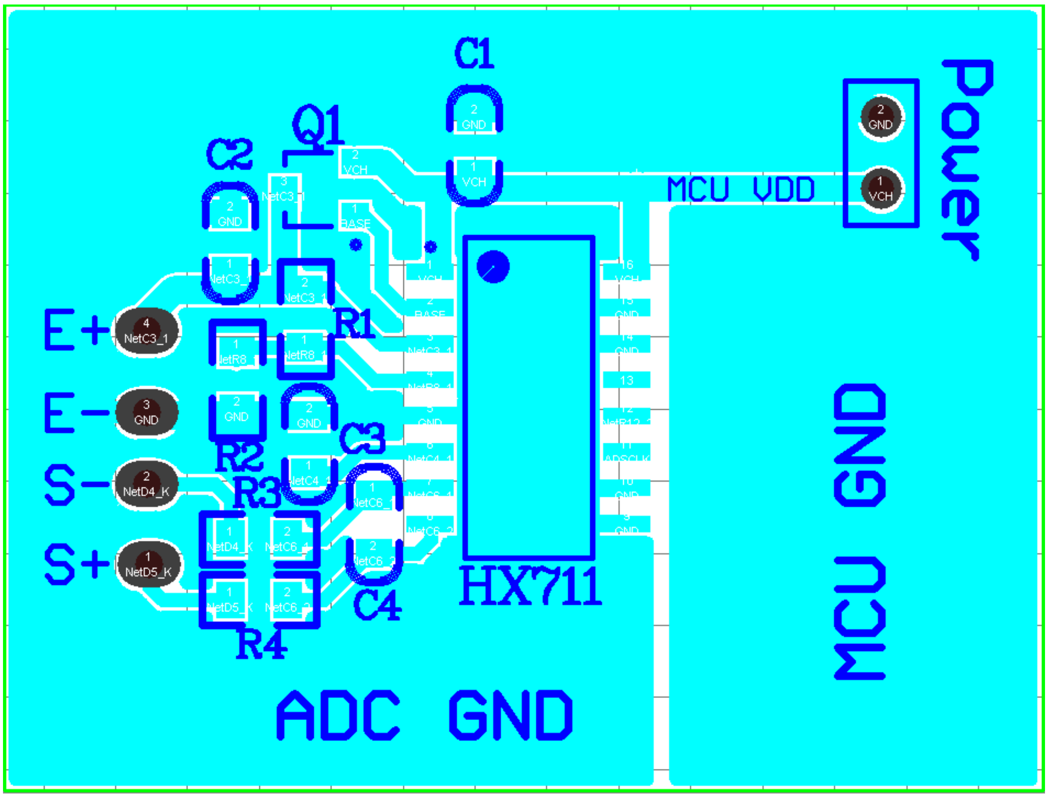


Fig.5 Reference PCB board layout

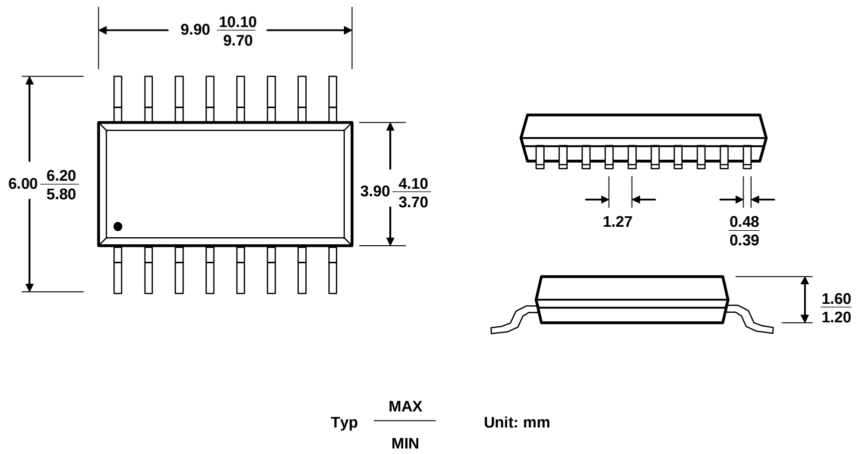
Reference Driver (Assembly)

```
/*--------------------------------------------------------------------
Call from ASM:      LCALL   ReaAD
Call from C:    extern unsigned long ReadAD(void);
                  .
                  .
                 unsigned long data;
                 data=ReadAD();
                  .
                  .
---------------------------------------------------------------------*/
PUBLIC          ReadAD
HX711ROM        segment  code
rseg            HX711ROM

sbit             ADDO = P1.5;
sbit             ADSK = P0.0;
/*---------------------------------------------------
 OUT:   R4, R5, R6, R7   R7=>LSB
---------------------------------------------------*/
ReadAD:
    CLR     ADSK                //AD Enable (PD_SCK set low)
    SETB    ADDO                //Enable 51CPU I/0
    JB      ADDO,$              //AD conversion completed?
    MOV     R4,#24
ShiftOut:
    SETB    ADSK                //PD_SCK set high (positive pulse)
    NOP
    CLR     ADSK                //PD_SCK set low
    MOV     C,ADDO              //read on bit
    XCH     A,R7                //move data
    RLC     A
    XCH     A,R7
    XCH     A,R6
    RLC     A
    XCH     A,R6
    XCH     A,R5
    RLC     A
    XCH     A,R5
    DJNZ    R4,ShiftOut          //moved 24BIT?
    SETB    ADSK
    NOP
    CLR     ADSK
    RET
    END
```

Reference Driver（C）

```
//-------------------------------------------------------------------
sbit   ADDO = P1^5;
sbit   ADSK = P0^0;
unsigned long ReadCount(void){
  unsigned long Count;
  unsigned char i;
  ADDO=1;
  ADSK=0;
  Count=0;
  while(ADDO);
  for (i=0;i<24;i++){
    ADSK=1;
    Count=Count<<1;
    ADSK=0;
    if(ADDO) Count++;
  }
  ADSK=1;
  Count=Count^0x800000;
  ADSK=0;
  return(Count);
}
```

Package Dimensions



SOP-16L Package

F Wiring of setup

Wiring the gauges is done following [23], colours are different in some cases. Yet, for all arms of the frame, the same wiring scheme is used. For rod 2, the wiring is shown in figure F.1

Firstly, the Arduino is connected to the HX711 by five wires.
VCC = VDD = 5 volt = red
GND = black
Data = A3 = blue (for rod 2 only, different for each rod)
Clock = A2 = yellow

The HX711 has thus four(five) connectionwires to the arduino. It has 4 wires connected to the wheatstone bridge connector, the fifth (YLW) is not connected.
Red = V+
Blk = V-
Wht = O+
Grn = O-

To this wheatstone bridge connector the strain gauges are connected.
R1 = Red + black
R2 = Yellow + brown
R3 = Purple + green
R4 = Blue + grey

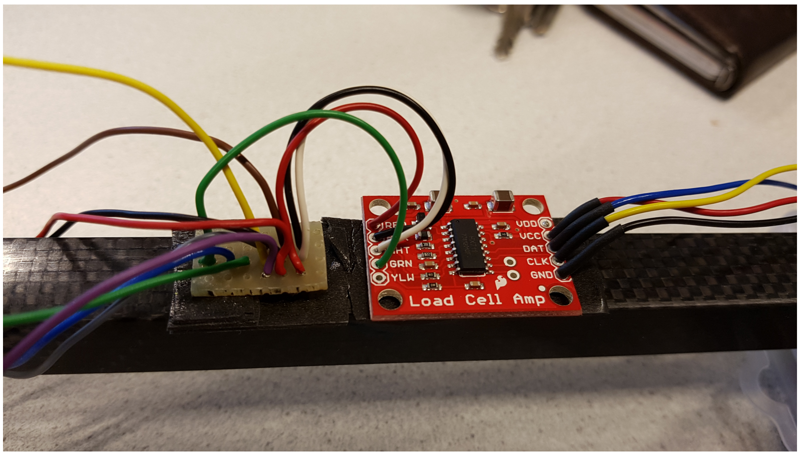


Figure F.1: Wiring of Hx711 and wheatstone bridge

G Arduino code + Matlab scripts

G.1 Arduino code for reading

An explanation of the code is provided in 5.3

```
/*
Arduino pin 2 -> HX711 CLK
3 -> DOUT
5V -> VCC
GND -> GND
*/

#include "HX711.h"

#define DOUT  3
#define CLK  2
long temp = 0;
HX711 scale(DOUT, CLK);

void setup() {
Serial.begin(9600);
}

void loop() {
temp = scale.read();
Serial.println(temp);
}
```

G.2 Matlab scripts for sensor placement

For this main script, values for the rectangular cage design are implemented in the code. A short explanation of main code is given in 3.3.2, while the helper files are added solely for later re-usage of the scripts and are self-explanatory.

G.2.1 Main code

```
%%
clc
clear all
addpath('C:\Users\Thomas\Documents\MATLAB\Cage Design\Helper_Files')

%% Vizualizing Cage and Sensor Locations
figure(1)

% Plot Cage Visualization
plotSphereInsideVolume(1)
% Plot Body Fixed Axis
drawAxis(eye(4))

axis([-1.5,1.5,-1.5,1.5,-1.5,1.5])
grid on
```

```
axis square

%% Defining Sensors

% S1
% Rotation matrix of frame 1 with respect to frame 0
R_0_1 = rodr([1;0;0],90);
% displacement of frame 1 with respect to frame 0
xi_0_1 = [0;-1;0];
% Homogenous matrix of frame 1 with respect to frame 0
H_0_1 = homeMat(R_0_1,xi_0_1);
drawAxis(H_0_1)

% S2:
R_0_2 = rodr([1;0;0],-90);
xi_0_2 = [0;1;0];
H_0_2 = homeMat(R_0_2,xi_0_2);
drawAxis(H_0_2)

% S3:
R_0_3 = rodr([0;1;0],90);
xi_0_3 = [1;0;0];
H_0_3 = homeMat(R_0_3,xi_0_3);
drawAxis(H_0_3)

% S4:
R_0_4 = rodr([0;1;0],-90);
xi_0_4 = [-1;0;0];
H_0_4 = homeMat(R_0_4,xi_0_4);
drawAxis(H_0_4)

% S5:
R_0_5 = rodr([0;0;1],0);
xi_0_5 = [0;0;1.];
H_0_5 = homeMat(R_0_5,xi_0_5);
drawAxis(H_0_5)

% S6:
R_0_6 = rodr([1;0;0],-180);
xi_0_6 = [0;0;-1];
H_0_6 = homeMat(R_0_6,xi_0_6);
drawAxis(H_0_6)

%% Mapping Matrix

Ad_1 = rigidAdjoint(rigidInv(H_0_1))';
Ad_2 = rigidAdjoint(rigidInv(H_0_2))';
Ad_3 = rigidAdjoint(rigidInv(H_0_3))';
Ad_4 = rigidAdjoint(rigidInv(H_0_4))';
Ad_5 = rigidAdjoint(rigidInv(H_0_5))';
Ad_6 = rigidAdjoint(rigidInv(H_0_6))';
```

```
% Assuming that only 1D SENSORS are placed at each frame and that the force
% measured is in the Z-Axis
% So the i-th mapping matrix's vector is always the 6th column of the Adjoint of
% the i-th frame

v1 = Ad_1(:,6);
v2 = Ad_2(:,6);
v3 = Ad_3(:,6);
v4 = Ad_4(:,6);
v5 = Ad_5(:,6);
v6 = Ad_6(:,6);

M = [v1,v2,v3,v4,v5,v6]

%take only the forces (no torques yet)
r1 = M(4,:);
r2 = M(5,:);
r3 = M(6,:);

M_2 = [r1;r2;r3]
cond(M_2)
```

G.2.2 Helper files

drawAxis

```
function drawAxis(H)
% Plot a Right-Handed Coordinate Frame in a given Axes

d = getDist(H);

eps = 0.3;

unitV = [eps 0 0;
0 eps 0;
0 0 1.5*eps;
1 1 1];

V = H*unitV;

% X-Axis Blue
% Y-Axis Green
% Z-Axis Red

hold on
plotAxis(V,d,1,'b')
plotAxis(V,d,2,'g')
plotAxis(V,d,3,'r')
plot3(d(1),d(2),d(3),'ro','MarkerFaceColor','r','MarkerSize',10);

hold off
```

```
function plotAxis(V,d,i,c)
x = V(1:3,i);
plot3([d(1) x(1)],[d(2) x(2)],[d(3) x(3)],c,'linewidth',3)
```

getDist

```
function d = getDist(H)
% Extract Distance vector from a Homogeneous matrix
d = H(1:3,4);
```

homeMat

```
function H = homeMat(R,d)
% Construct Homogeneous Matrix
d = [d(1);d(2);d(3)];
H = [R,d;zeros(1,3),1];
```

plotsphereinsidevolume

```
function plotSphereInsideVolume(rad)
hold on
[x,y,z] = sphere;
sph_Hand = surf(x*rad,y*rad,z*rad);
set(sph_Hand,'FaceColor','r')
alpha 0.01
hold off
```

rigidAdjoint

```
function AdjH = rigidAdjoint(H)
% Computing Adjoint of Homogenous matrix

R = H(1:3,1:3);
d = H(1:3,4);

AdjH = [R,zeros(3,3);skew(d)*R,R];
```

rigidInv

```
function Hinv = rigidInv(H)
% Implement Analytical formula for computing inverse of Homogenous matrix
R = H(1:3,1:3);
d = H(1:3,4);

Hinv = [R',-R'*d;zeros(1,3),1];
```

rodr

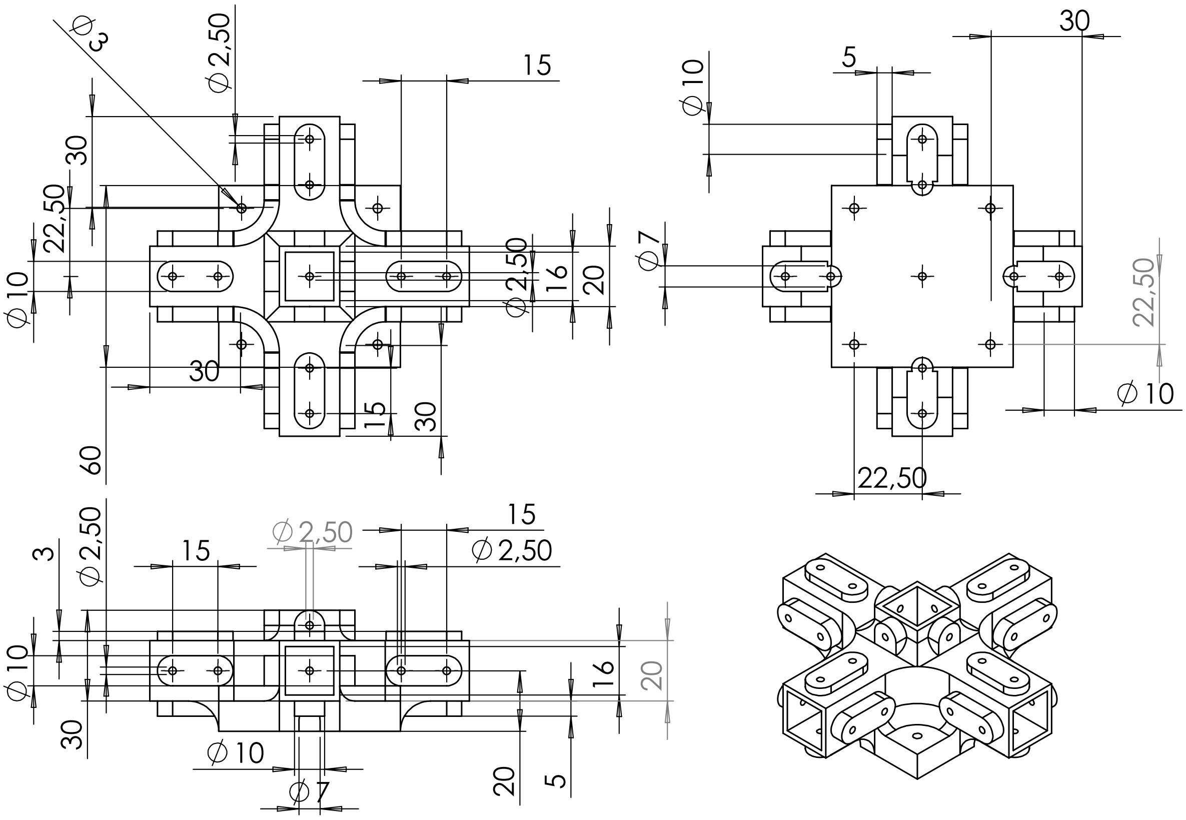
```
function R = rodr(axis,angle)
% Impement Rogriguez Formula
% Angle is in degree

% Tilde form of axis
w_t = skew(axis);
R = eye(3) + w_t*sind(angle) + w_t*w_t*(1-cosd(angle));
```

skew

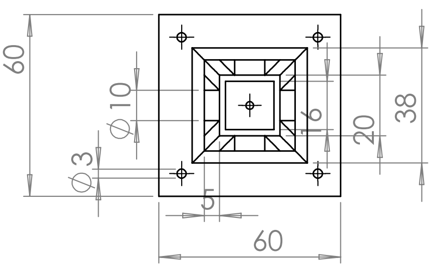
```
function w_t = skew(w)
% Construct skew-symmetric matrix given a 3D vector

x = w(1);
y = w(2);
z = w(3);

w_t = zeros(3,3);
w_t(3,2) = x;
w_t(2,3) = -x;

w_t(1,3) = y;
w_t(3,1) = -y;

w_t(2,1) = z;
w_t(1,2) = -z;
```

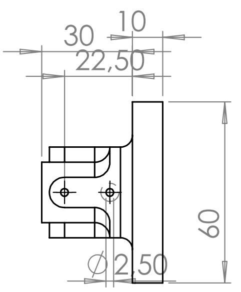
H Dimensions of cage/parts

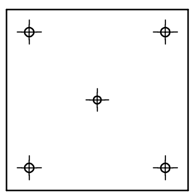
Drone to frame connector bottom

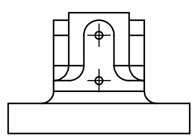


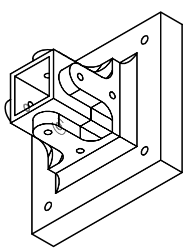
Drone to frame connector bottom



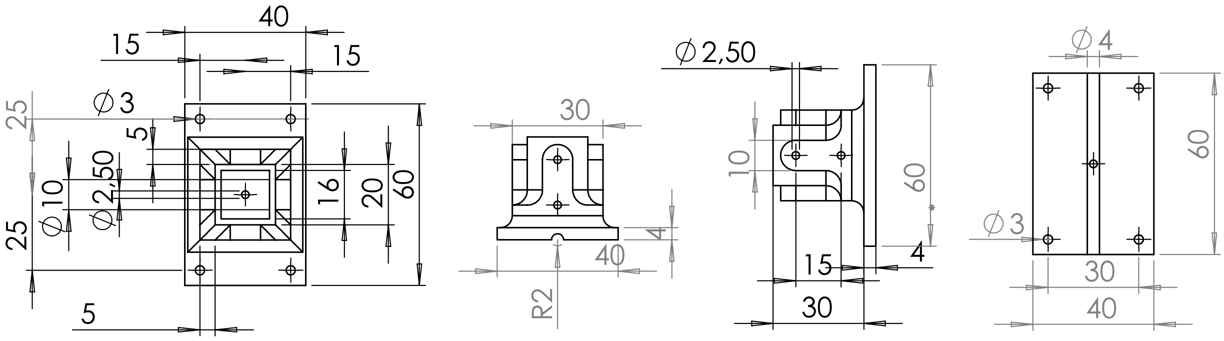


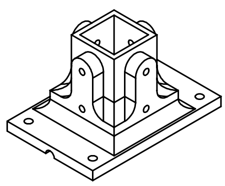




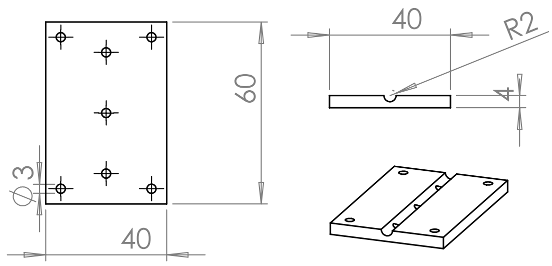


Drone+frame to cage connector top

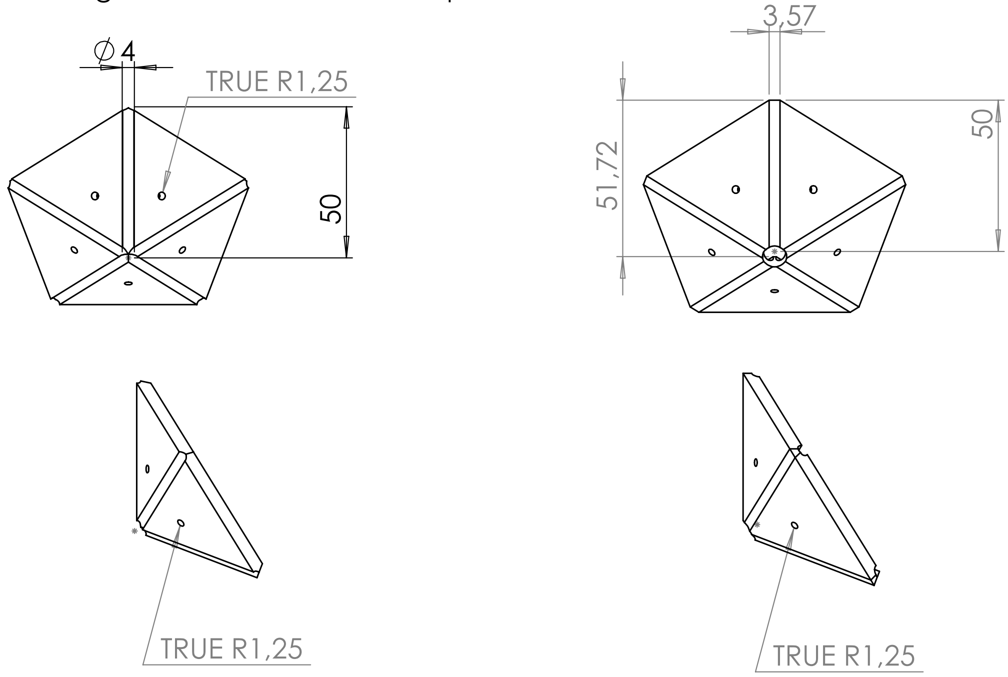


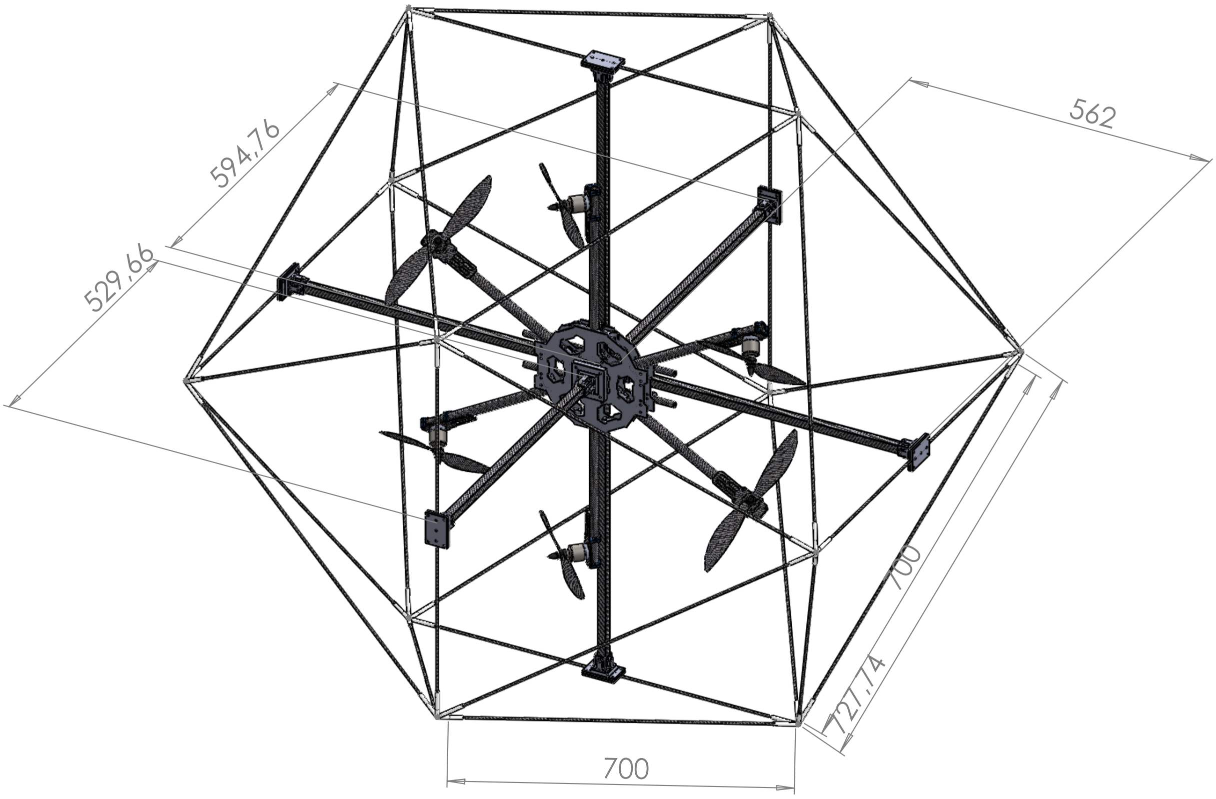


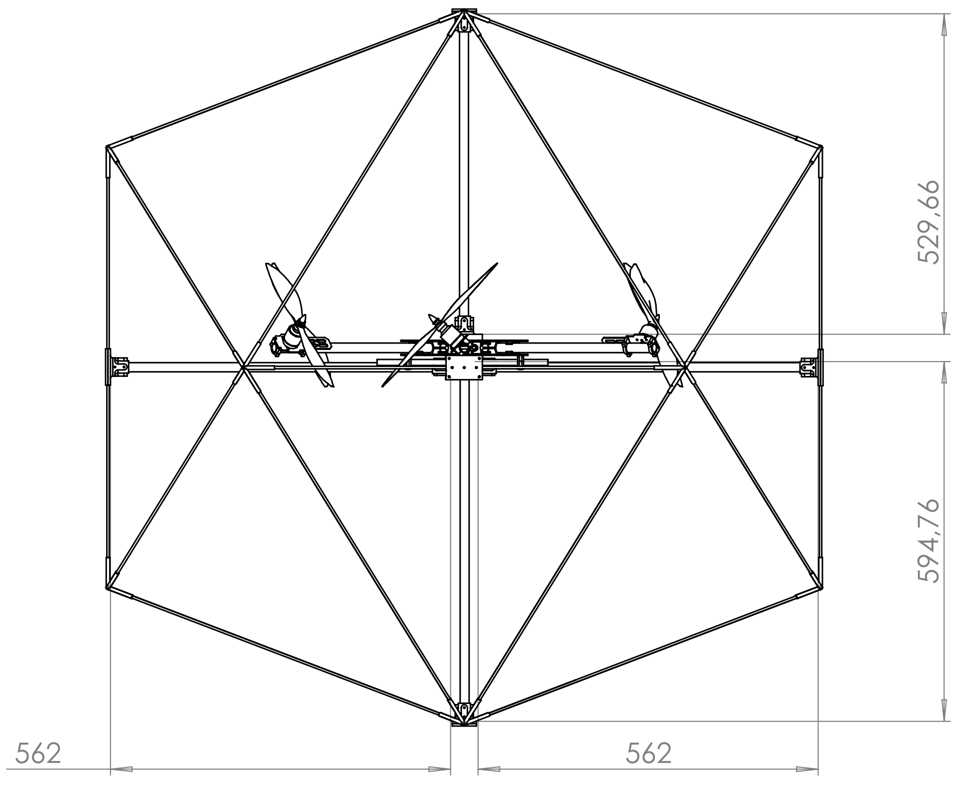
Drone+frame to cage connector bottom



cage to cage connector bottom - top




562
594,76
529,66
700
727,74
700


529,66
594,76
562
562

Bibliography

[1] Spectors, “Sensing the world,” 2016.

[2] Quanzer, “Qball2 - high-performance drone for indoor labs,” 2018.

[3] Agilent Technologies Inc., “Practical strain gauge measurements,” 1999.

[4] WikiCommons, “Straingaugevisualization,” 2011.

[5] HBM, “Tools for selecting the right strain gauges.”

[6] HBM, “Series universal strain gauges.”

[7] National Instruments, “Measuring strain with strain gages,” 2016.

[8] N. Gittins, “Short guide to strain gauging methods.”

[9] Micro Measurements, “Strain gage selection: criteria, precedures, recommendations,” 2018.

[10] HBM, “How to find the strain gauge that best suits your application.”

[11] H. N. Soundararajan, “Torque transducer sensitivity study,” 2014.

[12] Omega, “Positioning strain gauges.”

[13] Performance Composites Inc., “Carbon fibre composite design guide.”

[14] Tapplastics, “Pultruded carbon rods and tubes.”

[15] Sparkfun, “Sparkfun load cell amplifier - hx711.”

[16] Sheng A. Liu and Hung L. Tzo, “A novel six-component force sensor of good measurement isotropy and sensitivities,” pp. 223–230, 2002.

[17] M. M. Svinin and M. Uchiyama, “Optimal geometric structures of force/torque sensors,” vol. 14, December 1995.

[18] C. Collins, “Condition and stability.”

[19] Dassault SystÃÍmes SolidWorks Corporation, “Solidworks 2017 (software),” 2018.

[20] “Tera term pro 2.3 (software).”

[21] ATI industrial automation, “F/T sensor: mini 40.”

[22] Bogde (Github), “Hx711 master.”

[23] Sparkfun, “Load cell amplifier hx711 breakout hookup guide.”

[24] PurdueMET, “Strain gauge 2 - gauge factor.”

[25] J. Daniels, “Strain gauges.” 2014.

[26] G. Thorpe, “Eme 3214 wheastone bridge circuits,” 2013.

[27] J. Engelen and G. Folkertsma, *Control for UAVs.* 2018.