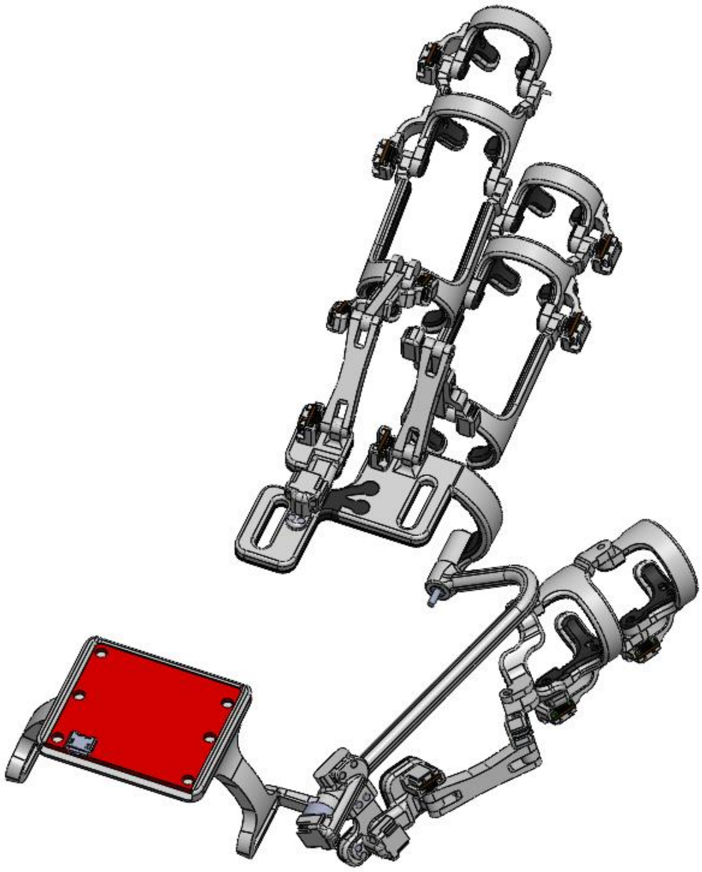
Exoskeleton Handover



1 Introduction

This project built on the previous exoskeleton work Antonia Tzemanaki had done in her PhD and subsequent projects.

The general aim was "To develop a comfortable (lightweight), easy to use, accurate and reliable 3-finger exoskeleton interface for tracking of fingers and wrist of a human operator during precision tasks"

1.1 Software required

- Solidworks 2021 (or later) - As of writing this can be downloaded through a UWE account via appsanywhere
- PulseVPN - This is required for the Solidworks licence and can be downloaded from appsanywhere, there is a simple guide on the appsanywhere information page on how to setup.
- Autodesk Eagle – If you sign-up to Autodesk using your university account you can download and access this software for free.
- Arduino IDE
- Text file editor e.g. notepad, notepad++, sublime text

2 Previous Work

In her PhD, Antonia Tzemanaki produced an exoskeleton design for hand tracking, see Figure 1. This was the basis of this project. In her PhD thesis it is recommended to read Chapter 4 'Master manipulator' on the develop of her previous exoskeleton. An awareness of chapter 6 is also beneficial for the kinematic mapping between the exoskeleton and a potential secondary (follower) system.

Figure 1 - Past Exoskeleton

Design features to improve were:

- Comfort – As this is subjective this was broken into fit, weight and range of motion
- Design complexity –The adjustable nature of the exoskeleton required a number of parts for the mechanism. This should be made simpler if possible.

3 Electronics

3.1 Melexis 90316 – Magnetic Rotatory Encoder

The exoskeleton utilises magnetic rotary encoders to measure the angles at joints. These are well suited to this application because they can be very accurate, provided fast feedback and does not add any resistance to the movement of the rotating object as it senses in a 'contactless' fashion.

The sensor used is the Melexis 90316. This is an absolute, not an incremental, encoder. These sensors can come with a factory calibration or not. If they do not come with a calibration they will need to be calibrated before data can be read from them. Ask Antonia about calibrating the sensors as this requires specialist equipment in the BRL.

The sensors can be used in Analog, PWM or SPI mode. For this application analogue mode is used.

They measure the magnetic field of a small dipole magnet, the magnetic specification can be found in chapter 10 of the Melexis 90316 datasheet. This project uses 3mm diameter and 1mm thickness diametric magnets sourced from HKCM engineering.

The magnet should be a distance of 1mm away from the sensor face and also aligned asymmetrically (0.5mm vertical offset) for best possible results. See chapter 19.3 "SOIC-8 – IMC Positioning" in the Melexis 90316 datasheet for specific details.

3.2 Melexis Sensor Custom PCM

The sensor are soldered onto a custom PCB made by Jason Welsby (BRL Technician). The PCB files can be viewed using Autodesk Eagle and are found in the shared OneDrive in "*Antonia's Personal Repository > Past exoskeleton files > PSBs > Melexis 1DOF > 5x8 – jst.*"

Figure 2 - Melexis sensor and custom PCB with JST connectors

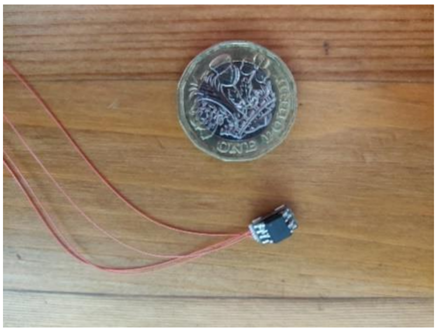


Figure 3 - Melexis sensor with pre-crimped cable and £1 coin for scale

These custom PCBs enables daisy chaining of the Melexis sensors. Each PCB has the option to connect one of two "zero ohm resistors" this routes the signal coming from the sensor either to the OUT1 or OUT2 channel of the JST connector attached to the PCBs underside, see Figure 4 and Figure 5. The JST connectors are wired so a cable can be plugged into either side of the board. For daisy chaining to work a PCB with the OUT1 signal needs to be connected to a PCB with the OUT2 signal, otherwise the sensors output will interfere will each other providing inaccurate data. For example in … this sensor will provide an … signal. This is the case

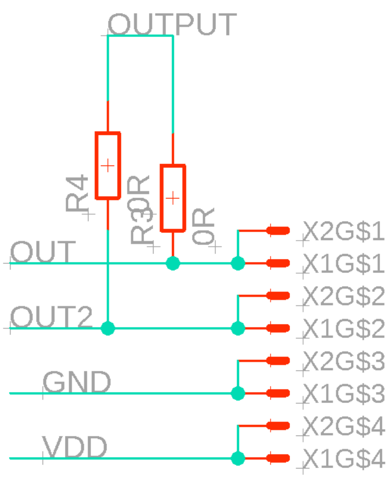


Figure 4 - Melexis custom PCB connector schematic

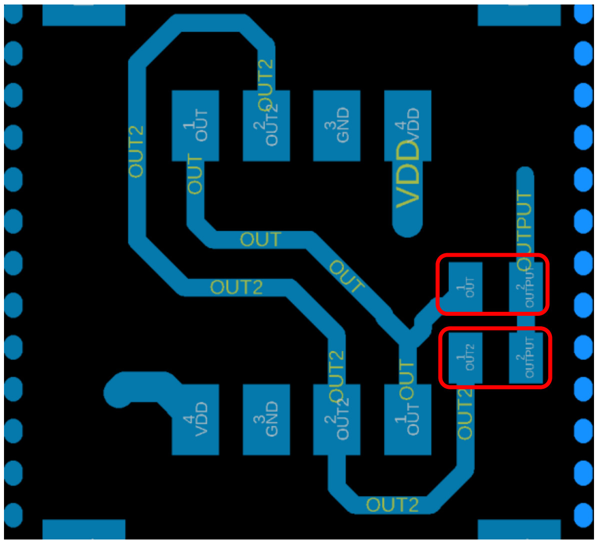


Figure 5 - Melexis custom PCB connector wiring

3.3 Sensor Cables

The sensors are daisy chained or connected to the main electronics board via pre-crimped cables, see Figure 3, which can be sourced from suppliers like [RS components]() or Farnell in various predefined lengths.

3.4 Mainboard

The mainboard for the exoskeleton project was custom made by Jason Welsby. The main features of this board is the [SAMD21G18A QFN]() microcontroller and the two [MCP3208]() analogue to digital converters (ADC). The board can support up to 16-24analogue input signals. The microcontroller has a 12-bit 8 channel in-built ADC on the microcontroller however not all ADC are functional, ask Antonia or Jason for more details. Each of the ADCs on the board are 12-bit 8 channel ADC as well. Each JST connector takes in two analogue signals, see PCB file for how these are wired up to the microcontroller and ADCs. The PCB file can be found in the shared Onedrive in "*Antonia's Personal Repository > Past exoskeleton files > PSBs > exo_data*". It is the one with 'SAMD21' in the file name.

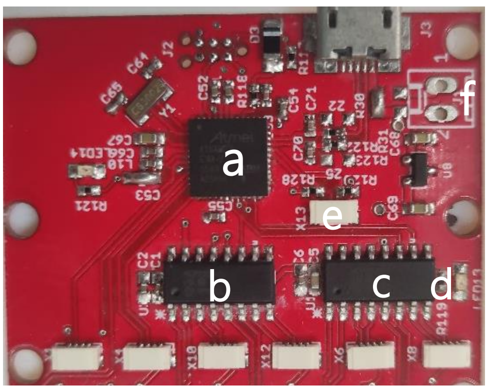


Figure 6 - Mainboard. a) Microcontroller. b) & c) ADC. d) Built-in LED. f) Power supply pins. e) i2c bus via jst connector

3.4.1 Microcontroller

The SAMD21G18A QFN can be programmed through the Arduino IDE using the Arduino Zero or a custom board file which Jason Welsby can provide. The pin layout when referenced in the Arduino IDE is not standard and requires asking Jason for a reference. Ask Antonia for the files.

3.4.2 Analogue to digital converter

The two MCP3208 are 8-channel, 12-bit, ADCs that communicate via SPI protocol. Use these 16 outputs first. If more outputs are needed, use the build-in. By default the MCP3208 ADCs are at a higher resolution than the inbuilt ADC but you can set the inbuilt ADC to 12-bit resolution as well using the analogReadResolution() function..

4 Design

The exoskeleton design is parametric, meaning that with hand measurements it is possible to provide a customised fit for the exoskeleton. However, hand measurements cannot accurately been taken due to a) human error and b) skin moves and hands change from day to day due to temperature etc. A good fit enables better joint alignment when preforming tasks.

The digits of the exoskeleton are based on an open ring design as this is a common item that is worn on the fingers and thumbs and techniques and tools exist to measure them. A ring sizing tool, Figure 7, used by jewellers has been purchased for that which should be in the tool draws located in Bay 15 When using it, make sure that the ring chosen is comfortable on the corresponding phalange of the finger. Keep measurements in the same file, and compare differences between left and right hand as taking them, to spot errors and re-measure if needed. See Appendix 1 on how to take measurements.

Figure 7 - Ring sizer tool

4.1 CAD

The CAD for the exoskeleton is done in SolidWorks. The files are organized by the digit of the hand, the wrist or common parts, Figure 8. A unifying assembly file "handAssm.SLDASM" contains all the sub-assemblies and parts for a full overview.

To gain an understanding of how individual parts are designed it is recommended to open parts and roll back the feature tree and then roll it forwards feature by feature.

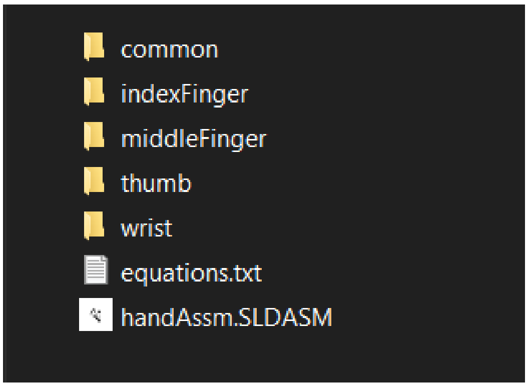


Figure 8 - Exoskeleton folder structure

Parts which are required to be re-sized based on hand measurements all have equations. These equations modify the geometry based on hand measurements and inputted into the text file ‘equations.txt’. Many different parts take values from this file so changes will affect multiple parts at once so it is advised to rebuild the main assembly or relevant sub assembly to ensure changes are as expected.

Due to the parametric nature of the parts, there are values which may cause errors or features to fail to re-generate. This could be because of a number of things. Double check you changes make geometrical sense. If the changes are geometrically sensible but still cause rebuild errors SolidWorks may have lost a reference due to a faces or constraints disappearing which is common. Normally it is possible to re-constraint features/sketches to remove errors. Fillets are best to not be included in the parametric design. Edges and parts should be filleted and made smooth looking after the CAD has been adjusted to a specific measurement. This will help with the breaking of the parametric design and should take less time to complete.

Equations can be viewed and changed by selecting Tools>Equations,

Sometime when copying files SolidWorks will keep referencing an old equations.txt file so ensure your parts are using the correct one by checking the external reference path, highlighted in Figure 10. To update this path just click the button with three dots next to the path and browse to the relevant equations file. If prompted by Solidworks to overwrite equations in the external file select ‘No’. This is important as Solidworks will erase all equations if you allow it to overwrite the file.

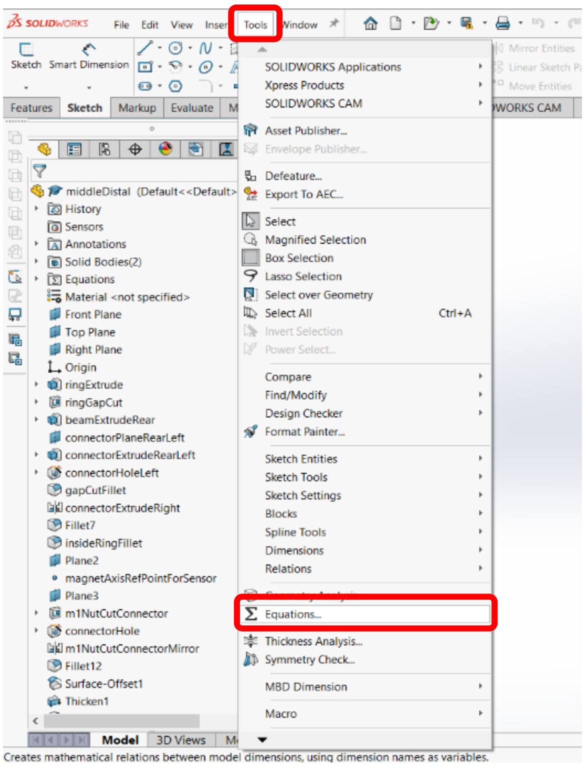


Figure 9 - Opening SolidWorks equations dialog

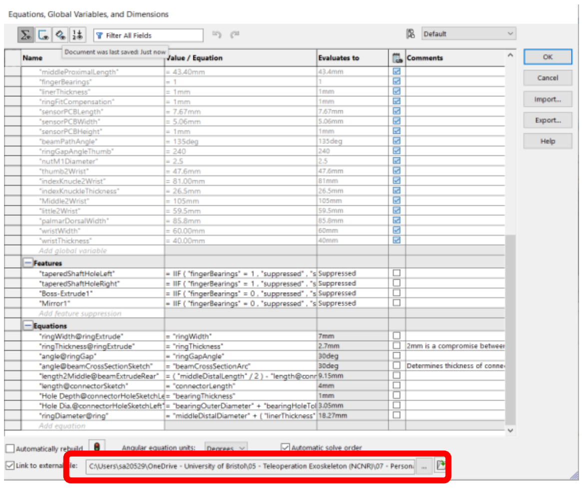


Figure 10 - Solidworks equations dialog box

4.1.1 Right hand parts

Currently a full left-hand assembly exists; however, if parts for a right-handed exoskeleton are required it is recommended to copy the existing set of parts into a new folder. It is then possible to open all asymmetrical parts and change the configuration to the right-handed versions. This should provide a mirrored parts which can then be printed and assembled. If changes have been made to the parts, ensure the mirror and delete/keep body feature is at the bottom of the SolidWorks feature tree to ensure the latest modification has been applied to this configuration.

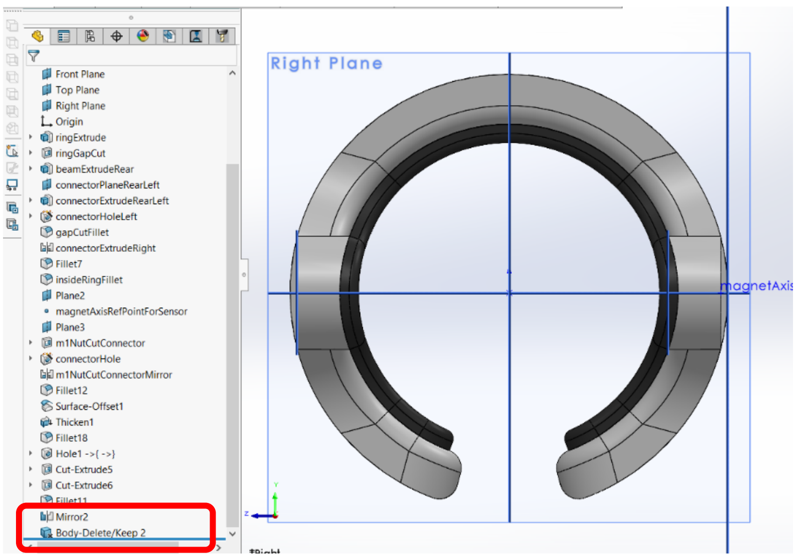


Figure 11 - Delete/Keep Body in the feature tree

It would be beneficial to build a full right hand assembly to minimise the length of this procedure, however to keep two different assemblies composed of unique left-handed and right-handed parts in sync with each other would be time consuming. Therefore it is recommended to either make a right-handed configuration in the existing top level assembly and re-assemble all the individual parts using their 'right' configuration, alternatively a second top level assembly could be made using parts in their 'right' configuration. For simplicity the second method is recommended.

Parts for both assemblies need to also have identifiers added to them (small embedded writing) as this will help in the assembly process.

4.2 Non-3D printed mechanical part list

Below is a table of the number of non-3d printed part required per hand.

Component	Number
Diametric magnet	18
Melexis sensors	18
M1 nut	49
3mm OD Bearings	58
M2 nut	2

5 Manufacture

The design of the exoskeleton is heavily influenced by the manufacturing process. The 3D printed parts made using the Objet Connex 260 3d printer from Stratasys. This is a SLA 3d printer that uses resin and is cured by light. This offers the possibility to have very fine details on small parts which would not be possible using more common FDM printers, like the Ultimaker S5, which is critical for producing some of these parts. This unfortunately means this design it limited to the select number of materials available to this printer.

Another feature of the design that is enabled by the Objet printer is multi-material printing. The printer is able to print with multiple materials at a time. This enables features in the exoskeleton design such as the soft lining on the inside of the finger parts or the flexible section on the palm. To learn more about how to print using multiple materials on the Objet printer ask Sam Coupland (Technician).

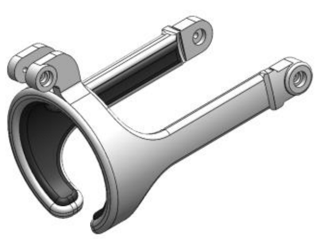
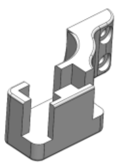
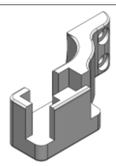
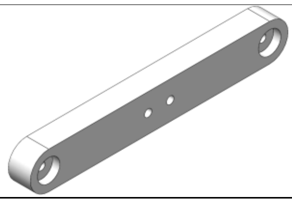
This printer uses support material that must be cleaned off. Unfortunately this support material cannot be dissolved in the rapid prototyping lab and requires the water jet cleaner, it is recommended to get signed-off on how to use this so you can clean parts yourself and not wait on Technicians to be free. This cleaning can take up to 1-2 hours for a full exoskeleton print, depending on proficiency and how well the glovebox gloves fit you. It is recommended to clean the majority of the support material before using the water jet and use special tweezers to clean holes for bearings/nuts/screws.

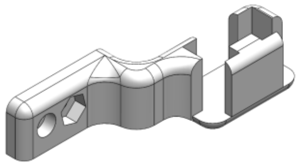
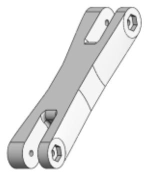
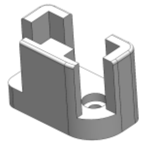
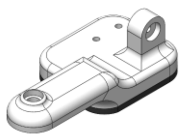
6 Assembly

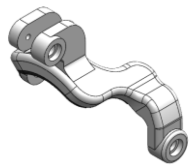
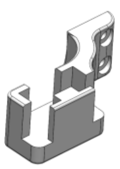
6.1 Part identification

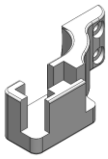
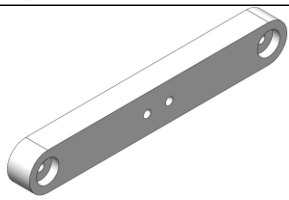
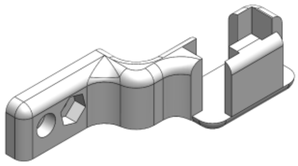
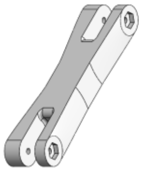
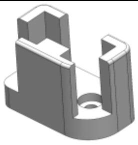
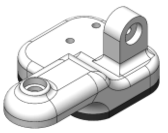
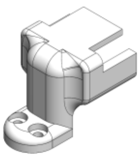
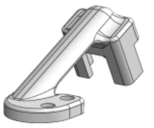
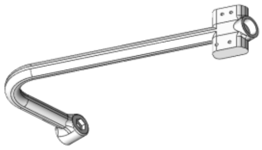
It has been possible to distinguish parts however this requires an intimate knowledge of the design and some validation measurements. To make assembly easier it is recommended to come up with a shorthand for each part ideally also taking location into account. For example all parts related to the thumb could start with 1 followed by that part number e.g. 100 could be the thumbDistal, 200 indexDistal, 300 middleDistal. These codes can be embossed onto parts.

A full list of items here, with pictures and labelling would be useful. Also which screws are what length in the picture etc.

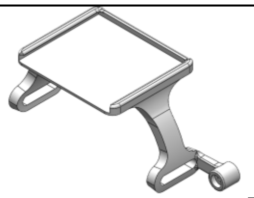
Index/Middle Parts	
indexDistal/middleDistal	
indexMiddle/ middleMiddle	
indexProximal/ middleProximal	
indexDistalSensorMount/ middleDistalSensorMount	
indexDistalSensorMount/ middleDistalSensorMount	
indexMCPTriA/ middleMCPTriA	

indexMcp2SensorMount/ middleMcp2SensorMount	
indexMCPTriB/ middleMCPTriB	
indexMcp3SensorMount/ middleMcp3SensorMount	
indexDorsal	
midddleMcpDorsal	

Thumb parts	
thumbDistal	
thumbProximal	
thumbMetacarpal	
thumbDistalSensorMount	

thumbProximalSensorMount	
thumbMCPTriA	
thumbMcp2SensorMount	
thumbMCPTriB	
thumbMcp3SensorMount	
thumbCMCDorsal	
thumbWristMount	
thumbAbductionSensorHolder	
thumbPalmarAbductionSensorHolder	
alignmentRod	

Wrist parts	

wristRingTop	

6.2 Tools

It is recommended to use the small screwdriver set and bent tweezers to make manipulating small components such as nut/screws/bearings easier.

Figure 12 Screwdriver set and tweezer

7 Issues

The following are issues which have been noted with this version of the exoskeleton.

7.1 Hand measuring

As hand measurements will have human error due to imprecision and interpretation. Also skin moves and hands change from day to day due to temperature. Therefore a perfect fit will not be possible even with perfect parametric equations however better alignment of joints will improve the performance of hand tasks.

7.2 Thumb palmar adduction hardware

An issue on the previous exoskeleton was the coupling of the thumb palmar abduction/adduction and it's radial abduction/adduction. Therefore a new mechanism was implemented in this version in an attempt to decouple these on the exoskeleton, however some coupling of these DoF is still present and may affect the accuracy of the calculated kinematics.

7.3 Thumb 'proximal' part

The proximal thumb part does not still on the surface of the users thumb. This can cause movement in the thumb PIP joint to not be registered.

7.4 Thumb palmar pieces

Depending on the dimensions of the users hand these components might be longer than the distance between the wrist of the base of the proximal phalange and cause them to not sit on the surface of the users hand.

7.5 Strapping

The way the exoskeleton is strapped to users allows for some components to not sit directly onto the skin of the user which might allow for unregistered movement of the hand.

7.6 Loose screws

To enable free movement of the finger joints the joints of the exoskeleton must move freely. To enables this the screws used cannot be tightened fully. Therefore after prolonged use the screws may become loose and reduce the structural integrity of the exoskeleton. This may be overcome using thread lock or some grease on the screws.

8 Recommendations

These have been mentioned previously but a right hand assembly and part identification number debossed onto individual parts would provide initial familiarity with the design and save time in the manufacturing and assembly process

9 Appendix 1 – Hand measurements

To update the parametric design for a new user their hand measurement has to be inputted into the equation.txt file. This is done via measuring the hand in three steps.

Firstly the ring sizes of the index, middle and thumb phalanges are measured using the ring sizing tool, see Figure 7. When taking the ring measurements new users are advised to pick a ring size which is can slide on and off relative easy but sits snuggly at the desired location.

All measurements should be recorded into this excel template file, create a copy before adding measurements. There is also a ring size conversion chart to mm.

The second step is to measure the lengths of the phalanges using a digital vernier calliper.

The proximal phalanges should be measured roughly where the ring would sit at the base of the finger to the PIP joint as in Figure 13.

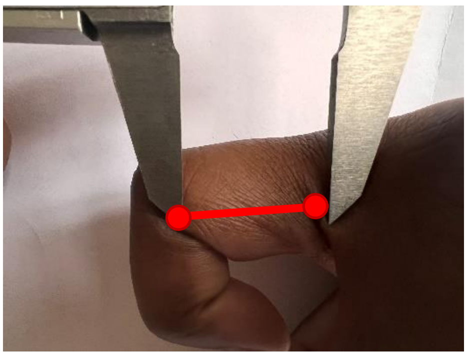


Figure 13 - Measuring index proximal phalange with vernier calliper

For the middle phalanges the centre of the phalange should be measured from the estimated point of rotation indicated by the red dots as indicated in Figure 14 - Measuring index middle phalange with vernier calliperFigure 14.

The distal phalanges should be measured from the centre of rotation of the DIP joint to the tip of the finger.

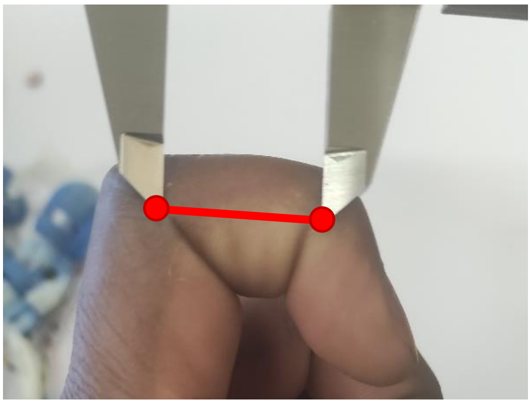
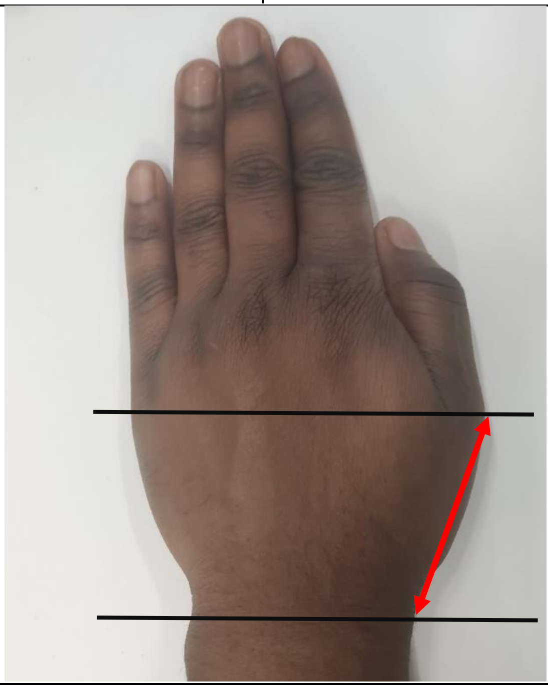
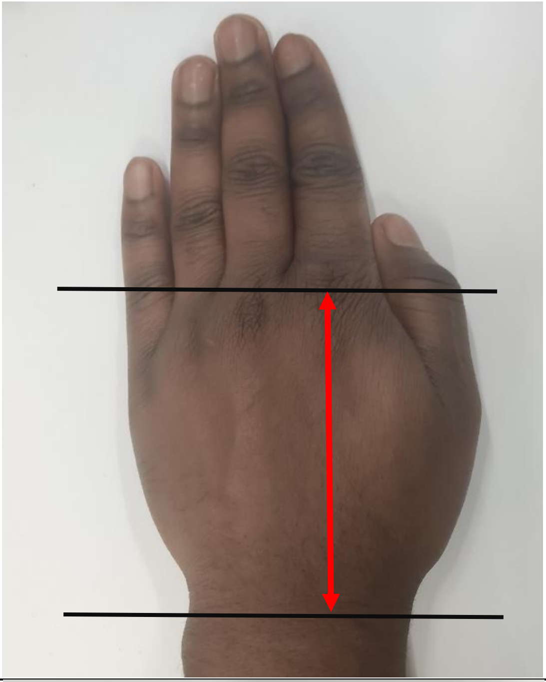
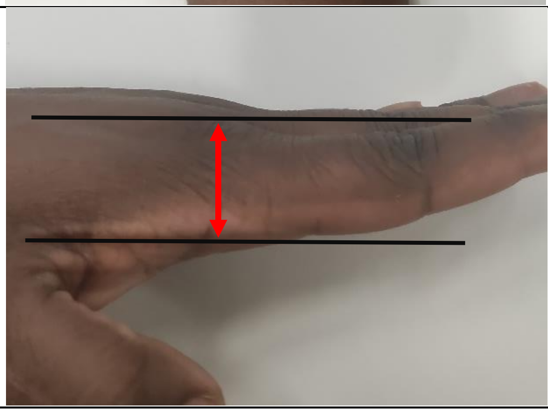
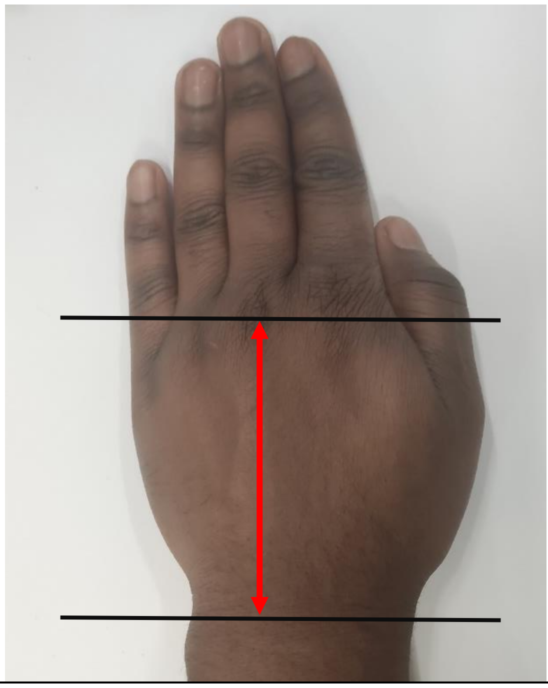
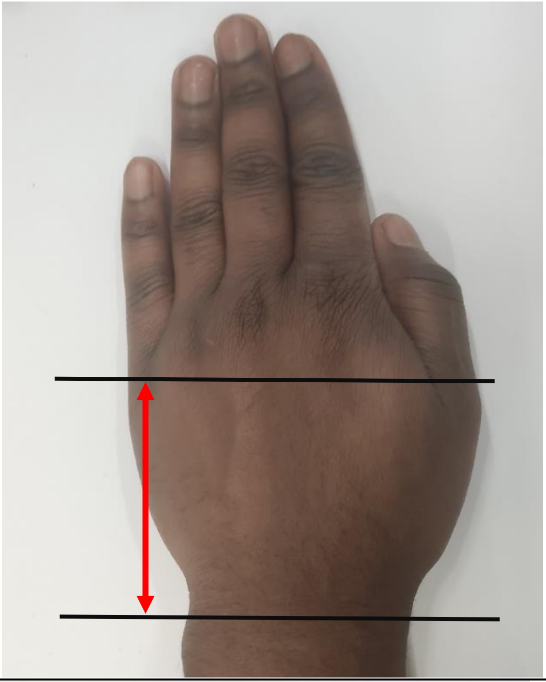
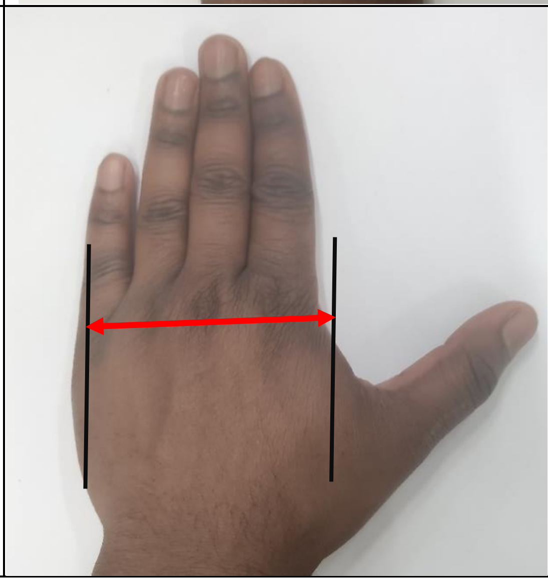


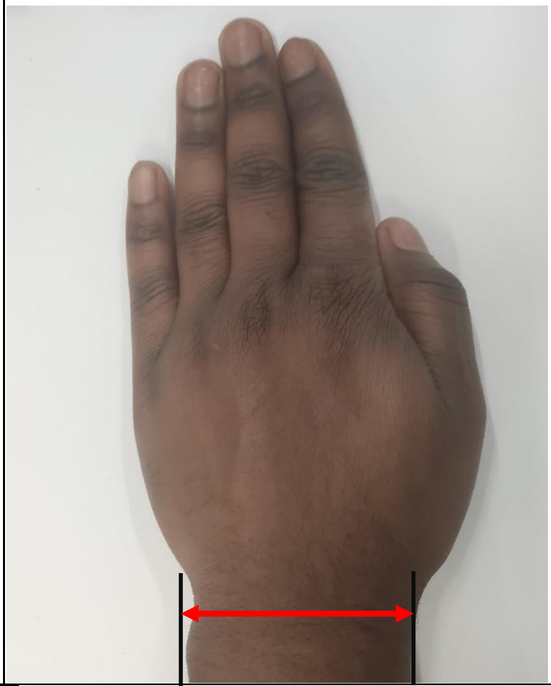
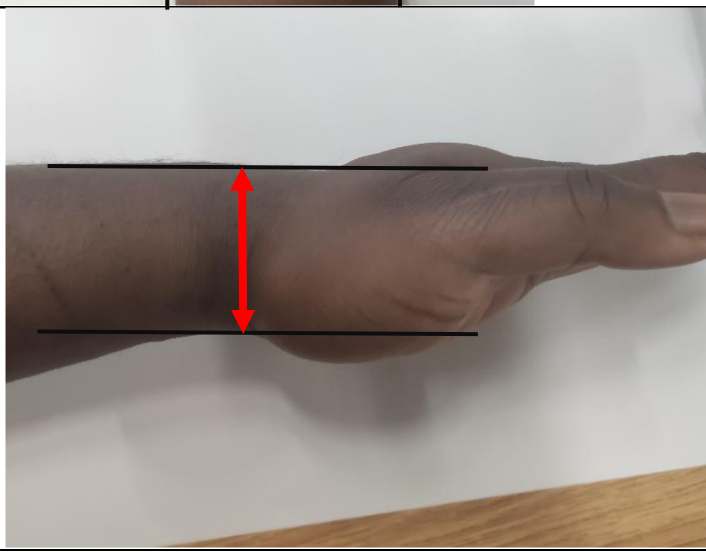
Figure 14 - Measuring index middle phalange with vernier calliper

The third step is to use the vernier callipers again to measure the miscellaneous dimensions of the hands.

Measurement label	Measurement Example
thumb2Wrist	

indexKnuckle2wrist	
indexknuckleThickness	
middle2Wrist	

little2Wrist	
palmarDorsalWidth	

wristWidth	
wristThickness	

10 Appendix 2 – Formulas

10.1 Formula glossary

Name	Unit	Comment
"thumbDistalDiameter"	mm	

"thumbProximalDiameter"

"indexDistalDiameter"

"indexMiddleDiameter"

"indexProximalDiameter"

"middleDistalDiameter"

"middleMiddleDiameter”

"middleProximalDiameter"

"thumbDistalLength"

"thumbProximalLength"

"indexDistalLength"

"indexMiddleLength

"indexProximalLength"

"middleDistalLength”

"middleMiddleLength"

"middleProximalLength"

"thumb2Wrist"

"indexKnucle2Wrist"

"indexKnuckleThickness"

"Middle2Wrist"

"little2Wrist"

"palmarDorsalWidth"

"wristWidth"

"wristThickness"

"ringWidth"

"ringThickness"

"ringGapAngle"

"ringGapAngleThumb"

"beamCrossSectionArc"

"bearingOuterDiameter"

"bearingHoleTolerance"

"bearingThickness"

"magnetThickness"

"magnetHoleTolerance"

"magnetDiameter"

"printerNozzleDiameter"

"bearingInnerDiameter"

"connectorLength"

"sensorLengthPlaceHolder"

"sensorConnectorWidth"

"bearingShaftHoleTaperAngle"

"sensorPCBLength"

"sensorPCBWidth"

"sensorPCBHeight"

"beamPathAngle"=

"bearingShaftDiameter"

"ringFitCompensation"

"linerThickness"

"fingerBearings"

"nutM1Diameter"

"linerFilletRadius"

"sensorRecess"

"sensor2magnet"

10.2 – Proximal, Middle and Distal Formulas

Name	Equation/Value	Comments
ringWidth@ringExtrude	ringWidth	Fixed value decided by user
ringThickness@rinExtrude	ringThickness	Fixed value decided by user: This value was used as a minimum thickness to allow for the M1 nut and bearing to be embedded into the part.
angle@ringGap	ringGapAngle	Fixed value decided by user. Size of gap cut in the bottom of the ring.
angle@beamCrossSectionSketch "arc@beamCrossSectionSketch"	beamCrossSectionArc	Fixed value decided by user. Size of the arc for the beam that connectors to the next part. E.g. the beam which connects the distal part to the middle part
length2Middle@beamExtrudeRear length2Proximal@beamExtrudeRear	(middleDistalLength/2) – length@connectorSketch (middleMiddleLength/2) – length@connectorSketchRear	Variable value dependant on hand measurements. The length of the beam to the estimated centre point of the joint taking into account the length required by the connector which houses the bearing, m1 nut and screw.
length@connectorSketch length@connectorSketchFrontLeft length@connectorSketchFrontRight	connectorLength	Fixed value decided by user. This is the length of the connector which is the where the bearing, m1 nut and screw as housed for these parts
Hole Depth@connectorHoleSketch	bearingThickness	Fixed value decided by user. This is the depth

		of the cut for the embedded bearing
Hole Dia.@connectorHoleSketch	bearingOuterDiameter + bearingHoleTolerance	Fixed value decided by the user. This is the diameter of the cut required for the embedded bearing.
ringDiameter@ring	middleDistalDiameter + (linerThickness*2) + ringFitCompensation	Variable value dependant on hand measurements. Inside diameter of the ring section. This taking into account the thickness of the rubber lining on both sides of the ring and also an optional compensation to increase or decrease the diameter to ensure a better fit.
lateralDistance@beamPath	ringThickness - ((middleMiddleDiameter - middleDistalDiameter) / 2)	Variable value dependant on hand measurements. This is the distance radially that a beam going towards the fingertip has to extend so the beam faces coincide with each other at a joint. This is the thickness of the ring minus the difference in radius of the two parts to be connected.
majorLength@beamPath	(middleMiddleLength / 2) - minorLength@beamPath - length@connectorSketchFrontLeft	Variable value dependant on hand measurements. This is the variable part of the length of the beam required to meet the next part at the finger joint. This is half the middle phalanges length minus the fixed length of the beam determined which is the minor length minus

		the length of the connector which is where the bearing, m1 nut and screw as housed for these parts.
Hole Depth@magnetHolderHoleSketch Hole Depth@connectorHoleRightSketch		Fixed value decided by the user. This is the depth of the cut for the magnet
Hole Dia.@magnetHolderHoleSketch Hole Dia.@connectorHoleRightSketch		Fixed value decided by the user. This is the diameter of the cut required for the magnet

10.3 – Index Dorsal Palmar Formulas

Name	Equation/Value	Comments
D3@Sketch6	((indexProximalDiameter + middleProximalDiameter) / 2)	Variable value dependant on hand measurements. Distance between the two centrelines of the middle and index fingers .
D4@Sketch6	indexKnucle2Wrist / 2	Variable value dependant on hand measurements.
D1@Sketch37	indexKnuckleThickness * 1.25	Variable value dependant on hand measurements. This is the thickness of the hand at the index knuckle with a scaling factor to accommodate extra space for flex and movement.
D2@Sketch37	indexProximalDiameter/2 indexProximalDiameter - 14 + 3	Variable value dependant on hand measurements. This maximum value of the curve. A suitable equation based on the

		hand dimensions has not been finalised either of these two seem to produce okay results. In the second equation, the 14 was a fixed value which brought most values into a sensible range. The +3 is to account for the thickness of the part. The first equations sometimes produced errors or strange results for people with large diameter fingers.

10.4 – Alignment Rod Formulas

Name	Equation/Value	Comments
D1@Sketch1	indexKnucle2Wrist	Variable value dependant on hand measurements. This is the distance between the knuckle and the wrist
D5@Sketch1	indexKnucle2Wrist / 2	Variable value dependant on hand measurements. This is a good ratio for this dimension.

10.5 – Wrist Formulas

Name	Equation/Value	Comments
D1@Sketch1	wristWidth + 5	Variable value dependant on hand measurements. This is the distance between the knuckle and the wrist
D2@Sketch1	(wristThickness / 2) * 1.25	Variable value dependant on hand measurements. This is a good ratio for this dimension.