A User Interface for a Seven Degree of Freedom Surgical Robot

by
Jacobus Stephanus Heunis

*Thesis presented in fulfilment of the requirements for the degree of
Master of Science in the Faculty of Engineering at
Stellenbosch University*

Supervisors: Prof. Cornie Scheffer and Prof. Kristiaan Schreve

December 2012

DECLARATION

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: ....

i

ABSTRACT

This thesis describes the process of developing a user interface for a seven degree of freedom (DOF), minimally invasive surgical robot. For the first two main stages of the overall project, completed by previous students, a primary slave manipulator (PSM) and a secondary slave manipulator (SSM) were developed. The stage in this thesis concentrates on creating a joystick that can control the combined movement of the PSM and SSM.

Background information on the field of robotic surgery, with specific reference to current systems' user interfaces, is given and the technical aspects of the PSM and SSM are determined. This is followed by the motivation and main objectives of the thesis. Objectives were divided into the main categories of mechanical design, electronic design, control system design and testing.

The mechanical design of the joystick progresses through a concept development stage, before a final seven DOF articulated arm design is presented and evaluated based on engineering specifications. Aluminium is used as the construction material; electromagnetic brakes are specified for each joint, leading to the final assembly, which is a constructed joystick fulfilling all requirements.

The electronic design implements magnetic rotary encoders for the joystick's position and orientation tracking as well as designs of the necessary power and control circuitry to enable correct joystick functioning. The interfacing of the PSM and SSM had to enable successful communication capabilities between the master and the slave. Several necessary adjustments were therefore made to the slave system, after which the joystick and robot were electronically interfaced to provide a direct serial communication line.

For control system design, the joystick and robot were modelled according to the Denavit-Hartenberg principle, which allows direct relation between the position and orientation of the respective end effectors on the joystick and robot sides. Forward kinematic equations were then applied to the joystick; the desired position and orientation of the robot end effector were determined, and inverse kinematic equations were applied to these data to establish the robot's joint variables. This stage ended with the development of two operational modes: one where only the SSM motors are controlled in order for the slave to follow the master's movements, and the other where the PSM's motors are controlled separately. The simultaneous control of all robot motors could not be demonstrated due to fundamental mechanical flaws in the PSM and SSM designs.

Finally, testing was undertaken to demonstrate movement control of the robot by the joystick. The intuitiveness of the product was also tested successfully. The study ends with the presentation of the conclusions, the main conclusions being the successful development and testing of a joystick that controls the movement of a surgical robot, as well as the achievement of all main thesis objectives.

OPSOMMING

Hierdie tesis beskryf die proses vir die ontwikkeling van 'n gebruikerskoppelvlak vir 'n sewevryheidsgraad-, minimaal indringende chirurgiese robot. In die eerste twee hoofstadia van die algehele projek, voltooi deur ander studente, is 'n primêre slaafmanipuleerder (PSM) en 'n sekondêre slaafmanipuleerder (SSM) ontwikkel. Die stadium in hierdie tesis konsentreer op die skep van 'n stuurstok waarmee die gekombineerde beweging van die PSM en SSM beheer kan word.

Agtergrondinligting oor die gebied van robotiese chirurgie word verskaf, met spesifieke verwysing na die gebruikerskoppelvlakke van huidige stelsels, en die spesifikasies van die PSM en SSM word vasgestel. Daarna volg die beweegrede sowel as die belangrikste oogmerke van die projek. Die oogmerke is in die hoofafdelings van meganiese ontwerp, elektroniese ontwerp, beheerstelselontwerp en toetsing verdeel.

Die meganiese ontwerp van die stuurstok behels 'n konsepontwikkelingstadium, wat uitloop op 'n finale sewevryheidsgraad-ontwerp, wat dan op grond van ingenieurspesifikasies aangebied en beoordeel word. Aluminium word as boumateriaal gebruik; elektromagnetiese remme word vir elke koppeling gespesifiseer, en die finale samestel is 'n gekonstrueerde stuurstok wat aan alle vereistes voldoen.

Die elektroniese ontwerp behels die gebruik van magnetiese draaikodeerders om die stuurstok se posisie en oriëntasie te bepaal, sowel as meganismes met die nodige krag- en beheerstroombaanwerk om die stuurstok reg te laat funksioneer. 'n Koppelvlak tussen die PSM en die SSM moes suksesvolle kommunikasie tussen die meester en die slaaf bewerkstellig. Verskeie nodige aanpassings is dus aan die slaafstelsel aangebring, waarna die stuurstok en robot elektronies gekoppel is om 'n direkte reekskommunikasielyn te skep.

Vir beheerstelselontwerp is die stuurstok en robot volgens die Denavit-Hartenberg-beginsel gemodelleer, wat 'n direkte verhouding tussen die posisie en oriëntasie van die onderskeie eindpunt-effektors aan die stuurstok- en robotkant daarstel. Voorwaartse kinematiese vergelykings is daarna op die stuurstok toegepas; die gewenste posisie en oriëntasie van die robotiese eindpunt-effektor is bepaal, waarna terugwaartse kinematiese vergelykings op hierdie data toegepas is om die robot se koppelingveranderlikes te bepaal. Hierdie afdeling word afgesluit met die ontwikkeling van twee bedryfsmodusse: een waar slegs die SSM-motore beheer word sodat die slaaf die meester se bewegings kan navolg, en die ander waar die PSM se motore afsonderlik beheer word. Die gelyktydige beheer van al die robotmotore kon nie getoon word nie weens fundamentele meganiese tekortkominge in die PSM- en SSM-ontwerp.

Laastens is 'n toets uitgevoer om die bewegingsbeheer van die robot deur die stuurstok te toon. Die intuïtiwiteit van die produk is ook suksesvol getoets. Die studie sluit af met die projekgevolgtrekkings, waarvan die belangrikste die suksesvolle ontwikkeling en toetsing van 'n stuurstok is wat daarin slaag om die beweging van 'n chirurgiese robot te beheer, sowel as die verwesenliking van alle hoofprojekoogmerke.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

Entity	Description	Unit
Calculations		
E	Modulus of elasticity	GPa
I	Moment of inertia	mm^4
L	Moment arm length	m
M	Moment	Nm
P	Force	N
$P_{dissipated}$	Power dissipated	W
R_{thJA}	Thermal resistance	°C/W
$T_{holding}$	Holding torque	Nm
T_{rise}	Temperature rise	°C
ρ	Density	kg/m^3
v	Deflection	mm
Control aspects		
a	Link length	mm
A	Transformation matrix	-
d	Link offset	mm
M	Referring to the Master reference frame	-
o	Origin of a 3D reference frame	-
p	A point in a 3D reference frame	-
q	Joint variable	-
R	Rotation matrix	-
S	Referring to the Slave reference frame	-
T	Transformation matrix	-
α	Link twist	rad
θ	Joint angle	rad

ABBREVIATIONS

2D	Two Dimensional
3D	Three Dimensional
BERG	Biomedical Engineering Research Group
CCW	Counter Clockwise (measured w.r.t. motor output shaft)
CW	Clockwise (measured w.r.t. motor output shaft)
CAD	Computer Aided Drawing
DC	Direct Current
DH	Denavit-Hartenberg
DOF	Degree(s) of Freedom
GND	Ground (electrical)
GUI	Graphical User Interface
IDE	Integrated Development Environment
LED	Light-emitting Diode
LHS	Left Hand Side
MEMS	Micro-electrical Mechanical Systems
MIRS	Minimally Invasive Robotic Surgery
MIS	Minimally Invasive Surgery
P	Prismatic
PC	Personal Computer
PCB	Printed Circuit Board
PDF	Portable Document Format
PSM	Primary Slave Manipulator
PWM	Pulse Width Modulation
R	Revolute
RHS	Right Hand Side
RPM	Revolutions Per Minute
SPI	Serial Peripheral Interface
SSI	Serial Synchronous Interface
SSM	Secondary Slave Manipulator
SU	Stellenbosch University
USB	Universal Serial Bus
ZAR	South African Rand (monetary unit)

CHAPTER 1   INTRODUCTION

The process of minimally invasive surgery (MIS) allows surgeons to operate on patients without having to make the large incisions that are necessary with conventional surgical methods. During MIS, small incisions are made in the abdomen or thorax (depending on the type of surgery) and laparoscopic tools with which the surgery is to be performed are inserted through these incisions. The small incision length contributes to several positive factors: improved survival statistics, fewer post-surgical complications, shortening of the patient recovery period and a quicker return to normal life (Mack, 2001). According to Childress (2007), a typical wound from a traditional surgical incision may require a six-week recovery period; on the other hand, the recovery time for a laparoscopic hysterectomy is more or less two weeks. This greatly reduces the recovery (and therefore inactive) period for the patient while simultaneously decreasing hospitalisation time and thus costs. These advantages are the primary reasons why laparoscopic surgery is one of the most widely performed surgical procedures today.

With this prominent system at hand, technological advancements in this field led to the development of computer and robot-assisted surgical procedures during the 1980s. This has developed so well that currently, more than 1000 surgical robots are in regular clinical use worldwide and research and development is done at more than 100 universities (Dai, 2010). The use of surgical robots for laparoscopic surgery ensures enhanced dexterity, more degrees of freedom for tool movement, better visual feedback to the surgeon (by using cameras) and ultimately positive increases in all of the advantages provided by MIS. With the addition of motion scaling, the possibility of microscopic surgery is also introduced, which increases the number of procedures that would not have been viable with normal MIS (Childress, 2007).

Additionally, minimally invasive robotic surgery (MIRS) has the ability to reduce human error. Methods of motion scaling and tremor filtering are actively used in surgical robot systems to increase accuracy. Through programming the correct interface the surgical tool will have the ability to carry out the precise movements made by the surgeon at the master console, which effectively avoids the reverse-fulcrum-induced movements of normal MIS (Camarillo *et al.,* 2004). Further additions to the system are also enabled with the use of robotics. Advances in the area of micro-electrical mechanical systems (MEMS) point to the use of miniature sensors and actuators to enable haptic feedback in the robot. According to Camarillo *et al* (2004), high-fidelity force sensors can be used to improve force sensation beyond what the human hand can sense on its own.

As part of a surgical robot project for the Biomedical Engineering Research Group (BERG) at Stellenbosch University, Christiane (2008) developed a four degree of freedom (DOF) primary slave manipulator (PSM) that is responsible for manipulating its main surgical tool. In addition to that, Worst (2012) developed the secondary slave manipulator (SSM) that is responsible for controlling the primary manipulator as well as adding another three DOF to the system. This gives the robot a total of seven degrees of freedom and makes it comparable

1

with the seven DOF 'da Vinci' system from Intuitive Surgical (*Intuitive Surgical Inc.*, 2010). The purpose of this thesis is the design and construction of the user interface for the existing robot, essentially enabling the user to control the robot as the surgeon would the normal surgical tool. On overview of the thesis is given below.

In Chapter 2, the background information necessary for a full understanding of the project environment is described. Chapter 3 supplies the project definition, the scope and the main thesis objectives (together with the resulting engineering specifications) that the final design has to adhere to. In Chapter 4, the mechanical development section is presented. The important specifications are provided, the concept development process is discussed and the final design is given. Similarly, Chapter 5 describes the electrical and electronic design and development. In Chapter 6, interfacing of the PSM and SSM is discussed in terms of mechanical aspects, electronic aspects, communication and software. Chapter 7 and Chapter 8 present the mathematical modelling and control system development of the joystick and robot systems. The software that was created to support the control system design is discussed in Chapter 9.

In order to evaluate the final mechanical, electronic and control system designs – and to see if the final user interfaced satisfies the main thesis objectives – a testing section is presented in Chapter 10. Finally, the body of the thesis is concluded in Chapter 11, where the main conclusions and future recommendations are given.

CHAPTER 2   BACKGROUND INFORMATION

This section provides the background information relevant to the thesis. Firstly, a short overview of current MIRS systems is provided, concentrating on the different types of user interfaces for these systems. This is followed by a look at certain mathematical modelling conventions in robotics. Finally, this section contains a thorough description of the existing robot's primary and secondary manipulators, developed by Christiane (2008) and Worst (2012) respectively, and how they influence the research presented in this thesis.

2.1   Current MIRS systems

MIRS combines the advantageous non-invasiveness of MIS with the positive aspects of precision technology. Camarillo *et al.* (2004) lists the advantages of robot capabilities as being *"repeatability, stability and accuracy, tolerant of ionising radiation, [the use of] diverse sensors, optimised for [the] particular environment, spatial hand-eye transformations handled with ease, and [able to] manage multiple simultaneous tasks"*. On the other hand, although not compromising the effectiveness of a robot during surgery, the costs involved and the size of the surgical robot are definite drawbacks of MIRS. In this thesis, minimization of the costs plays an important role.

Currently, several MIRS systems are used at research facilities and in the commercial field. Christiane (2008) and Worst (2012) discussed the UCB/USCF RTW, HISAR, ARTEMIS and KaLAR systems and the commercially developed AESOP, ZEUS and da Vinci systems in detail. The da Vinci system is of particular importance as it is a widely used surgical robot in the field of laparoscopic surgery. The overview of MIRS will therefore be given with regards to the different aspects of the da Vinci system.

The system typically consists of the surgeon's console, the surgical arm cart and a high resolution 3D imaging system. Figure 1 shows the surgeon seated at the console on the left while the surgical arm cart is busy operating on a patient on the right. This also highlights another advantage of robotic surgery: the possibility of remote surgery or telesurgery, which enables the surgeon to be fully removed from the site of surgery.

The da Vinci system makes use of its patented EndoWrist design (Figure 2) to provide the seven DOF movement for the surgical tool. This, together with the robotic arms that manipulate the EndoWrist, corresponds to the combined primary and secondary manipulator systems pertaining to this thesis.

Figure 1: The da Vinci surgical system (*da Vinci system*, [S.a.])

Figure 2: The EndoWrist design with its 7 DOF indicated (*EndoWrist design*, [S.a.])

Figure 3: The surgeon's console as seen by the surgeon (*Surgeon's console*, [S.a.])

4

The movement of the surgical tool is fully controlled by the surgeon at the operator's console. This console, with its ergonomic design, provides a comfortable position for the surgeon during a surgical operation, which helps to keep the surgeon's position steady and therefore minimizes human error due to fatigue. While sitting at the console, the surgeon can observe the 3D, real time video image (often magnified up to 10-15 times) as if looking down on the physical operative site (Lobontiu & Loisance, 2007). This image is recorded by a 10 mm high-resolution 3D endoscope that is fixed to one of the robotic arms on the surgical cart, which allows the surgeon to control the position of the camera. Figure 3 displays the surgeon's 3D view.

Figure 3 also shows how the surgeon's hands fit into the main user interface controls. The surgeon's hand movements are picked up by the encoders contained in the user interface and are related directly to the movement of the robotic arms on the cart. Precision is enhanced by scaling down the motion of the master interface to the ultimate movement of the surgical tool by a factor of 3:1, as well as filtering out unwanted movements caused by tremors in the surgeon's hands (Lobontiu & Loisance, 2007). As can be deduced, the user interface should also allow seven DOF movement (along with the recording of movement in each direction) to enable the master console to translate its movement to the robotic arms.

The concept of haptic feedback can be introduced to the robot to further advance the accuracy of the surgical procedure. This implies that torque, force and tactile feedback are related back to the user's console, which allows the surgeon to experience the sense of touch as is common in conventional MIS. Van der Meijden & Schijven (2009) reported that their studies show *"benefits when adding force feedback to MIS devices and, moreover, indicate drawbacks when haptic feedback is absent"*. The problem with haptic feedback is that it immediately increases the complexity of the robotic system, as sensors have to be added at the tool end and actuators at the user interface end of the robot.

Literature also shows that different methods for master controller design exist in the surgical robot environment. The design of the user interface mainly depends on the specific attributes of the existing slave robot as well solution-specific requirements, as will also be seen in this thesis. Tavakoli *et al.* (2003), for example, developed a system incorporating an existing laparoscopic surgical tool in conjunction with the 6 DOF Phantom Premium haptic tool from SensAble Technologies (*SensAble Technologies*, [S.a.]), shown in Figure 4.

Because the surgical tool pivots around the point where it enters the patient through the incision (the trocar is the tool used to create and hold this incision), the conventional MIS surgeon has to make opposite movements to obtain the correct tool positioning. This phenomenon, known as the fulcrum effect, is often seen as a major drawback of MIS and is therefore normally absent in most MIRS systems.

Figure 4: The force reflective master-slave system of Tavakoli *et al.* (2003)

The example above, however, shows that designs are usually situation or project-dependent and not all the advantages of MIRS are always incorporated into the design.

The abovementioned haptic tool from SensAble Technologies is only one of many in their line of multi DOF user interfaces that *"provide precision positioning input and high fidelity force-feedback output"* (*SensAble Technologies*, [S.a.]). Another, the Phantom Desktop (shown in Figure 5), was used as the master controller by Queirós *et al.* (2010) in their design of a control system for robotic-assisted MIS. This emphasises the versatility of such devices in the MIRS environment.

Figure 5: The Phantom desktop haptic tool (*SensAble Technologies*, [S.a.])

Another attractive option in surgical robot master console design is to develop a user interface that has the same geometry (although scaled down in size) as the actual robot. This allows for less complicated position tracking and control system design, because the actuators of the robot would only have to execute the same (scaled down) movements as experienced by the respective links on the master system. The "Master and Slave Transluminal Endoscopic Robot" of Phee *et al.* (2009) illustrates how this is done in Figure 6 below.
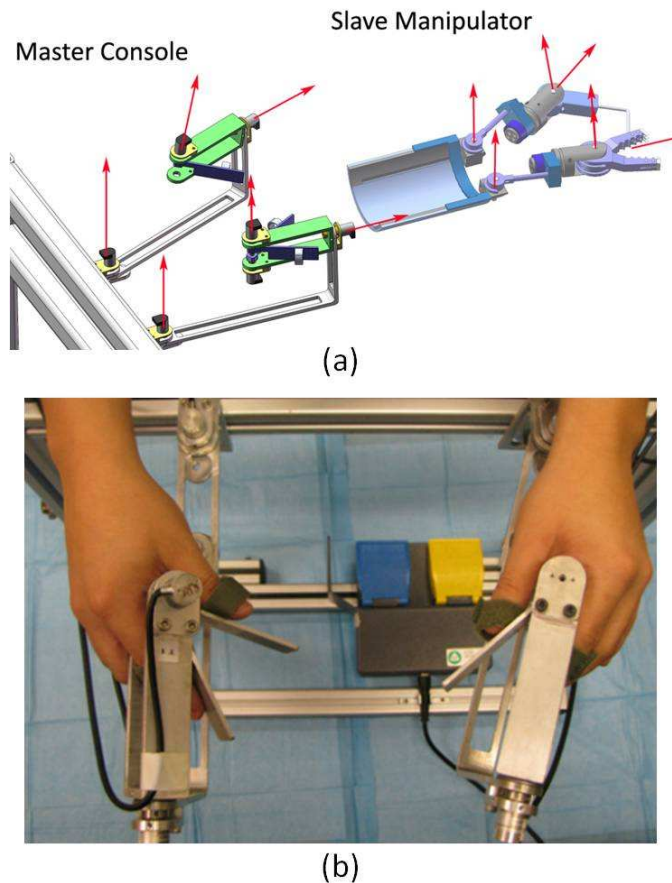
Figure 6: Master and Slave system of Phee *et al.* (2009)

Figure 6 (a) shows a diagram of the master console and the slave manipulator as well as their corresponding rotation axes and the clear similarities in geometry, while Figure 6 (b) shows the actual master console. Each movement at this console can be directly relayed, for each DOF, to the slave manipulator.

Although simplifying the physical and control system design processes greatly, the geometric relation between the master and slave is not seen as a necessity. One drawback is the fact that such a design restricts the master console to only being used for that specific project. If a design change is necessary at the slave end of the system, the master also has to be changed accordingly. The versatility of the interface is thus limited, which is why many current MIRS systems follow a more generic approach. The master console can be designed with the sole requirement of providing enough DOF in order for the surgeon to move the end-tool to the correct position and orientation necessary at the robot tool end. With encoder capabilities and the correct mathematical modelling, the desired position and orientation can be calculated and movement instructions can be sent to the actuators of the slave. Examples of systems with this type of user interface include Intuitive Surgical's da Vinci system (see Figure 3) and SOFIE – the first documented MIS, tele-operated, master-slave system with haptic feedback (Van den Bedem *et al.*, 2009).

To summarise, the three main user interface design types that emerged from previous authors are: designs incorporating currently used MIS tools together with haptic devices; interfaces that resemble the robotic manipulator (DOF to DOF) that it has to control; and versatile designs that do not resemble their slave systems but provide enough DOF to enable full position and orientation control. The inclusion of haptic feedback immediately increases the complexity of the system and is unlikely to be incorporated into systems where provisions have not already been made at the slave manipulator side, as is the case for this thesis. Also, the use of current MIS tools implies that the unwanted fulcrum effect will still be present. A user interface based on the design of the surgical robot is easy to control, but also limits the master-slave control system to the physical constraints of the specific robot. An independent design, providing enough versatility of movement while not constrained in any way to the robot's design, is therefore a probable solution.

2.2   Robot modelling background

The description of the robot's primary and secondary slave manipulators, as well as the modelling and control chapters later in the thesis, use specific notations and terminologies that require elaboration.

Spong *et al.* (2006) describe robot manipulators as comprising "of links connected by joints to form a kinematic chain". Joints can be either one of two main types: revolute (R) or prismatic (P). Revolute joints allow rotary motion where one link can be rotated relative to another, while prismatic joints are linear and allow relative linear motion between links. Figure 7 below explains these concepts while giving the correct 2D and 3D graphical notation that will be used in this thesis to indicate a joint.
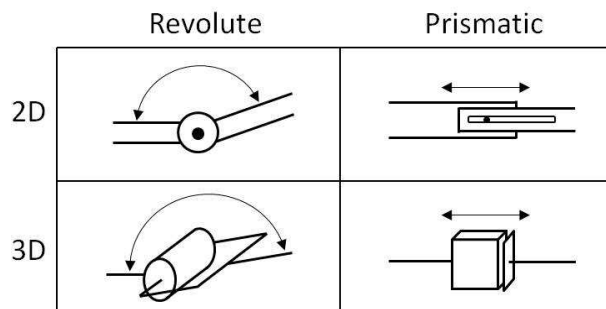
Figure 7: Notations for revolute and prismatic joints

Each joint in a kinematic chain corresponds to an additional DOF with its own axis of actuation. Most current industrial manipulators have six or fewer DOF, with a distinction being made between the first three joints of the manipulator, also known as the arm, and the rest (Spong *et al.*, 2006).

During operation, the arm usually determines the position of the end effector on the manipulator, while the other extra joints determine its orientation. In typical six DOF manipulators the joints between the arm and the end effector are referred to

as the wrist (Spong *et al.*, 2006), with the spherical wrist design being very common. The spherical wrist is particular in the fact that all three of its joint axes intersect at a common point, known as the wrist centre point. Figure 8 explains this phenomenon, which greatly simplifies determining the end effector position and orientation during kinematic analysis.
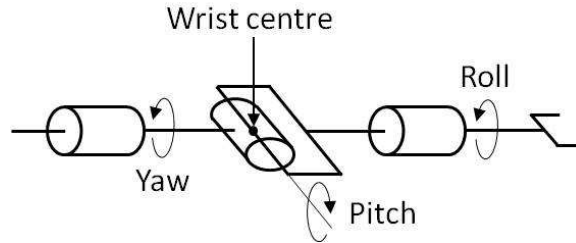
Figure 8: A spherical wrist, showing the angles and the wrist centre

2.3   The surgical robot

In order for the objectives and scope of this thesis to be clear, an overview of the existing parts of the surgical robot is necessary. The main parts of the surgical robot, or slave system, have already been developed and are known as the primary and secondary slave manipulators. They are discussed below in order to provide a basis for the further development of the master controller system.

2.3.1   The primary slave manipulator

The initial part of the overall project entailed creating a four DOF manipulator, developed by Christiane (2008), as shown in Figure 9 below. The PSM consists of a typical spherical wrist with three joints and an added gripper constituting the forth DOF. This final DOF does not contribute to the position or orientation of the robot, but is necessary as the surgeon's tool.

Also indicated in Figure 9 are the respective degrees of freedom provided by the different links and joints on the manipulator. These are summarised in Table 1.
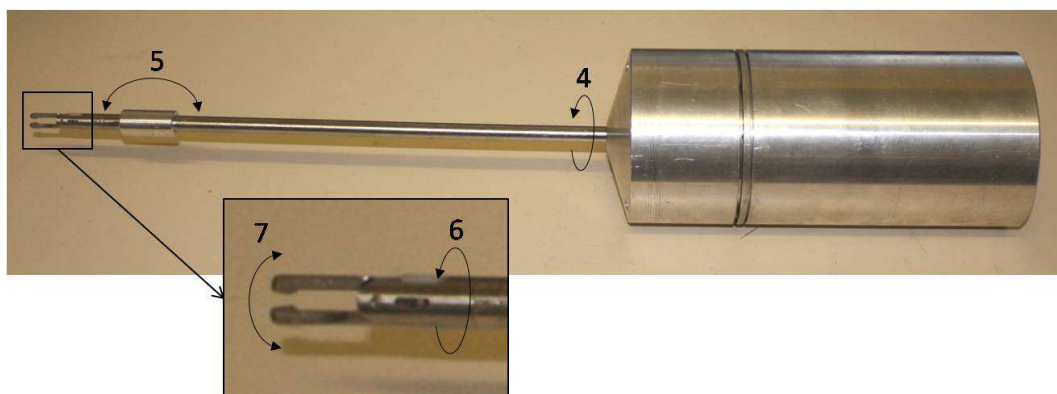
Figure 9: The PSM with its DOF indicated

9

Table 1: The degrees of freedom of the PSM

DOF	Description
4	The main PSM shaft rotates about its own longitudinal axis. 360° rotation is possible.
5	The distal part of the main PSM shaft beyond the elbow joint can rotate 55° from the extended positi on (where the distal part's axis is in line with the Joint 4 axis) to the extreme flexion position. The physical aspects of the design do not allow further rotation.
6	The front part of the shaft can rotate 90° about the distal shaft axis, due to physical design constraints.
7	The gripper constitutes the final joint. It can open and close.

Table 1 indicates that the PSM's degrees of freedom are numbered 4, 5, 6 and 7 respectively. This was done in order to conserve the convention that is followed throughout this thesis: the surgical robot's degrees of freedom start at 1 at the base motor of the SSM, after which they follow on each other in numerical order up to DOF 7, the gripper. DOF 1, 2 and 3 will therefore be discussed below in the SSM section.

The four joints of the PSM are actuated by five 12 V Faulhaber brushed DC motors (Christiane *et al.*, 2010), each with their own attached encoder and a 1526:1 planetary gear head. Stainless steel cables are reeled onto these motor output shafts in order to actuate the joints. Two motors are used together as a kind of pulley system to actuate the elbow joint (Joint 5), one motor is used in conjunction with a torsion spring to actuate Joint 6, one motor is used together with an axial spring for Joint 7 and the final motor is fitted with an 28 tooth spur gear that turns a 84 tooth internal spur gear and thereby actuates Joint 4.

The PSM also contains a printed circuit board (PCB) with the correct control and power circuitry (mainly one L6225 Full H-Bridge motor controller chip per motor) to control the motors and capture the encoder data. The motors and their attachments, the gears, the cable reels and the PCB are all located in the cylindrical steel housing (shown in Figure 9) while the main shaft guides the cables to the joints.

2.3.2 The secondary slave manipulator

The SSM was designed and constructed by Worst (2012) during the second phase of the overall project. It is shown in Figure 10, with the respective degrees of freedom indicated at each joint. Table 2 gives a list of the SSM's joints and how they operate. It is also important to notice that the SSM was designed in such a way that the actuation axes of Joints 1, 2 and 3 all intersect at one point (indicated in Figure 10). When the PSM is fitted to the main assembly, the axes of Joints 3 and 4 coincide, which results in the main tool shaft passing through the said intersection point. This was intentionally designed so as to provide an entrance point for the surgical robot into the trocar in the patient's body. Once the

robot is in the correct position, any actuation of Joints 1, 2, 3 and 4 would cause the end effector to move but the physical position of the tool insertion point would be unaltered and the patient will be unharmed. Physical constraints, and not only mathematical modelling and control, therefore help to provide the patient with a safe procedure.
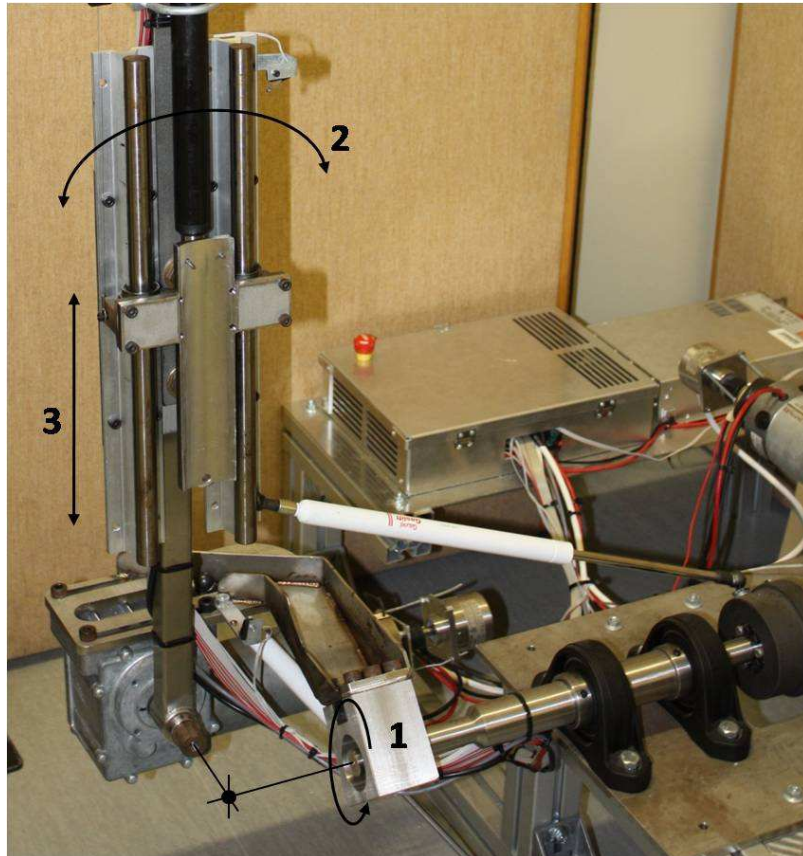
Figure 10: The SSM with its DOF indicated

Table 2: The degrees of freedom of the SSM

DOF	Description
1	The base joint rotates about a horizontal axis. When looking at the robot from the front right position, and regarding the second link of the robot in the upright position (as indicated in Figure 10), actuation of Joint 1 can result in a 11° CCW rotation and a 30° CW rotation due to physical constraints of the design.
2	This joint rotates about an axis perpendicular to DOF 1. When regarding the robot from the front and right hand side, again considering the second link in the upright position, actuation of this joint can result in a 11° CCW rotation and a 24° CW rotation, due to physical constraints of the design.
3	This is the only prismatic joint in the whole kinematic chain. The linear motor is attached to the second link (used as the reference above) and allows a 283 mm stroke length from the topmost position.

Two 24 V Parvalux PM95GWS Brushless DC motors with worm-spur combination gearing (154:1 ratio), each fitted with a failsafe brake and a Hengstler RI59-3600 rotary encoder, are used for Joint 1 and 2 respectively (Worst, 2012). The linear movement of Joint 3 is provided by a Linak LA30 self-locking linear actuator fitted with a UniMeasure JX-EP-20 linear encoder.

The control circuitry for the SSM consists of two complex OSMC H-bridge controller boards for the Parvalux motors (due to the fact that they require high levels of electrical current to operate), L6225 controller circuits for the linear motor and brakes, circuitry to capture the encoder data and communicate with the PSM and an Arduino Mega 2560 development board used as the main controller of the robot (Worst, 2012). This, together with the 24 V - 3 kW power supply, was fixed to the base frame onto which the robot was fastened. For this thesis, however, this system was improved to provide better interfacing, which is discussed further in later chapters.

When the PSM and SSM are assembled, the system is as shown in Figure 11 below. This is the surgical robot for which a user interface is to be created.
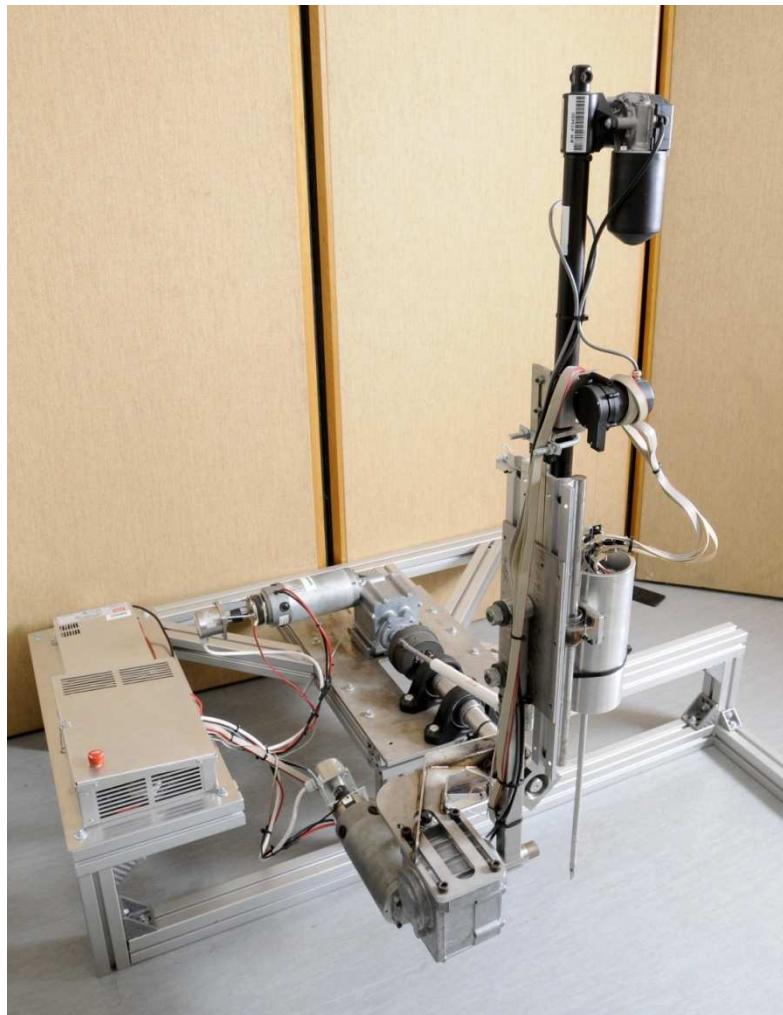
Figure 11: The assembled surgical robot

CHAPTER 3   PROJECT DEFINITION

In this chapter the origin of the project is stated. The original client requirements are discussed, along with all of the deduced objectives and the changes that occurred during the course of the project. These changes culminated in a list of main thesis objectives. A list of engineering specifications, derived from the objectives, is also given.

3.1   Project origin and motivation

The project originally started in 2006 as a collaboration between two brothers, Dr. Almero Viljoen (a medical doctor) and Mr. Jacob Viljoen (an engineer), and the Biomedical Engineering Research Group (BERG) at the Stellenbosch University Department of Mechanical and Mechatronic Engineering. At that stage the surgical robot environment was open for new projects and designs and studies into the motivation for such a project delivered positive results. The main motivational points for the use of MIRS were stated in Chapter 1 and they support the feasibility of this thesis.

Another important factor in the commercial viability of surgical robots is the costs involved. According to the latest available information in literature, the da Vinci robot's latest version sells for $1.5 million (Yash, 2008). New systems that are being developed attempt to minimize costs in order to get a competitive edge. The advantage of this thesis is that the costs are greatly reduced in comparison with current commercial systems. The expenses up to date for the overall project, excluding those for engineering and labour time, have totalled R125000 (Worst, 2012) and Appendix F shows that this stage has also kept the added costs minimal.

3.2   Initial requirements and adjustments

It was originally decided that the overall project should be divided into three main subprojects: the PSM, the SSM and the user interface. Objectives for the first two project phases were discussed in the theses of Christiane (2008) and Worst (2012) and will not be repeated here as this thesis concentrates on the development of the user interface.

The clients' original idea for the master console was that the surgeon hands should be fitted with an encoder system that allows movement of the hands in a virtual environment that tracks the position and orientation of the fingers. The surgeon would not hold on to anything tangible, but the hand and fingers would represent the surgical tool and their position and orientation would be relayed to the surgical robot. An example of such a system was developed by Li *et al.* (2011) at the Nanyang Technological University in Singapore and is shown in Figure 12. Optical linear encoders are placed on different parts of the hand and they sense the movement of different finger joints relative to each other.

Alternatively, the same type of design can incorporate accelerometers to the same end.

Figure 12: The finger motion capturing device of Li *et al.* (2011)

This concept was considered, but certain inherent problems were evident. Firstly, the absence of anything tangible is not necessarily an advantage. In normal MIS the surgeon receives tactile and force feedback when he/she operates with the MIS tools on a patient. This helps the surgeon to decide if the correct amount of force is applied in the appropriate direction and increases the accuracy of the procedure. With its absence, and only visual feedback by means of a camera available to help guide the surgeon, this method of master control is bound to have a decrease in accuracy as well as not being extremely intuitive. Another potential problem is the fact that no stationary reference point is available in the design as the hand is moving continuously. This increases the difficulty of modelling and control system design for this device. A mechanical system, on the other hand, can have a reference point at its stationary base, which immediately simplifies control system design. These points led to the decision that a mechanical joystick type master controller would be a better solution for the user interface.

Another initial possibility was the inclusion of haptic feedback into the user interface. As explained in the previous chapter, haptic feedback allows torque, force and tactile feedback to be related back to the user's console, which allows the surgeon to experience the sense of touch as is common in conventional MIS. This concept only became a real possibility after evidence of precedent was found in literature, of which SOFIE at the *Technische Universiteit Eindhoven* is an example (Van den Bedem *et al.*, 2009). Unfortunately, the PSM development stage of the project was already completed by that time and the use of haptics could not be incorporated into its design. Force sensors at each joint of the PSM and the SSM, as well as on the tool tip, would be necessary to capture force and tactile feedback; this would in turn result in redesigning the PSM and an increase in development time. Another option for the inclusion of haptic feedback would be

based on a position/position architecture, where the position of the robot end effector is tracked and compared with the input reference position, after which force feedback is supplied the operator based on the error. This would not require force sensors to be included on the slave system. But, due to the high gear ratio on several joints of the slave side (refer to Chapter 6), the slave system is not back-drivable and haptic feedback necessitates the inclusion of sensors.

Due to these reasons, and because of the fact that creating a user interface is the main objective of this stage of the overall project, haptic feedback was excluded as a requirement for this thesis. This decision again supports the choice of a mechanical user interface above the glove-type interface in Figure 12: if haptic feedback is excluded as required criteria, a joystick-type user interface would still a better solution because its encoder implementation and finding a zero position will be much simpler.

With further consideration of the now decided requirement of creating a mechanical joystick type user interface, other concerns came to light. Mostly, restrictions in the versatility and manoeuvrability of the PSM and SSM led to extra constraints being put on the design of the master device (refer to Chapter 6 for a detailed discussion of these aspects). The physical constraints of the robot then limit how well the master can control the slave, but not because of an ineffective control system design at the master end. This then has an effect on how the objectives for the user interface are evaluated. Essentially, slightly alternative control system design from the traditional master-slave control setup as explained by Spong *et al.* (2006), especially in terms of robot velocity control, has to be incorporated.

3.3   Main objectives

With all of the above requirements and adjustments in mind, the set of main objectives that guide the thesis and determine its scope could be derived. The most important objective is to develop a user interface for the seven DOF minimally invasive surgical robot consisting of the PSM and SSM. As is evident, this main problem can be divided into several subsections: a mechanical joystick system has to be created, an encoder system used for master position monitoring must be incorporated, an electronic system regulating the above two systems is necessary, communication between the master and the slave is essential and control of the slave by the master system would be a determining outcome. These subsections essentially provide the main objectives as they relate to different phases of the thesis and they are provided in the list below:

1. Obtain a full understanding of the field of robotic surgery, specifically with regards to user interface systems, by means of researching current surgical robots and their specific designs. Studying the operation of the PSM and SSM and how their designs affect this thesis is also important for this objective. This section is fully covered in Chapter 2.
2. Design and construct a mechanical user interface system (the joystick) that the surgeon can use to control the movement of the seven DOF surgical tool.

3. Design (and implement into the mechanical design) the encoder system that will be used to track the movement of the surgeon's hands.
4. Design and implement the necessary electronics that will allow the mechanical, electronic and encoder components of the joystick to work together as one system.
5. Implement an effective communication system between the master console and the collective PSM and SSM system.
6. Design the control system that will regulate how the joystick is able to control the surgical tool.
7. Design and conduct experiments to test the working of the control system and the accuracy with which the robot's movements are controlled.
8. Write a full technical report, supported by any necessary extra documents, on all of the above sections and the findings of the thesis.

It is important to note that, although it is not one of the main listed objectives, costs should be kept low to make the product feasible. Listed in the motivation for the existence of this project is the fact that lower cost surgical robots are necessary and costs have to be minimised if this outcome is to remain applicable.

3.4   Engineering specifications

As a result of the absence of client requirements and other specifications, a list of engineering specifications was not originally available. Instead, these entities had to be determined throughout the course of the different design sections. Although they are discussed in more detail in their appropriate chapters, the list below shows the final set of engineering specifications that the end-product had to satisfy:

1. The joystick should be designed to be controlled with only one hand.
2. The user interface should provide an unhindered movement space of 30 cm x 30 cm x 30 cm, allowing a scaling factor of 3:1.
3. The encoders should be small and lightweight, preferably custom solutions rather than bulky commercial encoders.
4. The combined encoder resolution of the user interface should be 9 mm in order to satisfy the surgical robot resolution requirement.
5. The combined encoder resolution of the user interface should be 0.9 mm in order to satisfy the suture resolution requirement.
6. The design should allow a 'safe space' volume of 10 cm x 10 cm x 10 cm for the movement of the robot end effector.
7. The joystick's base frame should not experience more than 1 mm deflection and the combined link deflection of the kinematic chain should not be more than 0.6 mm.
8. It must be possible to actuate the joystick joints separately from each other, while keeping the applicable joints stationary. This is to enable direct 'DOF to DOF' control of the PSM motors.
9. A safety switch should be incorporated with the following requirements: when pressed, the joystick should be able to control the robot's

movement; when released, the robot should not execute any movement occurring at the joystick side.

10. Minimal mass should be carried by the operator while operating the joystick.

11. The movement of the joystick's end effector should directly be translated to the movement of the robot end effector, taking the scaling factor into account.

12. Detailed knowledge of how the joystick is designed and how it functions should not be a prerequisite to being able to operate it.

CHAPTER 4   MECHANICAL DEVELOPMENT

This section describes the mechanical aspects of creating a mechanical joystick type user interface. Firstly, the main thesis objectives stated in the previous chapter are converted into a set of engineering specifications. After this the concept development stage is described: different concepts were generated and evaluated in order to choose the final concept. The detailed computer aided design (CAD) executed on this final concept is presented and the joystick assembly is discussed.

4.1   Mechanical specifications

The objectives described in Chapter 3 are mostly qualitative in nature. In order to have more specific guidelines for the thesis, a set of quantitative engineering specifications had to be created from the objectives. These parameters can then guide the design process more closely and any generated concept should adhere to them.

The second and third objectives, those that directly relate to the mechanical aspects of this thesis, state the necessity to create a joystick that the surgeon can use to control the movement of the seven DOF surgical robot and that encoders should be implemented to track the position of this joystick. From this the specifications can be divided into different applicable sections: movement, the surgical robot and encoders. The concepts of safety and intuitiveness are also vital and specifications for these sections are also necessary.

4.1.1   Movement

The movement of the surgeon's hand at the joystick should be captured and then translated into movement of the surgical tool at the robot end. If the robot is capable of seven DOF movement, the joystick should provide enough versatility in order for every desired position and orientation to be possible at the robot end effector in a specified operating space. Also, enough space should be available for unhindered movement. Literature suggests that the downscaling factor of 3:1 used by the da Vinci robot yields positive results (Lobontiu & Loisance, 2007)[1]. For the purposes of this thesis, and due to the physical constraints of the SSM, the operating space of the surgical tool in MIS was assumed to be a cube of size 10 cm x 10 cm x 10 cm. When taking the scaling factor into account, the unhindered movement space at the joystick should then be approximately 30 cm x 30 cm x 30 cm.

To specify the accuracy of the robot, one must first consider the eventual setup. The accuracy of conventional robots might be tested by giving the robot a position to reach and measuring how well this instruction is executed. But in this case, with visual feedback and the operator being able to correct a faulty instruction, the measured accuracy is not such an important factor. Rather, the resolution of movement plays a bigger role. This can be understood in terms of

---

[1]   During the testing phase, several other factors higher and lower than 3:1 (between approximately 2:1 and 4:1) were tested and it was noted that the best response was still evident at 3:1. Both higher and lower ratios caused the robot movement to become non-intuitive.

18

how small a movement is possible at the robot end effector, a factor which is highly dependable on the motors specified for the SSM and PSM systems and which the joystick system would not be able to increase. Worst (2012) documented the SSM resolution results (measured at the PSM's end effector) as 0.7 ± 0.2 mm, 0.5 ± 0.2 mm and 0.2 ± 0.2 mm for the Joint 1, 2 and 3 motors respectively, while the resolution of the PSM was undocumented by Christiane (2008). Due to the absence of useful information, the combined resolution of the surgical robot was assumed to be about 3 mm. This is an optimistic assumption when taking into account the problems with the robot's rigidity (see Chapter 6). If the scaling factor of 3:1 is taken into account, a minimum resolution of 9 mm would be necessary at the joystick's end effector.

On the other hand, the required resolution of the joystick can also be determined according to the eventual sutures that the master-slave system would have to apply. If a typical suture is assumed to be 3 mm from end to end, with 10 intermediate positions providing sufficient resolution at this scale, the required resolution would be 0.3 mm at the robot and consequently 0.9 mm at the joystick end. This requirement was added, not as a guiding specification, but as a comparative measure. As the main objective is to demonstrate control of the existing slave system by the joystick and not to perform suturing, the decision was made to conform to the surgical robot resolution requirement. Where applicable, data for the suture requirement will still be shown to provide ample information for comparisons.

4.1.2　　　　The surgical robot

The physical constraints of the PSM and SSM, as explained in Table 1 and Table 2 respectively, should also be considered. The robot may not extend past the specified ranges for each degree of freedom. The user interface should provide measures for holding the robot within these ranges.

4.1.3　　　　Encoders

The encoders provide the necessary information to track the movement of each link on the joystick. An encoder will therefore have to be placed at each joint of the joystick. In order to keep the weight and size limits to a minimum and to minimize interference with the joystick's movement, small encoders are necessary.

For the surgical robot resolution requirement, the required resolution of the combined joystick joints is 9 mm. If it is conservatively assumed that the kinematic chain consists of six joints (each of which can contribute an additional positional error in the worst case scenario of same-directional rotation axes) and that it is 240 mm in length from the first link to the end effector, the combined required resolution in radians is 0.0375. This implies a required angle resolution of 0.00625 radians per encoder, converting to 0.358° or 1005 positions per revolution. If the suture resolution requirement, which is 0.9 mm, were to be

satisfied, the required resolution would be 10 times higher at 10050 positions per revolution.

4.1.4 Safety

This is a very important aspect of the user interface as, in this particular application, failure to adhere to safety regulations can have detrimental results. Firstly, the movement of the robot should be limited, not only with regards to staying within the physical constraints of its links, but also by limiting the reach and velocity of the end effector in the patient. A certain 'safe space', within which the surgical tool is allowed to operate, should be created. This volume should correspond to the 10 cm x 10 cm x 10 cm cube that was specified in Section 4.1.1 above.

Another probable necessity in terms of safety would be the ability of the robot and/or the joystick to hold its position at times when the surgeon does not want the joystick's movements to be carried out at the robot end. An obvious example of such a situation is when the operator needs to let go of the controls in order to rest his/her hand or arm or for any other probable reason. A further extension of this requirement may be the need for an emergency stop function, as is already incorporated in the PSM-SSM design (see Chapter 6).

During the literature review, several medical safety regulations for the robotic surgery environment were noted. It was however decided that they would not contribute to the required specifications of the joystick, as this thesis concentrates on the functional aspects of controlling the robot with its user interface. Further iterations of the design would have to take these regulations into account.

4.1.5 Intuitiveness

An important consideration is how intuitive the joystick is to the surgeon. As this is not necessarily a quantifiable concept, several qualitative requirements are applicable here. The idea is that holding the joystick should give the surgeon the feeling of actually having a hand inside the surgical area and using his own fingers as he would the surgical tool.

An intuitive joystick implies that it is easy to use. Minimal time should be spent on trying to teach a surgeon how to use it, which implies a clear and simple design. Also, in depth knowledge of the joystick's or robot's working principles should not be a prerequisite to being able to operate the joystick.

On the mechanical side, the joystick must enable the surgeon to change the position and orientation of the robotic tool easily and the gripper should be actuated by a gripping action of the surgeon's fingers. Each joint should also have the ability to be actuated separately. The joystick should allow smooth and unhindered movement and should not put unnecessary strain on the surgeon's

hand. This necessitates the use of a light-weight, although still durable, material for link construction.

4.2   Concept development

This section is concerned with the stage where different joystick concepts were generated before the final design was chosen. It follows a very iterative process where each concept was evaluated against the engineering specifications, the main problems were identified and the gained information was used to help improve the next concept by excluding evident design errors. This stage underwent five phases, representing five main concepts, and resulted in a final concept that satisfied all of the applicable engineering specifications.

Detailed discussions pertaining to the development of the five joystick concepts are contained in Appendix A. The design decisions that were made during the applicable phases and how these decisions led up to the final design concept are also discussed in Appendix A.

4.3   Detailed joystick design

The final concept's detail design is shown below in Figure 13.
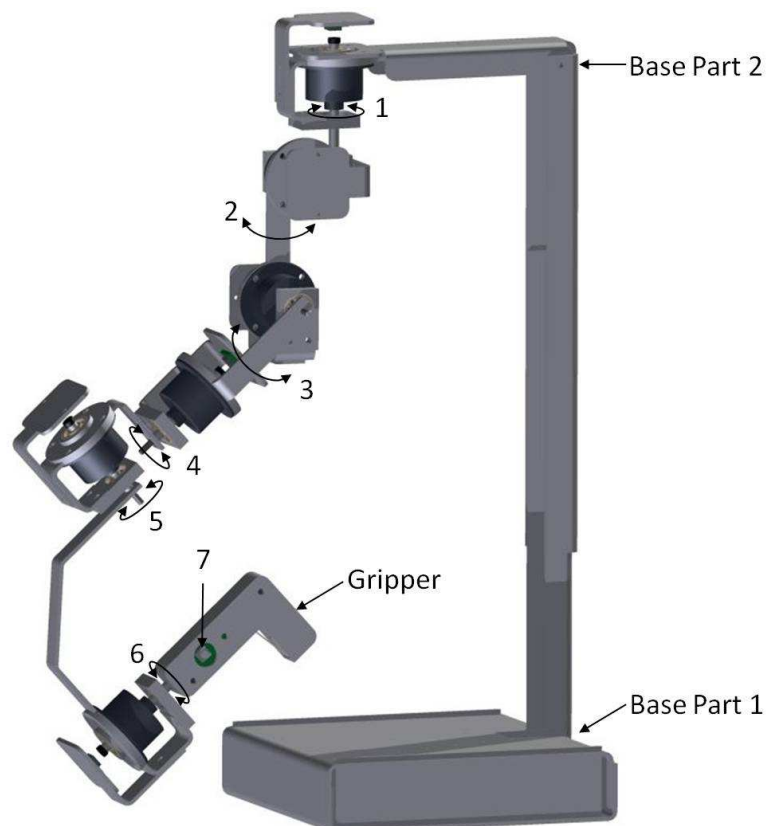
Figure 13: The final detailed joystick design

The two base parts, the gripper and each joint showing each DOF are indicated. In this section, the different aspects of this final design are explained in more detail. These aspects refer primarily to the base frame design, joint design, brake specification, encoder inclusion, gripper design, material specification and design evaluation.

4.3.1   Base frame design

As one can see from Figure 13, the base frame adheres to the description of Concept 5 (refer to Appendix A) in that the kinematic chain can hang down and no torque limiters are necessary. It consists of two parts: a hollow square sheet metal base with an extension on one side extending 25 cm upward; and an L-shaped part fitting over the base extension and serving as the base-surface to which the first joint is fixed. The size of the square part is 150 mm x 150 mm x 20 mm and the assembled base is 400 mm high, although this height is adjustable. The extensions on both parts of the base contain several holes that line up together. By sliding the two base parts over each other and fixing it with bolts and nuts in the appropriate holes, the height of the base can be adjusted.

A U-shape profile was used for the extensions of the base parts to create a strong frame that will be able to hold the mass of all the joints. Supporting calculations to show that the base frame is strong enough to hold the joystick joints can be seen in Appendix C.

4.3.2   Joint design

The layout of the joint components is shown in Figure 14.
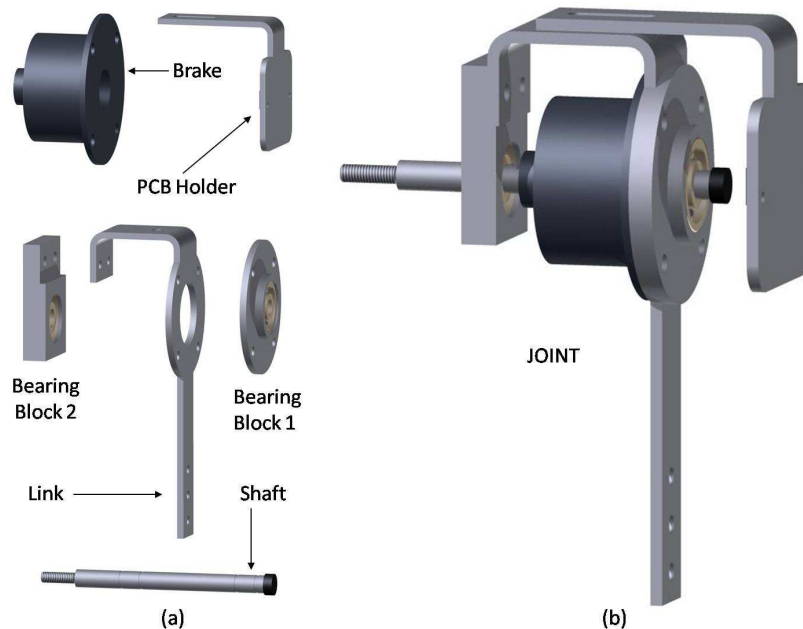
Figure 14: The joystick joint design

The fact that all joints were revolute simplified this part of the design. It is evident that the joints and links differ in length, orientation and how they are fixed, but the main parts of each joint are the same: a sheet metal frame found at the end of a link, two bearing housings, two bearings and a shaft to which the next link in the chain is fixed. The encoder and brake are also part of this setup, but they will be specified shortly.

In Figure 14 (b) the assembled joint of DOF 1 can be seen, while Figure 14 (a) shows the different parts of the joint. These parts are standard to each joint, although their dimensions may differ. The link is the base part of the assembly and bearing blocks 1 and 2 are bolted to the link, with the bearings inserted into the housings using a very slight transition fit. The circular parts of the link and bearing block 1, as well as the hole positions on these parts, were designed so that they agree with the dimensions on the base of the brake, thus simplifying the assembly procedure. A simple design, with a shoulder to make it fit tightly to the end of the link, was used at bearing block 2.

The bearings and the shaft are the main rotating components. A basic bearing requiring little or no maintenance, that operates at normal temperature and low speeds and is lightweight, was necessary. Several options were considered and it was decided that the Xiros® B180 plastic deep grooved radial ball bearing (with a 5 mm inner diameter) from Igus (*Igus*, [S.a.]) would be used. The bearing is lubrication and maintenance free, nonmagnetic and washable, corrosion-resistant and lightweight, which are ideal specifications for this application.

The inner diameter of the bearing was chosen to fit the external diameter of the shaft. The shaft is kept axially stationary by using four circlips, one on each side of the two bearings, to fix the shaft to the bearings. For this reason, four thin grooves were machined into the shaft at the appropriate places. The shoulders of the bearing housings were designed to oppose each other, meaning that both bearings would be inserted into their housings from the outer side of the U-shaped link. This factor, together with the circlips fixing the bearings to the shaft, keeps the shaft from shifting when an axial force is experienced.

Another aspect of the shaft is the threaded part (size M3) at the one end. This allows the shaft to be fixed to the next link in the kinematic chain. Similarly, the purpose of the holes in the long end of the link is to connect it to the previous joint shaft in the chain. The other parts indicated in Figure 14, the brake and PCB holder, will be specified and explained below.

4.3.3   Brake specification

As a safety precaution, six brakes (for Joints 1 to 6) are included in the final design. They can be used to stop movement on any joint (except for the gripper joint) at any point in time, whether it is necessary for safety reasons, for the specific surgical procedure, or just to allow the surgeon to keep the specific position and orientation of the joystick.

The required specifications for these brakes were very similar to those of the bearings: lightweight, easy to use, little or no maintenance, small parts and easy to incorporate into the design. The inner diameter would also have to agree with the shaft dimensions, as this value was already specified for the shaft and the bearings.

The most logical solution was an electrically actuated magnetic brake, as it only requires a voltage to be applied for the unit to exert a braking force. The Miki-Pulley 112-02-11 electromagnetic micro brake (*Miki-Pulley*, [S.a.]) was chosen for its simple lightweight design, ease of use, simple assembly procedure and 5 mm inner diameter. It is shown in Figure 15.

Figure 15: The Miki-Pulley electromagnetic brake

These brakes consist of two parts: the bigger flanged stator unit and the thin rotor unit. The flanged base is bolted to the circular part of the link (see Figure 14) while the rotor unit is fixed to the shaft with a grub screw at a short distance (specified in the 112-02-11 model's datasheet) from the stator. When a voltage of 24 V is applied to the stator a magnetic field is created and the magnet in the rotor snaps shut against the stator. This effectively locks the two parts together and stops any movement of the joint. The datasheet specifies the brake's torque to be 0.4 Nm (*Miki-Pulley*, [S.a.]), which was calculated in Appendix C to be adequate for the purposes of this thesis.

4.3.4   Encoder specification

Different encoder options were considered and researched, with the two most prominent options being normal rotary encoders (optical or magnetic) and magnetic positioning systems. Magnetic positioning systems, like the FASTRAK from Polhemus (*Polhemus*, [S.a.]) and the trakSTAR from Ascension (*Ascension*, [S.a.]) use a base transmitter to create a magnetic field within which a sensor can move. Six DOF tracking (i.e. the position coordinates and the orientation) of the sensor is then possible.

The ease of use of these magnetic systems provides for a very attractive solution to the problem of six DOF position and orientation tracking of the joystick. Only one system would be necessary and no separate encoders would have to be placed on each joint. On the other hand, the magnetic field transmitter is a big and bulky piece of equipment and the price of these products ranges from R50000 to R60000 (*Fastrak*, [S.a.]). These drawbacks, especially the cost of the

system, were in conflict with the desired outcomes of this thesis and therefore the magnetic tracking option was discarded.

The rotary encoder option was therefore chosen. Many commercial, ready-made encoder units that could be mounted directly onto an output shaft and had standard output wires were available, most using either optical or magnetic sensing technology. These were acceptable solutions with a wide range of good resolution capabilities, but they had drawbacks: firstly, the enclosed unit which represents a commercial encoder would make for a bulky design when fitted to the joint shown in Figure 14, also adding weight to the system and stressing the surgeon's hands; and secondly, commercial systems were expensive. Six or seven of these units would also amount to about the same cost as the magnetic positioning systems mentioned above.

This motivated the use of a custom designed encoder system. For this purpose, the AS5040 10 bit programmable magnetic rotary encoder from Austria Microsystems (*AS5040*, [S.a.]) was chosen (Figure 16).
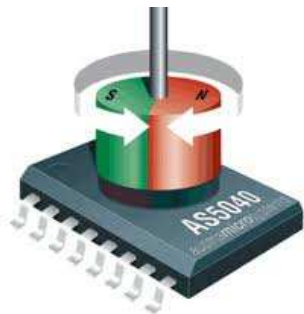
Figure 16: A representation of the AS5040 chip and magnet

The AS5040, fitted to a PCB with its controlling circuitry, is kept stationary while a specified magnet, connected to the shaft end, rotates above it. The shift in magnetic field, brought about by the rotation of the two poles on the magnet, is sensed by the component and then related back to a change in radial position. The AS5040 allows 10 bit resolution, effectively outputting 1024 positions (in the form of pulses) per revolution (*AS5040*, [S.a.]). These components and the accompanying magnets are very cheap and easy to acquire and the encoder unit can be designed to fit the specific requirements of the joystick.

In Figure 14 one can see the black, two-pole magnet mounted on the end of the shaft at the bearing block 1 end. Then the indicated PCB holder part is used to hold the PCB with the AS5040 stationary. The encoder PCB (refer to Chapter 5 for this design) is mounted and bolted onto the inner flat side of the holder, with the encoder component centred on the shaft axis. The PCB holder is then bolted to the link on top (the bottom of the U-shape), using the slot in the PCB holder to shift its position and get the required air gap between the AS5040 and the magnet, as specified by the datasheet (*AS5040*, [S.a.]).

The choice of an encoder with 10 bit resolution (giving 1024 positions per revolution) demonstrates that the surgical robot resolution requirement of 9 mm (which converts to 1005 positions per revolution) is satisfied.

4.3.5   Gripper design

The gripper design from Concept 3 (see Figure 47, Appendix A) proposed the use of two plates controlled by the operator's thumb and index finger. These plates could be opened and closed, resembling the motion of the gripper at the surgical robot end. This would then require the design of a communal pivoting axis for the two plates, as well as incorporating an encoder to record the pivoting action. Detail design showed that it would be a very intricate and difficult job on a small scale. Considering the absence of haptic feedback (essentially creating only an open and closed state for the gripper), it is evident that a simpler solution could be used.

The exclusion of a pivoting axis, and therefore an encoder, for DOF 7 meant that a simple two-state button system could be incorporated. An electronic solution, where a button could be pressed to actuate the gripper, was incorporated. To keep the feeling of actuating the gripper mechanically, the design in Figure 17 was used.
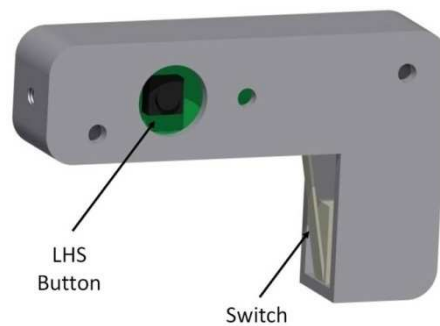
Figure 17: The gripper design

The part is designed to have an L-shaped body that fits into the palm of the operator's hand. The thumb and index finger can then push the buttons (one on each side of the part) to actuate the gripper. The gripper is machined out of two aluminium blocks, each resulting part being one side of the gripper. Space is created on the inside where the PCBs containing the buttons can be fixed.

Two buttons were used for two reasons: firstly to keep the part intuitive by resembling the action of pressing a gripper closed and secondly for extra safety reasons – both buttons have to be pressed or released in order for the gripper to be actuated.

As another safety precaution, a switch (indicated in Figure 17 above) was added to the system. The purpose of this switch is to act as a so-called "dead man switch" which implies that it will have to be activated in order for the joystick to

26

function correctly and send instructions to the robot. The detail aspects of its functioning will be covered in Chapter 8.

4.3.6   Material specification

The choice of material for the different parts of the joystick was a relatively simple task. The starting options were plastic (produced by rapid prototyping) or metal. The rapid prototyping option was discarded because of the costs involved and the fact that it was not the optimal choice for a first design iteration. Then there was mainly a choice between aluminium and stainless steel sheet metal, as the designed parts were all made of 2 mm thick material. The advantages of stainless steel were that it is a strong (although heavy) type of steel that would not be subject to strain on such a small scale, while the aluminium would be a lightweight solution.

Therefore, a trade-off had to be made between weight and strength. To facilitate this decision, calculations were made to determine the strain experienced by the end effector position on the joystick based on the two different materials' properties. These calculations (refer to Appendix C) proved that aluminium would still be strong enough to result in minimal strain. Aluminium sheet metal, 2 mm thick, were then used as the manufacturing material for all the laser cut and bended parts. Solid aluminium blocks were used to machine the gripper parts and the bearing housings.

4.3.7   Design evaluation

To confirm that the final design was indeed the best solution that resulted from the concept generation, this design had to be evaluated according to the engineering specifications summarised in Chapter 3. The basic evaluation and comparison of all concepts against the specifications are contained in Appendix A. This resulted in Concept 5 being chosen as the final concept. Table 3 below contains a more detailed summary of how the final design performs in each required specification section.

Table 3: The final design evaluation

Specification	Final design performance
Movement	The movement of the final design is considered the best of all the concepts. The all-revolute design provides unhindered movement, while the lightweight aluminium keeps weight off the surgeon's hands. The use of the same "Joint 4 to 6"-design from Concept 3 enables good movement and orientation tracking and simplifies controlling the spherical wrist on the robot, while in no way constraining the joystick operation to the particular design of the robot. From the parts and assembly drawings, it can be seen that the joystick is operational within a 3D square shaped volume of 30 cm x 30 cm x 30 cm.
Surgical robot	The final design is not constrained to the design of the PSM and SSM, which broadens its use to possible next design iterations of the robot. Although not resembling the robot, the seven joints on the joystick will allow any necessary position and orientation of the robot end effector to be reached, therefore not falling short of the robot's abilities. The physical constraints of the robot itself will be provided through software at a later stage.
Encoders	The chosen magnetic rotary encoders provide the simplest and most cost effective solution for encoding capabilities, while still providing a 10 bit resolution. This value, converting to 1024 positions per revolution, is well within the specified requirements and provides reliable position and orientation tracking.
Safety	The final design provides several safety aspects: the "dead man switch", two buttons to actuate the gripper and brakes on Joints 1 to 6 of the joystick. The light-weight, gravity oriented design provides a setup that requires minimal input from the user, thus making the system safer by removing extra weight from the operator's hands. These safety aspects are all a result of combining the best features from the different concepts.
Intuitiveness	Unhindered movement in a 3D space, with the operator's thumb and index finger acting as the gripper, gives the intuitive sensation of working inside the surgical area. There is no fulcrum effect for which the surgeon has to compensate. Unlike previous concepts, the movement of the final design is not hindered by links based on the surgical robot – no knowledge of the working principles of the PSM and SSM is necessary to control the robot.

4.4 Assembly

Assembly of the joystick is fairly straightforward, a process simplified by the assembly and subassembly drawings created during the detail design process. Appendix A contains the final assembly drawing, while Figure 18 shows the assembled joystick.
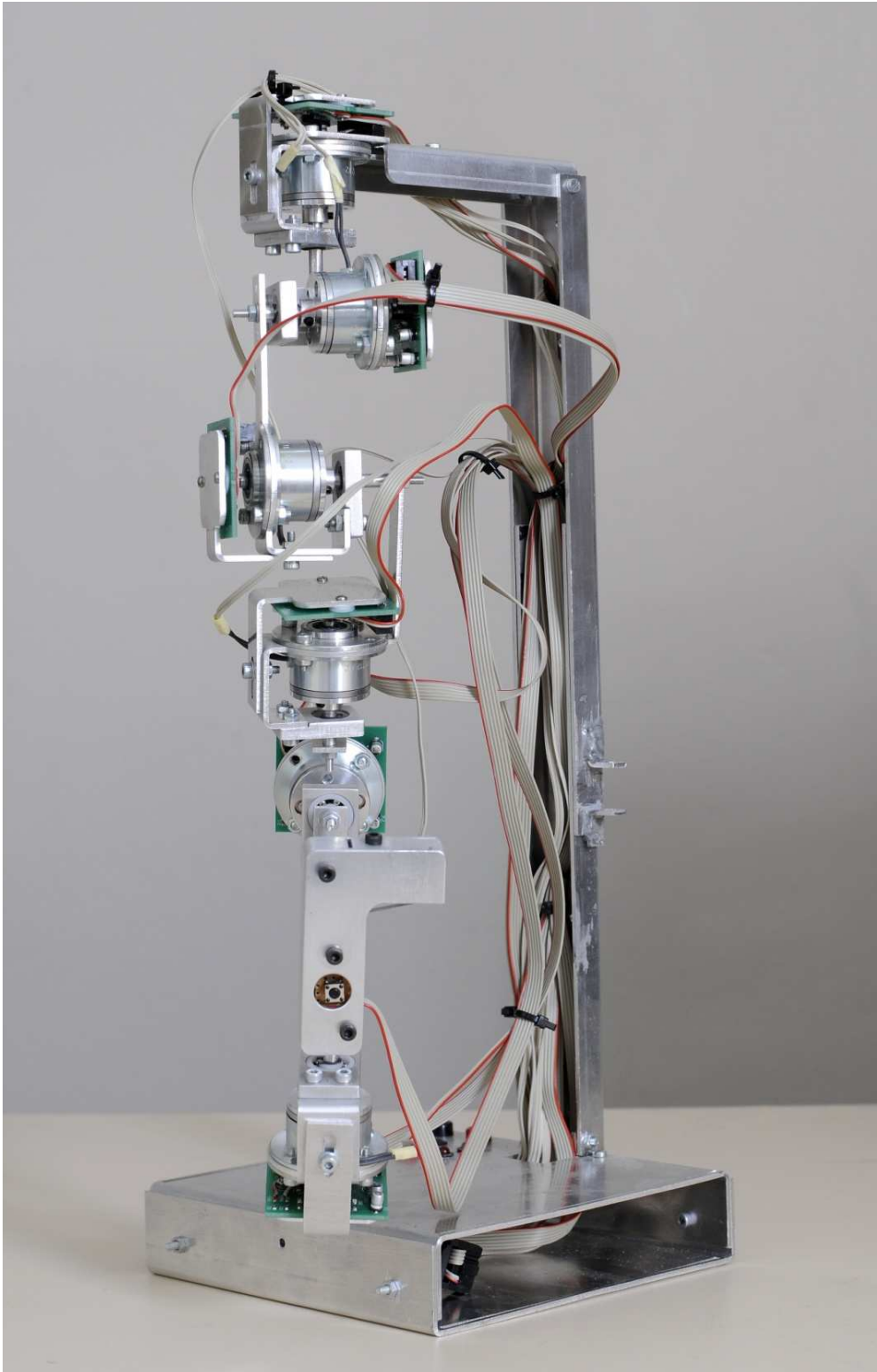
Figure 18: The assembled joystick

CHAPTER 5   ELECTRONIC DEVELOPMENT

This chapter covers the design and manufacturing of circuitry responsible for the electronic aspects of the joystick. Firstly, the main objectives pertaining to the joystick's electronics are listed and the applicable engineering specifications are derived from this. These specifications guide the electronic design process. The chapter covers the design and implementation of the encoder systems, the brake circuitry, the power supply and regulation circuitry, the processor, communication between all of these components and any other necessary features to ensure the correct functioning of the joystick. Preparations for communication with the SSM and PPM are also presented.

5.1   Electronic specifications

The objectives that influence the joystick's electronic development state the need to design the encoder system and to implement electronics enabling the combined functioning of the joystick components. An effective communication system within the master setup, as well as between the master and the slave, is also necessary. These aspects can be summarized conceptually as system inputs, system outputs and the main controller that regulates everything in between. Figure 19 shows these relationships diagrammatically:
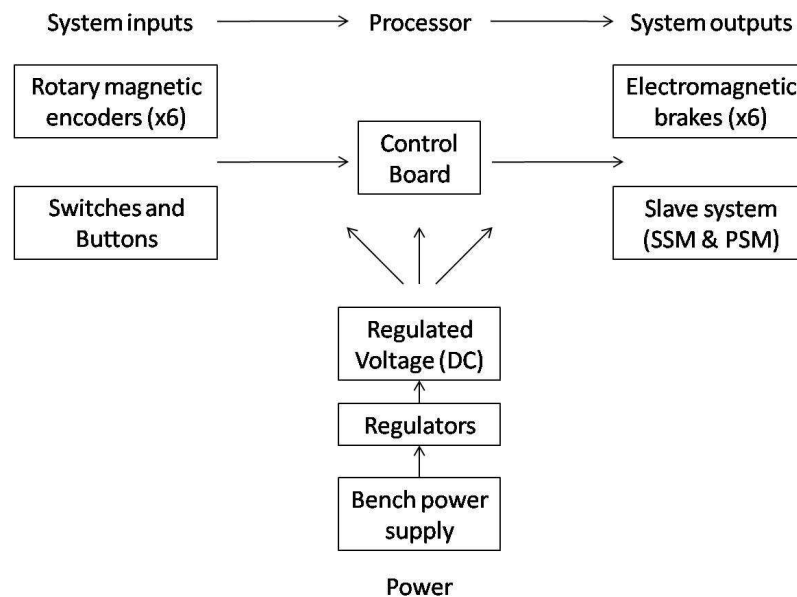
Figure 19: The electronic specifications diagram

The system inputs are the rotary magnetic encoders, the buttons and switch in the gripper and any other input switches, as they all provide one-way information that is to be interpreted by the control board. For the inputs, several specifications should be met. Firstly, circuit design and PCB construction for the encoders and the buttons/switches are necessary – they have to be supplied with

the correct operating voltages and a way for them to communicate with the control board should be available.

The system outputs are the six electromagnetic brakes, as well as the PSM and SSM slave systems; they receive instructions from the control board and have to execute certain tasks. Specifications for the outputs include: providing the correct switching voltages for the brakes and creating communication capabilities between the control board and the outputs.

It is evident that a power supply to the system is necessary. As the different components are bound to have different supply voltages, a voltage regulating system will have to be implemented. Extra circuitry to aid the correct functioning of the abovementioned system will certainly be necessary and will have to be designed and constructed accordingly. Finally, a suitable control board or processor will have to be chosen to enable all of the control and communication capabilities of the joystick. The following sections will explain the detail behind each of the abovementioned aspects of the electronic development.

5.2   Control board

This is the centre point of the master system. It interprets data from the system inputs and sends instructions to the system outputs and is responsible for calculations, control, data sampling and communication. Considering this, the control board should be chosen based on the following main requirements: it should be fast enough to allow ample time for data sampling and control, it should be easy to use, it should have enough input/output pins to account for the system inputs and outputs, it should allow a master-slave communication setup and its functionality should cover any requirements from the joystick.

Chapter 2 mentioned that the Arduino Mega 2560 development board was used as the control board for the SSM, which implies that software for its functioning had already been created in the Arduino platform. To build further on this basis, and due to previous experience with the Arduino platform, another Mega 2560 board (shown in Figure 20) was chosen to be the master control board.

Figure 20: The Arduino Mega 2560 (*Arduino Mega 2560,* [S.a.])

The Mega 2560 is built around the 16 MHz ATmega2560 microprocessor from Atmel. Its specifications that satisfy the control board requirements include 54

31

digital input/output pins of which 14 can be used as pulse width modulation (PWM) pins and six can be used as external interrupts. It has four serial ports with transmit and receive functionality, as well as Serial Peripheral Interface (SPI) and I^2C communication capabilities. Furthermore it has 16 analogue input pins, it operates at 5 V with a 7-12 V supply voltage, it has a built-in voltage regulator that also provides a 3.3 V output, it communicates via a USB interface with a computer and the open-source software platform is very user friendly with more than enough online support available (*Arduino Mega 2560,* [S.a.]).

5.3   Encoder implementation

As already specified in Chapter 4, the AS5040 magnetic rotary encoder was chosen for the joystick joints. When designing a circuit for this component, several specifications from its datasheet are important to note (*Arduino Mega 2560,* [S.a.]):

- The AS5040 requires a supply voltage of 5 V.
- It has five different output modes, namely: synchronous serial interface (SSI), PWM, quadrature (with index), step/direction (with index) and 3-phase commutation output modes.
- Serial readout of multiple interconnected AS5040 encoders is possible with the Daisy Chain method.
- It has a failure detection mode for the magnet position.

Support was found in the form of an online project (*Magnetic Rotary Encoder 1.0*, [S.a.]) where a circuit was designed for the AS5040. With this as a basis, a prototype circuit was constructed and tested with a BMN-35H magnet (specified by the AS5040's datasheet). After successful testing, a circuit was designed using the freeware version of Eagle layout editor for circuits and PCBs. The complete, assembled PCB is shown below in Figure 21. The electronic circuit of the encoder PCB is given in Appendix B, where all subsequent electronic circuit designs can be found.
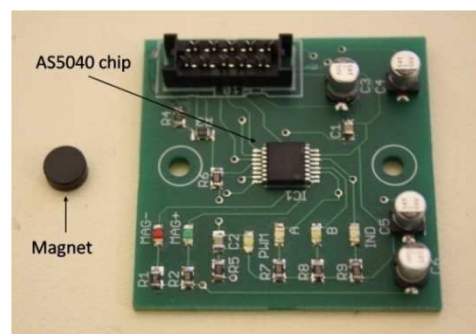
Figure 21: The encoder PCB and magnet

The circuit was designed to enable all five output modes, although the quadrature output mode was chosen because of the following reasons. Firstly, quadrature encoding is a straightforward, tested and easy to use method that senses rotation accurately and Arduino support provides several libraries to simplify the

necessary software development; and secondly, it only requires the A and B quadrature pins on the AS5040 chip to be connected to two input pins on the Mega 2560.

Then, the magnet position sensing option was included to enable failure detection. There are two pins on the AS5040, a MagINCn pin and a MagDECn pin, that indicate when there is respectively an increase (magnet moves too close) and a decrease (magnet moves too far) in the magnetic field strength. These pins have to be connected to two input pins on the Mega 2560. The only other connections necessary for the circuit to function are the 5 V input voltage and ground, which results in each encoder using four Mega 2560 input pins (totaling to 24) and common 5 V and ground connections.

Finally, the use of six encoder boards, and therefore six AS5040 chips, motivated the possible incorporation of the Daisy Chain method to read data in parallel on the same serial line. This method was, however, not used because the Mega 2560 provided enough digital input pins to warrant a separate communication interface for each encoder and using this method would unnecessarily complicate the software setup.

5.4   Switches and buttons

Many switches and buttons were incorporated to serve as interfaces through which the operator can give instructions to the surgical robot. The gripper contains one micro-switch and two basic push buttons, while the joystick plate houses several toggle switches. All circuit designs for these buttons/switches are shown in Appendix B.

The basic design of all the button and switch circuits are the same: it requires a 5 V supply level, a resistor to draw the current to GROUND and a connection to the Arduino control board. If the switch or button is pressed, the circuit closes, the resistor draws current and a 5 V level is registered at the Arduino pin. This instruction can then be interpreted by the control board and a task can be executed.

5.5   Brake implementation

The Miki-Pulley 112-02-11 electromagnetic micro brake is easy to incorporate, with very straightforward circuit design. The datasheet specifies that it requires a 24 V supply voltage and that surge protection is necessary for the brakes (*Miki-Pulley*, [S.a.]). For the latter reason a TNR9G820K varistor, which has to be connected in parallel with the brake voltage input, was included with each brake.

Furthermore, direct driving of the brakes by the Mega 2560 is not possible because the digital output pins on the board can only supply 5 V. The need for switching relays was evident and consequently six PCB-mountable Reed relays (*Reed*, [S.a.]) were acquired, one for each brake. The coil of such a relay requires a 5 V switching level, which then closes the relay contact and provides

the brake with a 24 V level. The circuit designs for the brakes, the varistors and the relays can be found in Appendix B.

5.6 Communication setup

An accurate means of communication was necessary between the master and slave systems to enable control of the surgical robot and the electronic design of the joystick had to account for this. The Mega 2560 provides several ways to enable communication: it has four serial ports as well as SPI and I^{2}C communication capabilities. Serial communication was chosen due to its simplicity of use. The multiple serial ports (to enable different serial lines for different instructions/tasks) support this decision. A direct 4-wire cable connection is used between the master and slave systems.

The Serial3 and Serial2 ports of the Mega 2560 were chosen on both the master and slave control boards – two ports were chosen to allow multiple messages to be transmitted and received simultaneously. In order to ensure correct communication, the Serial3 transmit pin (TX3) on the master board was connected to the Serial3 receive pin (RX3) on the slave board, and vice versa. The same setup was used for the Serial2 pins. This communication method was tested successfully between the two Arduino boards at a baud rate of 9600 bits/s.

5.7 Power supply

It was mentioned previously that several of the components require different supply voltage levels to operate, specifically 24 V for the brakes, 7-12 V for the Arduino Mega 2560 and 5 V for the encoder and button/switch circuits. This necessitated the use of voltage regulators to supply different voltage levels from 24 V down to 5 V.

The joystick input voltage was chosen to be 24 V. During development of the joystick system a standard laboratory bench power supply system was used, but the SSM's 24 V – 3 kW power supply could also be used for this purpose. An ON/OFF switch (with an LED to indicate the state) was used on the main 24 V line to the joystick to enable the operator to start or stop the joystick operation independently.

From 24 V the level was brought down to 12 V by an L7812C voltage regulator (*L78xx Voltage Regulator*, [S.a.]). The circuit (shown in Appendix B) includes a 1N4007 diode to ensure one way current flow and circuit protection. Due to the high voltage drop over the regulator, the high power dissipation (a value calculated and shown in Appendix C) made the use of a heat sink necessary. When tested without a heat sink, the regulator shut down and did not function, but the heat sink allowed the circuit to work correctly. A basic 12 V (DC) computer fan was included to ensure no overheating.

From the 12 V output of the L7812C component, an L7805C voltage regulator (with the same circuit design and diode implementation as above) was used to

bring the 12 V level down to 5 V. The inclusion of a heat sink was not necessary for this regulator.

This then provided the overall joystick circuit with 24 V, 12 V and 5 V levels that are sufficient for the requirements of the different components.

5.8   Construction

All of the circuit designs above had to be constructed to enable functioning of the joystick. Construction and assembly of the joystick's electronic system were done as follows:

Firstly, all of the necessary components for the encoder circuits were acquired (including the six AS5040 chips, capacitors, resistors, etc.) and these were soldered to the six encoder PCBs. Each PCB then also contained a 10-way ribbon cable connector to connect it to the voltage supply and Mega 2560, although only a six wire cable was used to provide each encoder with the necessary connections.

The finished encoder PCBs were then connected to their respective PCB holders (see Section 4.3.2), which could be fixed to the different joystick joints. The ribbon cables from each encoder were arranged so as to allow minimal interference with the joystick's movement, they were fixed at the appropriate places with cable ties and all were brought together and lined down along the joystick base's vertical extension to the square base. At the cable ends, ribbon cable connectors were fixed to allow them to be plugged in where necessary. The two wires from each electromagnetic brake were aligned and fixed in a similar fashion, while all the brake wires were connected at their ends to one ribbon cable connector.

For the gripper buttons and switch, the PCBs were made from custom shaped prototype boards, as elaborate circuit design was not necessary. The PCBs were shaped to fit into the gripper assembly. When the switch, the buttons and their wires were soldered, the PCBs were bolted to the inside of the gripper and the wires were aligned and fixed as explained above. At the joystick base, these wires end together in a ribbon cable connector. Figure 18 depicts all of the wires in the final joystick assembly.

One main PCB then accounted for all of the rest: the connection ports for the encoders, the brakes and the gripper buttons, the power supply input, the voltage regulator circuits, the surge protection and relay switches for the brakes, and the necessary connections to the Mega 2560 (mainly power supply and digital inputs). This PCB and its circuit design can be seen in Appendix B.

The PCB contains a 60-way ribbon cable port where a connector can be plugged in to connect the Mega 2560 to the PCB and therefore to the system inputs. During the prototyping phase, the wires on the other end of this 60-way ribbon cable were soldered to specific ports of an Arduino Mega 2560 prototyping shield, which could be plugged in to the Mega 2560. This allowed the system

input wires to be connected to their respective input ports on the Mega 2560. This worked fine for prototyping purposes, but to provide a neat solution where loose wires were excluded, a new PCB was designed to replace the shield. This PCB has the same function of the shield, but with an added ribbon cable port identical to the one on the main joystick PCB. This allows the Mega 2560 to be connected to the joystick by means of a single 60-way ribbon cable with plug connections on both sides. The new PCB, plugged into the Mega 2560, is shown in Figure 22 below.
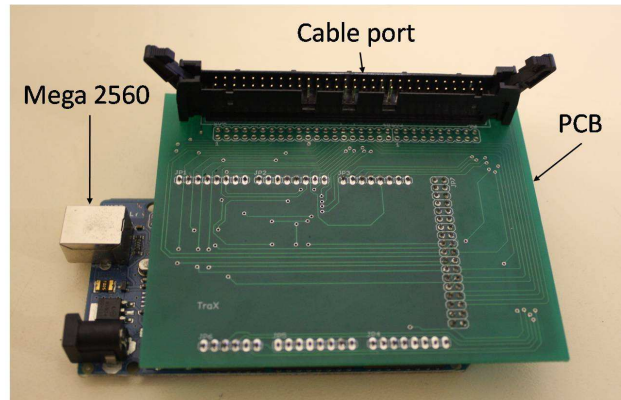
Figure 22: The final PCB-Arduino connection

The joystick plate mentioned in Chapter 4 served as a surface onto which the joystick could be mounted, but it was also used as a fixture plate for all the input switches with which the operator can control the robot. Six toggle switches were added, one to control each brake on the joystick and another three were added for robot operation purposes. Each switch circuit was designed in the same way as specified in Section 5.4 above, but LEDs were also added to indicate the state of the switch. Two PCBs, one for the brake switches and one for the other three switches, were designed and printed. Circuits are given in Appendix B, while the assembled brake switches are shown below as an example.
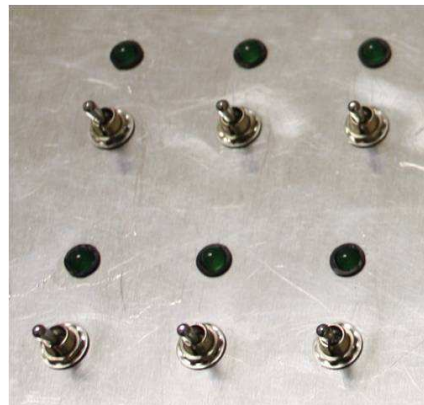
Figure 23: The brake toggle switches and LEDs

For all the specific circuit and PCB designs, prefer to Appendix B, where a pin configuration list showing the individual pin connections between the system inputs/outputs and the Mega 2560 can also be found.

CHAPTER 6   MASTER AND SLAVE INTERFACING

Chapters 4 and 5 covered the main objectives pertaining to the mechanical and electronic construction of the joystick. With these aspects completed, the next important objectives were to implement an effective communication system between the master and slave systems and to design the control system with which the joystick controls the surgical robot.

In order for these objectives to be satisfied, the master and slave systems had to be interfaced correctly. Firstly, the PSM and SSM systems had to function correctly according to requirements from the master side, which includes the correct functioning of each motor and encoder in the PSM and SSM and the ability to communicate with the central control board of the surgical robot. Where necessary, adjustments or changes had to be made. Then, there had to be a centre of control with which the master system could communicate. Once these aspects were set up, communication and control could be executed.

This chapter covers the work that was done to ensure correct interfacing between the master and slave system. The shortcomings of and resulting changes to the SSM and PSM systems are discussed, as well as the communication setup between the master and slave.

6.1   Slave system shortcomings

When the interfacing section was started the state and functioning of both the PSM and SSM were not as required – their mechanical aspects listed in Chapter 2 were only set up upon completion of this chapter. The intended state was for the PSM to be mounted to the SSM and for all the motors and encoders to work and for control to be executed by the central control board of the SSM. But, although work had been started on this process, the slave system was incomplete. The main control circuit on the PSM had been removed to allow control by the Mega 2560 on the SSM and hardware had been set up to accomplish this, but software was not in place to control the PSM. Furthermore, the PSM was fully disassembled, with two actuating cables broken. At the SSM side, very elementary software had been put in place to control its three motors and the electronic setup needed changes.

To successfully interface the master and slave systems and to allow control of the robot by the joystick, tests had to be conducted to find all problems and several changes had to be made. The main problems of the PSM and the SSM are listed in Table 4 and Table 5, respectively, below. They are divided into sections for motors, electronics, encoders, structure and functioning.

Table 4: Problems experienced with the PSM

Section	Description of problem
Motors	The 12 V Faulhaber DC motors used in the PSM have gearboxes with a ratio of 1526:1. With the output from the motor itself being 5200 rpm, this results in a high torque output at a maximum of almost 1 rpm, which is very slow and too strong. Complications arose when the actuating cables broke multiple times during testing. Furthermore, the motor controlling the gripper action did not function.
Electronics	The ribbon cable connectors through which the motor driving circuits on the PSM are connected to the SSM central control were not properly soldered to the ribbon cable wires and resulted in some of the PSM motors not functioning.
Encoders	No hardware was implemented to allow quadrature encoding of the encoder outputs on the Faulhaber motors. Only one line from each encoder output was used, implying that only the speed and not the direction could be sensed. Also, with the encoders being mounted at the back of the motors, and the high gear ratio and the cable mechanisms putting several degrees of freedom in between the encoders and the actual moving links, this method of sensing did not prove very accurate. Another problem was that no mechanical method for determining the encoder zero position, as it relates to the physical aspects of each joint, was incorporated.
Structure	It was determined that the choice of the motor gear heads was a result of an over-engineered process. Another negative aspect of their inclusion is the extra weight that was added to the system. This made the PSM heavy and increased the inertia of the whole slave system, making the rigidity and stiffness a problem.
Functioning	During testing of the PSM, one cable of the DOF 5 joint broke. This was determined to be the result of its motor malfunctioning and the opposing motor and cable mechanism putting too much tension on it. The gripper cable also broke several times, with a direct cause not evident – the motor and the gripper mechanism functioned correctly. Ultimately, it was ascribed to a fundamental flaw in the spherical wrist design of the PSM, causing Joints 5, 6 and 7 to work against each other, which resulted in high tensions in the cables. The high torque output from the motor gear heads aggravated these occurrences.

Table 5: Problems experienced with the SSM

Section	Description of problem
Motors	The two Parvalux motors of Joints 1 and 2, with their worm-spur combination gearing (154:1 ratio), both have several degrees of backlash on them, especially in orientations where a large amount of torque is exerted on the joints by the weight of the rest of the slave assembly. Worst (2012) had added damping shocks and brakes to the system, but the backlash was still present. The weight of the motor-gearbox combination also poses a problem as it increases the inertia of the system. The rails on which the linear motor causes the PSM to move were determined to be misaligned, causing an increase in torque and a movement hindrance.
Electronics	When the interfacing phase began, the SSM electronics were in disarray. The Mega 2560 was in place to control the three motors, but connections between the control board and the motors and encoders were arranged in a disorderly fashion, while documentation indicating the pin configuration was faulty. This created the risk of wires coming loose at any time and made fault-detection difficult. One of the two Parvalux motor brakes was faulty at the interfacing phase start.
Encoders	As was the case for the PSM, the encoders of the SSM were not interfaced with hardware to allow quadrature encoding. Only one wire was used per encoder to get a pulsed output, giving only speed and not direction. The two Hengstler encoders were also fixed to the two base motor outputs in a makeshift way that made it prone to misalignment. Also, they were connected to the motor output shafts and not the gearbox outputs, thereby putting a degree of freedom between the sensed output and the actual moving link. Furthermore, no manner of detecting the zero position of any of the three encoders was available.
Structure	The mass of the PSM, the two Parvalux motors and the mechanical links on the SSM result in a large moment of inertia. Added to the backlash on the gearboxes, this creates a system with low rigidity and stiffness. In turn this decreases the accuracy with which the movement and position of the surgical end effector can be controlled.
Functioning	The moment of inertia of the assembled robot creates the need to control the motors differently at places where more torque is needed. The backlash and lack of stiffness creates problems with position sensing, implying that the encoder information will always contain significant errors when compared with the true mechanical position. This becomes a problem when the robot is ordered to stop at a certain encoder value, but the backlash causes the robot to move slightly further afterwards. The damping shocks help to prevent this to a certain extent, but a negative implication was that they create constraints to the movements of the two Parvalux motors. Also, a very elementary way was in place to move each motor of the SSM on its own, but not simultaneously.

6.2   PSM adjustments

The adjustments for the PSM are divided into mechanical and electronic aspects.

6.2.1   Mechanical aspects

As the main objective is to create a joystick that can control the surgical robot's movement, the decision was made to not concentrate on fixing the fundamental design flaws in the PSM, as this would limit the time available to focus on the main objective. Therefore, no design changes were made to the mechanical aspects of the PSM.

However, extensive testing on the functioning of the cable mechanisms and the whole PSM were done. For this reason, and because of the broken cables, the PSM had to be disassembled and reassembled multiple times. It was noticed that the spherical wrist design resulted in more and less tension being put on different cables in different orientations of the wrist, all of which are unwanted. For example, when the gripper is actuated while Joint 5 (the "elbow" joint) is stationary and in the flexion position, enough force is exerted to rotate Joint 5 slightly towards its extended position. Also, during testing it was evident that the motors were too strong, as cables broke without any indication that the motor was exerting more torque than normal.

These problems were fundamental and could not be solved quickly. The emphasis was shifted to solvable problems that would increase the ability to satisfy the thesis objectives.

6.2.2   Electronic aspects

Firstly, the ribbon cables connecting the PSM motor controller PCB to the SSM were replaced. Five 10-way ribbon cables were used, one for each of the five PSM motors and their respective encoders. Four of the wires were connected to the encoder, two to supply 12 V and ground and two connected to the A and B quadrature output lines from the encoder. The connectors that plug into the motor driving circuits were also soldered and fixed to the ends of the new ribbon cables. Three wires (direction1, direction2 and enable) were necessary for this. The final three wires of each ribbon cable were not connected, except for the cable connected to one of the Joint 5 motors, where another two wires were used as 12 V and ground supply to the PSM PCB.

The new wires allowed quadrature encoding, i.e. sensing speed and direction, to occur. The incorporation of zeroing switches on such a small scale would necessitate design changes to the PSM and was therefore ignored.

Software and control adjustments of the PSM were only applied after the main SSM adjustments had been made and will be explained below.

6.3   SSM adjustments and changes

The adjustments for the SSM are divided into mechanical and electronic aspects.

6.3.1   Mechanical aspects

For the same reasons as mentioned for the PSM above, no major design changes were made to the structure of the SSM. The whole system was checked, however, to see if any adjustments could be made to improve the stiffness or general structure. The linear motor was removed from the Joint 2 link and fixed again to ensure that it was tightly fixed to the assembly. It was noticed that only one bracket was used for this fixture, an aspect that could be problematic and should be addressed in further design iterations. The coupling connecting the Joint 1 motor to its shaft was also tightened periodically to ensure stiffness – this connection was also noted as a possible addition to the backlash and position sensing problem.

The damping shocks were removed from the system, after it was realized that one was not functional. New shocks were acquired, after which they were fixed to the system again, although not both in the same position as before. The original fixing position for one shock constrained the movement of the first joint, as it was in the way of the Joint 2 motor while Joint 1 was rotating. It was then fixed in a new position to minimize the constraint on the movement although certain constraints were still present on both motors because of the damping shocks' maximum stroke length. The backlash problem was also not fully removed.

During a motor control test of the SSM, the Joint 1 motor moved too far in a CCW direction and the movement knocked the encoder at the back of the Joint 2 motor against the robot base structure. This base structure effectively constrained the movement of the Joint 1 motor (in much the same way as the shock constrained it) and a piece of the structure was removed to enable unconstrained movement (refer to Figure 11). During this occurrence, the Joint 2 motor encoder came loose and had to be fixed again. This was then used as an opportunity to fix both Hengstler encoders in a sturdier manner to prevent misalignment errors. It should be noted, however, that the correct mounting surface would require proper detailed design and machining.

While the encoders were being fixed, the absence of zeroing switches to determine the zero positions of Joints 1 through 3 was also addressed. Zero positions on each joint were chosen as the farthest position that the motors can reach, considering physical movement constraints, in a certain direction. For Joint 1 it was chosen as the farthest position in a CCW direction, for Joint 2 the farthest position in a CW direction and for Joint 3 (the linear motor) it was chosen to be the highest position. At each of these places, a micro switch was fixed so that, when the switch is pressed at the edge of the motor's movement, a signal would be sent to the control board, which would then in turn stop the motors and assign a specific and known value to the encoder position. These zeroing switches are shown in Figure 24 below.

41

DOF 1   DOF 2   DOF 3
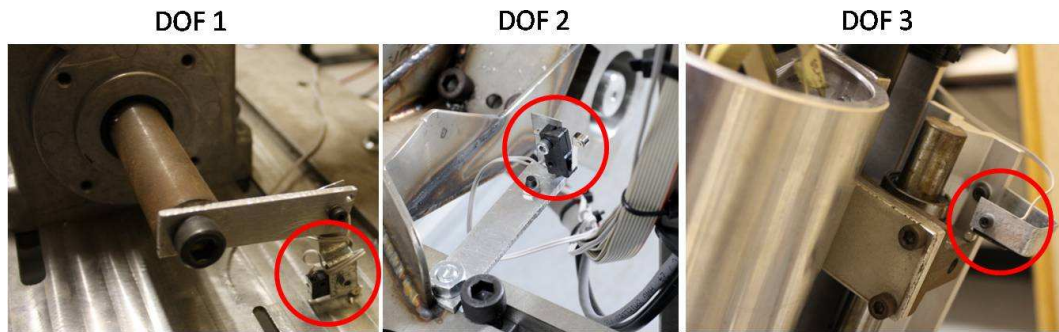
Figure 24: The SSM zeroing switches

The final adjustment to the mechanical aspects of the SSM was to design a holding plate and enclosure for all the SSM electronics. Before this addition, the power supply was placed loosely on the robot base structure and all of the electronics and excessive wiring were contained inside a makeshift wooden box on top of the power supply. To have an ordered setup, a few additions were made. Firstly, a large aluminium plate was designed to fit over the base structure of the robot and to serve as a fixing surface for the power supply and the electronics. Secondly, an aluminium box, which would be fixed to the aluminium plate, was designed to hold all of the electronic circuits and wiring. Figure 25 below shows the end result. One can see the plate on the base structure, the power supply at the back and the electronics box in front. The wiring from the PSM and SSM motors and encoders, and any extra wiring, all enter the box at a hole in the side, from where they disperse to their respective circuits.

Figure 25: The SSM electronics box and power supply assembly

42

6.3.2   Electronic aspects

As was mentioned, the original state of the electronic circuits and wiring made the system prone to loose wire problems, which made fault-detection difficult. Because of this, the decision was made to redo the main SSM circuitry. Several circuits were kept as is, including the power circuitry, the control circuits for the two Parvalux motors and the control circuits for the linear motor and the two Parvalux motor brakes. The main adjustments were applicable to all of the SSM and PSM encoder inputs to the control board, the motor control of the PSM, the zero switches and any other necessary connections.

A similar approach to that of the main joystick PCB design was followed. One PCB was designed to take all of the encoder inputs and all of the switch inputs, to give the information to the control board and then to send instructions to the motors and brakes. Figure 26 explains this setup in terms of system inputs and system outputs.
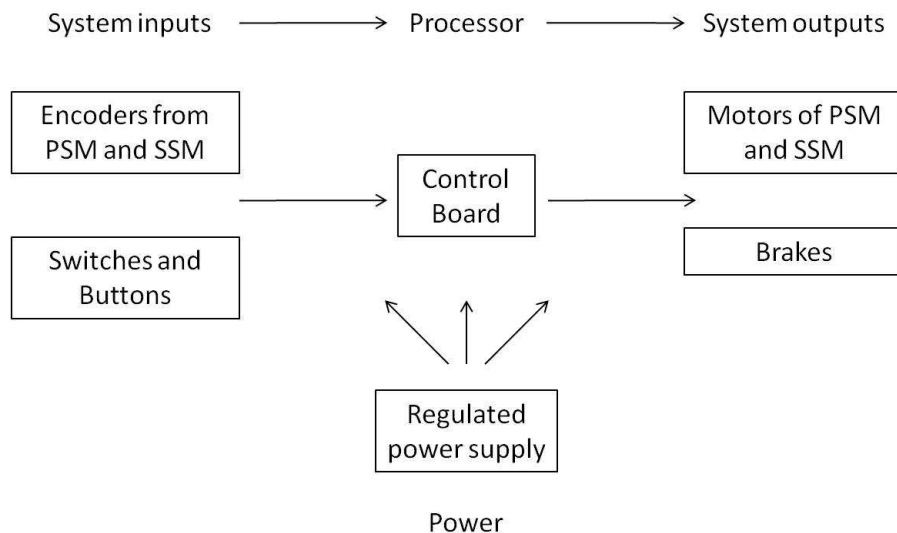
Figure 26: The SSM electronics diagram

The PCB receives a 12 V and ground input as a supply to the PSM and SSM encoders. It is fitted with several 10-way header boxes where the system inputs and outputs can be plugged in. For this reason, 10-way ribbon cable connectors were connected to the ends of the ribbon cables coming from the PSM – five cables, one for each motor as explained in section 6.2.2 above. The same was done for the encoders on the two Parvalux motors, this time incorporating the necessary wires to allow quadrature encoding. For the linear motor, the control outputs and encoder inputs were all routed to a single 10-way header box on the PCB. Again, as for the joystick setup, a 60-way ribbon cable (with connectors) was used to connect the PCB to an Arduino Mega 2560 shield, which could then be plugged into the Mega 2560. Also, after prototyping was done, a new PCB was designed to replace the shield and wires.

Finally, a safety switch was used on the main 24 V supply voltage line. This can be seen on top of the electronics box in Figure 25. This switch can cut off power to the whole robot at any given time and it serves as a safety feature to prevent accidents from occurring.

6.4   Software adjustments

No software was in place to control the PSM and an elementary software setup was used to move the motors of the SSM individually. Methods to start and stop the operation of any motor at any time, and to do this at any speed, were needed. This was accomplished by communicating with the Mega 2560 of the slave via a serial port on a computer.

The main software functions will be explained in more detail in Chapter 9, but after successful implementation of basic motor control functions, three instructions could be sent to each motor of the surgical robot: move CW at a certain speed, move CCW at a certain speed and stop. This allowed all of the motors to be controlled either simultaneously or individually.

A function was also created to zero the three base motors of the SSM: when the instruction is given, the motors move (one after the other) towards their zeroing switches. When the switches are activated, the motors are stopped and the appropriate position values can be written to the motor encoders.

For the quadrature encoding of the encoder signals, new software also had to be implemented. An open source Arduino library was found and used to determine the movement specifications at each joint. The library provides different modes of operation, but its optimal functioning requires one quadrature output line (A) from the encoder to be connected to an external interrupt pin on the Mega 2560, while the other line (B) can be connected to any digital input pin. A 10 bit resolution encoder value can then be read from the library at any time. The Mega 2560 has six external interrupt pins, which meant that this library could be implemented for every joint on the joystick, while at the robot there was still one joint left because of its seven joints. The encoder output lines of this joint, the linear motor of the SSM, was instead connected to two digital inputs on the Mega 2560. Another quadrature encoding routine with 9 bit resolution (512 positions) was then written to successfully determine the movement specification at this joint.

6.5   Communication setup

As was mentioned in the joystick's electronic development section, the Serial3 and Serial2 ports of the Mega 2560 were chosen on both the master and slave control boards as means of serial communication between the joystick and the surgical robot. The transmit and receive pins of the slave Mega 2560 serial ports were connected, respectively, to receive and transmit pins of the same serial ports on the master control board.

Two ports were chosen to allow multiple messages to be transmitted and received simultaneously. The necessary data to transfer between master and slave would be the position and orientation information that the robot has to follow and any other instructions to guide the robot. This was divided into two types of data transfer: continuous and intermittent. Serial3 was used for continuous data transfer, where the position and orientation data was sent at a specific rate to the slave, while Serial2 was used for any other instructions sent to the slave based on the operator's input.

The software functions provided by Arduino were used for data transfer through Serial2, while a specific data transfer library was used for Serial3. It provided a simple way to send several data values simultaneously and continuously to the slave's serial port.

This communication setup was tested successfully at a baud rate of 9600 bits/s, for both serial ports. With successful communication and the ability to send instructions from the joystick to the robot, which could in turn carry out the instructions, the interfacing of the master and slave systems was complete. This process proved that the objective of implementing an effective communication system between the master and slave systems was satisfied.

CHAPTER 7   JOYSTICK AND ROBOT MODELLING

The previous chapter covered the work that led to the successful interfacing of, and communication between, the master and slave systems. The next main objective was to design the control system that will regulate how the joystick is able to control the surgical tool. To accomplish this, several sub-objectives were deemed important. Firstly, to enable control it is necessary to understand the kinematics of both the joystick and the robot. It is important to understand how the end point of the joystick can be tracked and how this information can be used as an instruction to the robot in order for the end effector to carry out the same movement. The key to satisfying these sub-objectives is modelling and the aim of this chapter is to cover these modelling basics.

Firstly, a theory section is given to provide the necessary background information on which the kinematic models of the joystick and robot are based. This is followed by separate sections on the joystick and robot models.

7.1   Modelling theory

"Robot Modeling and Control" by Spong *et al.* (2006) is a widely used textbook in the field of robotics that presents important theory on the modelling and control of robots and the conventions that accompany it. The Denavit-Hartenberg convention and other important concepts, as presented by Spong *et al.* (2006), are incorporated in this section.

As is evident from the mechanical design of the joystick and robot systems, encoders measure rotation or translation of a joint with respect to the link onto which the encoder is fixed. For example, Joint 3 on the joystick is located at the end of Link 2 and the Joint 3 encoder measures its rotation with respect to Link 2. This implies that each joint has its own coordinate system according to which its rotation or translation is measured and that the whole kinematic chain is a sequence of coordinate systems fixed end to end. By the convention of Spong *et al.* (2006), a robot manipulator has $n + 1$ links (numbered from 0 to n), connected by n joints (numbered from 1 to n) and at the end of each link i, the ith coordinate system (o_i) can be found.

The actuation of each joint is associated with a joint variable denoted by q^i and this variable differs for the two different joint types. At a revolute joint, q^i is the angle of rotation (θ_i); at a prismatic joint, q^i is the joint displacement (d_i). With the movement of each joint variable, the coordinate system at the end of that specific link also rotates by θ_i or moves d_i with respect to that link. The propagation of this movement from the manipulator base to the last link can be difficult to track. For this reason, the transformation matrices were created. By definition, the homogeneous transformation matrix A_i gives the position and orientation of the frame $o_i x_i y_i z_i$ with respect to the reference $o_{i-1} x_{i-1} y_{i-1} z_{i-1}$ (Spong *et al.*, 2006). When it is necessary to express the position and orientation of any reference frame $o_j x_j y_j z_j$ with respect to another reference frame $o_i x_i y_i z_i$, as is the case with expressing the end effector coordinate system with respect to the base

46

coordinate system of the joystick, the 4 x 4 transformation matrix T_j^i [2] is used. It is defined as follows:

$$T_j^i = A_{i+1}A_{i+2}\cdots A_{j-1}A_j = \begin{bmatrix} R_j^i & o_j^i \\ 0 & 1 \end{bmatrix} \qquad [1]$$

Here, R_j^i is defined as the 3 x 3 rotation matrix of $o_jx_jy_jz_j$ with respect to $o_ix_iy_iz_i$ and o_j^i is the 3 x 1 position matrix of $o_jx_jy_jz_j$ with respect to $o_ix_iy_iz_i$.

With the DH convention, a manipulator is modelled according to a certain set of rules, specifically with regards to how the coordinate frames are defined. If it is assumed that the first coordinate frame is denoted $o_0x_0y_0z_0$, the three basic DH principles are as follows:

(DH0)   The *z*-axis of each frame should be the axis of rotation/translation.

(DH1)   The axis x_1 should be perpendicular to the axis z_0, i.e. x_i should be perpendicular to z_{i-1}.

(DH2)   The axis x_1 should intersect the axis z_0, i.e. x_i should intersect z_{i-1}.

The only other specification is that the resulting coordinate frames should also conform to the generally accepted right-hand rule for setting up coordinate systems in three dimensions. When the model is set up according to these principles, the DH convention states that there exist unique numbers *a*, *d*, *θ*, *α* such that the transformation matrix A_i can be calculated as follows:

$$A_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [2]$$

The values *a*, *d*, *θ*, *α* apply to each separate joint and link and are respectively referred to as the link length, link offset, joint angle and link twist. Table 6 below explains the physical interpretation of these quantities.

Table 6: The DH link and joint quantities

Quantity	Interpretation
a_i	This parameter is the distance between the axes z_{i-1} and z_i and is measured along the axis x_i.
α_i	This parameter is the angle between the axes z_{i-1} and z_i and is measured in a plane normal to x_i.
d_i	This parameter is the distance from the origin o_{i-1} to the intersection of the x_i axis with z_{i-1} and is measured along the z_0 axis. For revolute joints, *d* is constant, while it varies for prismatic joints.
θ_i	This parameter is the angle from x_{i-1} to x_i and is measured in a plane normal to z_{i-1}. For prismatic joints, *θ* is constant, while it varies for revolute joints.

---

[2]   For clarification: a superscript is used, where applicable, to denote the reference frame, while a subscript denotes the frame that applies to the indicated point, vector or matrix

With the DH convention, transformation matrices and the robot modelling basics explained in Section 2.2 used as a starting point, the joystick and robot modelling could be executed. The following two sections show the modelling results.

7.2   Joystick modelling

Figure 27 below shows the joystick model as created according to the DH convention. It should be noted that, although the model indicates the reference frames, joint types and variable dimensions to describe the joystick, it was not created according to scale. Table 7 contains the values a, d, θ, α that apply to this specific model (the asterisk * indicates a variable value). These values were mostly taken from the CAD model of the final joystick design and were verified with measurements on the actual constructed joystick.
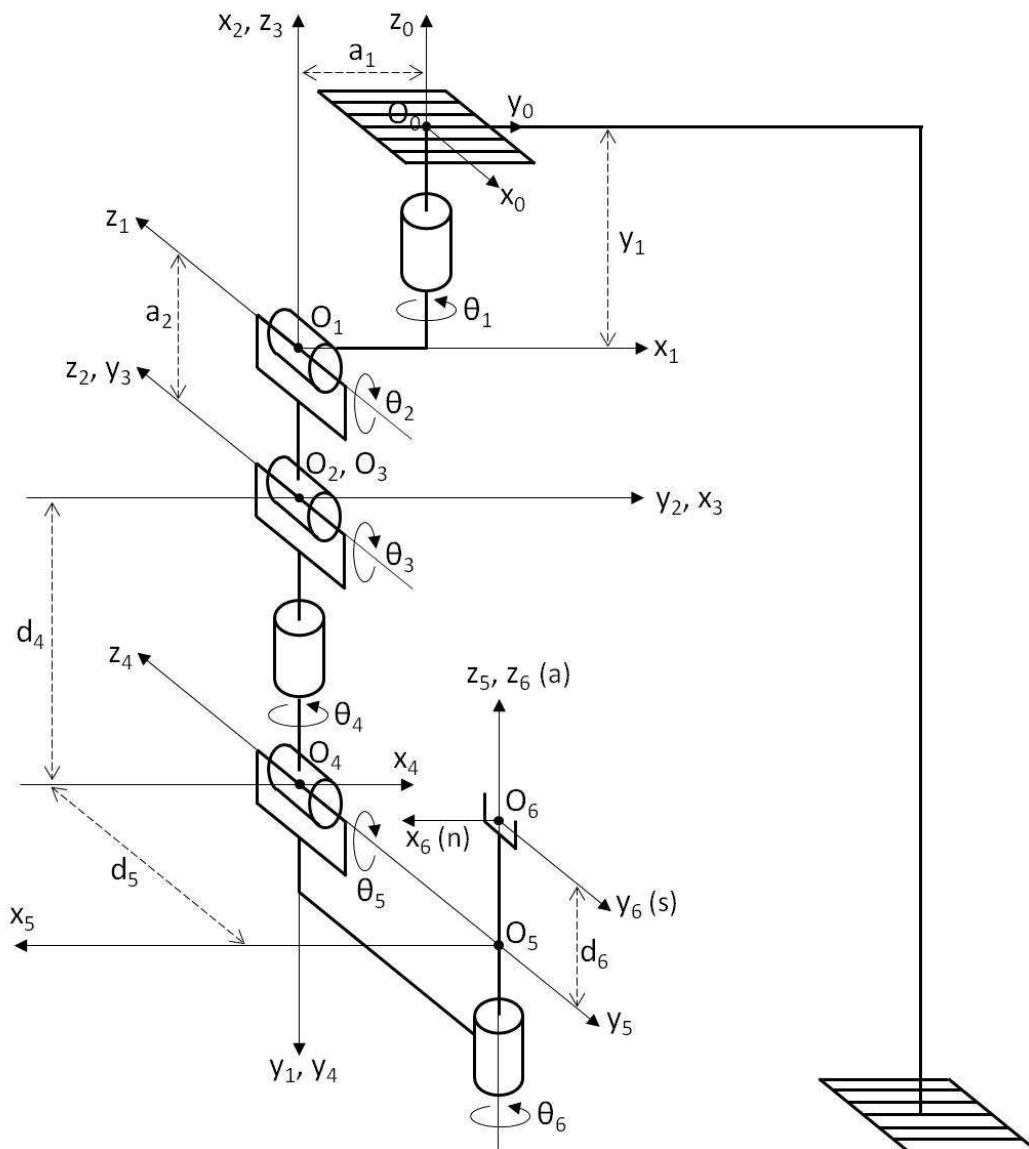
Figure 27: The joystick model

Table 7: The joystick link and joint variables

Link	a_i (mm)	α_i	d_i (mm)	θ_i
1	-20	-90°	-60.8	$\theta_1{}^*$
2	-65	0°	0	$\theta_2{}^*$
3	0	90°	0	$\theta_3{}^*$
4	0	-90°	-118.5	$\theta_4{}^*$
5	0	-90°	-95.3	$\theta_5{}^*$
6	0	0°	-27.6	$\theta_6{}^*$

Some positions are of interest in order to better understand the model. These include o_0, which was chosen to be situated in line with the rotation axis of the first joint and on top of the base layer surface, and o_6, which was chosen to be precisely in between the two gripper buttons. The other choices represent the physical aspects of the joystick and are simple to deduce. As far as possible, bordering reference frames were chosen in such a way as to minimise the number of resulting link variables, thus simplifying the model.

Another noteworthy point applies to the chosen orientations of the joints with respect to each other. According to the DH convention, θ_i is measured from x_{i-1} to x_i, which implies that θ_i is zero when axes x_{i-1} and x_i are parallel and in the same direction. It is evident from Figure 27 that the model was not set up in this way – rather, the starting θ_i values on the model were chosen to show the joystick in an easily understandable manner, thereby simplifying the model's interpretation. These starting values are shown in Table 8 below.

Table 8: Starting θ-values for the joystick model

θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
90°	-90°	90°	0°	180°	0°

7.3   Robot modelling

Figure 28 shows the robot model as created according to the DH convention, again not according to scale. Table 9 contains the values a, d, θ, α that apply to this specific model (the asterisk * indicates a variable value). An updated CAD model was not available at the time of creating this model and determining these variables had to be done through an alternative approach. It depended mostly on the specific position chosen for each reference frame's origin.

Table 9: The robot link and joint variables

Link	a_i (mm)	α_i	d_i (mm)	θ_i
1	0	90°	682	$\theta_1{}^*$
2	0	90°	-176	$\theta_2{}^*$
3	-162	0°	$d_3{}^*$	-90°
4	0	-90°	-520	$\theta_4{}^*$
5	0	-90°	0	$\theta_5{}^*$
6	0	0°	50	$\theta_6{}^*$
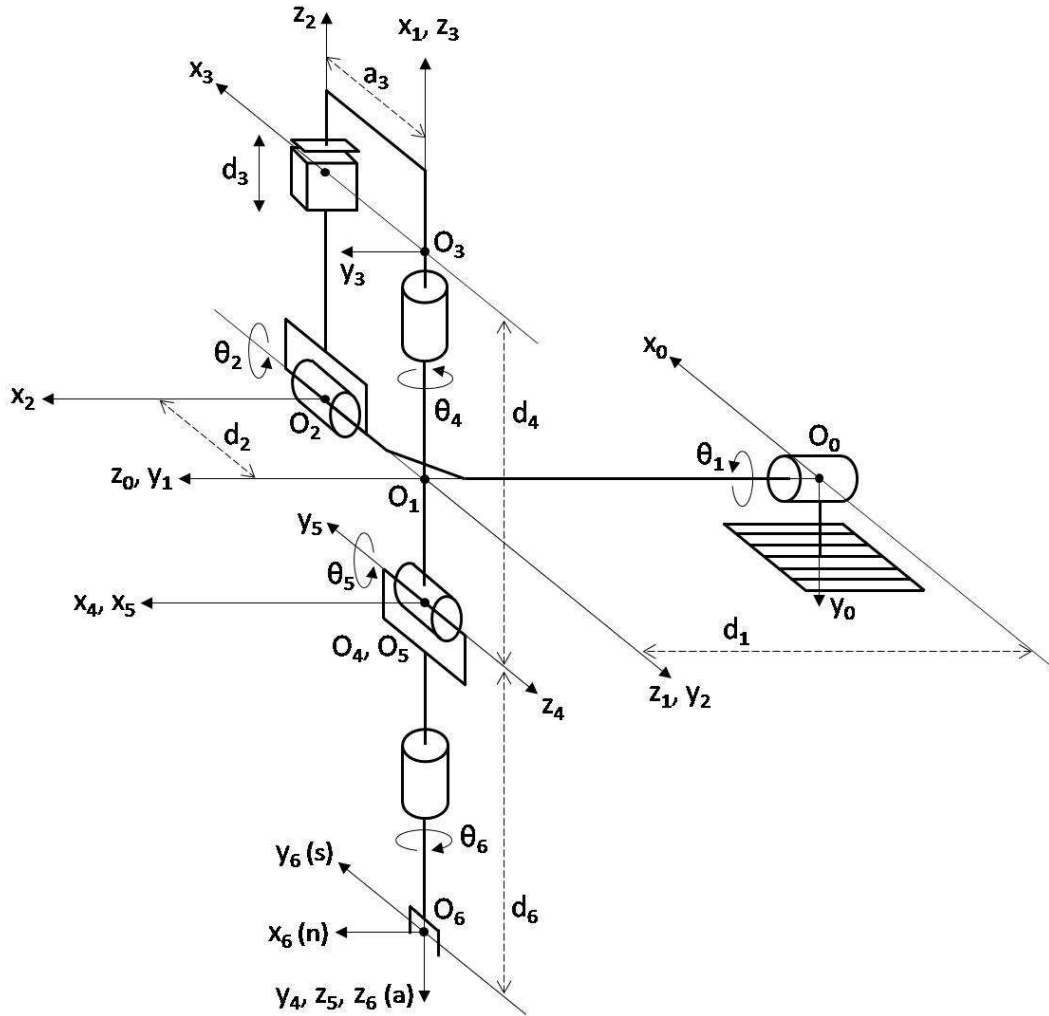
Figure 28: The robot model

Figure 29 below indicates the respective physical surfaces and positions chosen for o_0, o_2, o_3. For o_0 and o_2 the front flat surfaces of the Parvalux motors were chosen (where they intersect the respective z_0 and z_1 axes), while a flat surface on the linear motor bearing block was chosen (where it intersects axis z_3) for o_3. These points help to define the respective link variables d_1, d_2, a_3 and d_4.
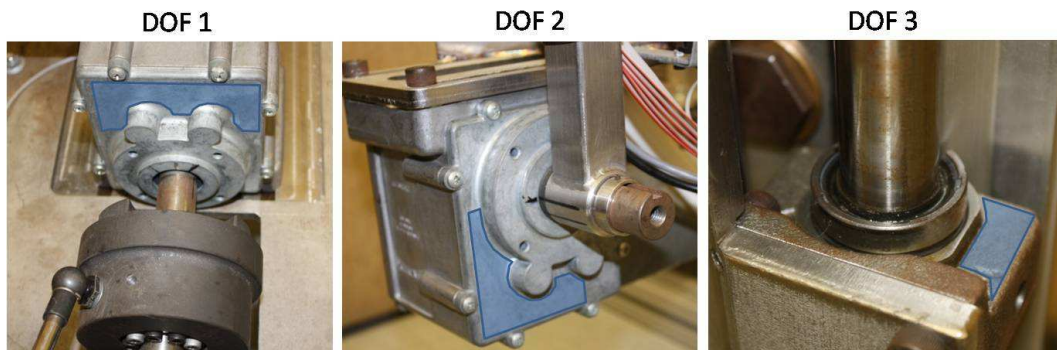
Figure 29: SSM zero position surfaces

50

On the model, the indicated position for o_1 is important in that no actuation of θ_1, θ_2 or d_3 will result in o_1 shifting from its position – it is always stationary. This aspect is inherent to the SSM design and was created to serve as the virtual trocar position. It is the point where the surgical robot enters the patient's body and around which pivoting takes place, thus ensuring the patient's safety.

Points o_4 and o_5 are the same and they are located at the spherical wrist (consisting of Joints 4, 5 and 6) centre point, commonly referred to as o_C. This phenomenon simplifies inverse kinematic calculations for the robot, as will be seen in the next chapter. As with the joystick model, o_6 was chosen to be situated in the middle of the gripper, in the area where the two claws come together.

Again, the variable angles and distance that are depicted in the robot model were chosen to best illustrate how the links fit together to form the easily interpretable robot model. These starting values are shown in Table 10 below.

Table 10: Starting joint variable values for the robot model

θ_1	θ_2	d_3	θ_4	θ_5	θ_6
-90°	90°	528 mm	90°	0°	0°

The generated models for the master and slave provided the required information on which to base the forward and inverse kinematic processes that follow.

CHAPTER 8   ROBOT CONTROL

This chapter combines the models developed in Chapter 7 and the necessary control systems to satisfy the thesis objective of regulating how the joystick is able to control the surgical tool. To accomplish this, some basic theoretical concepts are clarified. This builds further on the modelling theory that was provided in the previous chapter. With a proper theoretical background in place, the main specifications for this section are then established.

The generated control methods for both the joystick and the robot are presented. This covers the processes of deriving the necessary information from the joystick encoders and inputs, translating this into robot commands and then having the robot execute the necessary actions. With all the methods in place, the final operation of the control setup is explained.

8.1   Control theory

Forward kinematics is concerned with finding the end effector's position and orientation in terms of the joint variables, while inverse kinematics seeks to determine the desired joint variables for a given position and orientation of the manipulator's end effector. The concepts of forward kinematics for the joystick have already been described. Chapter 7 discussed how the DH convention can be used to create transformation matrices from the joint and link variables and how the position and orientation of the end effector with respect to the base reference frame can be found from this. A similar discussion on inverse kinematics for the robot follows below.

The primary objective is to have the robot end effector carry out the same movement as experienced by the joystick end effector, subject to elementary scaling and axis alignment operations. Thus, when the orientation and position of the joystick's end effector are known, the required orientation and position of the robot end effector (recalculated according to scaling and other factors) are also known. The problem is then essentially defined as the inverse of Equation 1: given the orientation and position on the right hand side of the equation, find the homogeneous transformation matrices A_i on the left hand side that can be used to calculate the joint variables.

Sections 2.2 and 2.3.1 respectively discussed the concept of a spherical wrist and the fact that the PSM has a spherical wrist configuration with an added gripper. Chapter 7 also referred to this concept, with the wrist centre o_C coinciding with points o_4 and o_5 on Figure 28. Spong *et al.* (2006) use this configuration to simplify the inverse kinematics problem with a process called kinematic decoupling, whereby the given information is used to first find the wrist centre and then the wrist orientation.

The fundamental principle is that actuation of the spherical wrist joints (those of the PSM) will not change the position of the wrist centre, meaning that o_C is only a function of the first three joint variables (those of the SSM). The desired end

effector position (o_6) can then be obtained by translating a distance d_6 along z_5 from o_C (refer to Figure 28). The mathematical interpretation of this concept is

$$o_c^0 \;=\; o \;-\; d_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{3}$$

which can also be written as

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} o_x - d_6 r_{13} \\ o_y - d_6 r_{23} \\ o_z - d_6 r_{33} \end{bmatrix} \tag{4}$$

Thus, with the required position and orientation (o and R) known, the wrist centre o_C can be calculated. Using a geometric approach, equations can be derived to determine the first three joint variables from o_C. These derivations are discussed in detail in Sections 3.3.3 through 3.3.5 of "Robot Modeling and Control" by Spong *et al.* (2006) and are considered too elaborate to repeat here. However, the equations (as they pertain to the forward or inverse kinematics) can be viewed in Appendix D.

Now, using forward kinematics and the first three joint variables, the orientation transformation R_3^0 can be determined. Also, from basic transformation matrix theory,

$$R \;=\; R_6^0 \;=\; R_3^0 R_6^3 \tag{5}$$

Using the inverse of Equation 5 and the fact that the inverse of a rotational matrix is equal to its transpose, we can derive that

$$R_6^3 \;=\; (R_3^0)^{-1} R \;=\; (R_3^0)^T R \tag{6}$$

The terms on the right hand side of Equation 6 are either known or can be calculated, while R_6^3 represents the rotation matrix that is needed to calculate the joint variables of the spherical wrist joints (all PSM joints except for the gripper). These variables are found as a set of Euler angles corresponding to R_6^3 (Spong *et al.*, 2006), the derivation of which will not be given here.

With the above methods, each joint variable of the surgical robot can be determined for a given position and orientation of the end effector. This enables control to be executed by the master on the slave.

8.2   Control specifications

In order to better execute control, a set of guidelines are necessary. The main thesis objective pertaining to control states that the joystick should be able to control the movement of the robot, but no detail is provided. There were also no requirements from the client as to what type of control was necessary. Specifications were then firstly created based on the important aspects from

other sections, e.g. safety aspects and size requirements of the working volume. The other main specifications were created with regards to the main outcome: moving the robot end effector. According to the above aspects, control specifications were divided into sections for safety, working volume and movement.

8.2.1   Safety

Several safety aspects have been discussed through different sections of this thesis; the responsibility of the control system is to incorporate these safety precautions. Firstly, the controlling software of the user interface should limit the movement of the robot joints where necessary. For this, the physical constraints of each joint have to be taken into account. This will create a 'safe space' within which the surgical tool is allowed to operate. The velocity of any or all of the robot joints can also be limited through software control.

The software should also provide a way of controlling the safety switch and how instructions are sent to the robot: only when the switch is activated should movement be transferred from the joystick to the robot.

8.2.2   Working volume

As explained previously, the working volume on the joystick side is downscaled to the robot side, where movements are smaller and more accurately executed. The control not only has to enable the working volume of the joystick to be related linearly to the robot working volume, but it should also ensure that these working volumes are in identical orientation. This also points to the importance of intuitiveness in the joystick-robot setup: the same movement as experienced by the joystick should be executed by the robot in the same direction and manner.

8.2.3   Movement

The movement specification for robot control is a complicated concept, as different control structures with different levels of complexity are theoretically possible. Spong *et al.* (2006) contains many sections with detailed discussions on kinematic control, dynamic control, force control and path and trajectory planning for robotic manipulators; and with ample time, all of these sections could be covered. However, since the main objective of this thesis was only to create a user interface for the surgical robot and to execute control on the surgical end effector, and since no requirements were provided as to the extent of the control, the decision was made to concentrate on a basic control structure in order to satisfy the objective. This decision was supported by the fact that time constraints did not allow a detailed venture into complicated control. Development of an advanced control structure that takes the velocity profile, dynamics, forces, etc. into account can be considered to be another stage of the overall surgical robot project.

The basic control structure for movement control should thus allow movement tracking (by means of forward kinematics) of the joystick end effector, relating this position to the wanted position in the robot working volume as described above in Section 8.2.2, determining the necessary positions of the individual joints of the robot (by means of inverse kinematics) and finally using speed control on the motors to bring the joints to these wanted positions.

Several other requirements, not directly relating to robot movement but still affecting it, are also necessary. These include options to send the robot to a zero position, sending the robot end effector to the operating volume and putting it in "operational mode" and being able to switch the joystick brakes on and off at any time.

8.2.4   Control summary

Figure 30 explains the desired control structure and how the master and slave systems fit into the setup. This diagram shows the required work to be done by, and the desired output from, each system and how they fit together. The denotations M and S are for 'Master' and 'Slave' respectively.
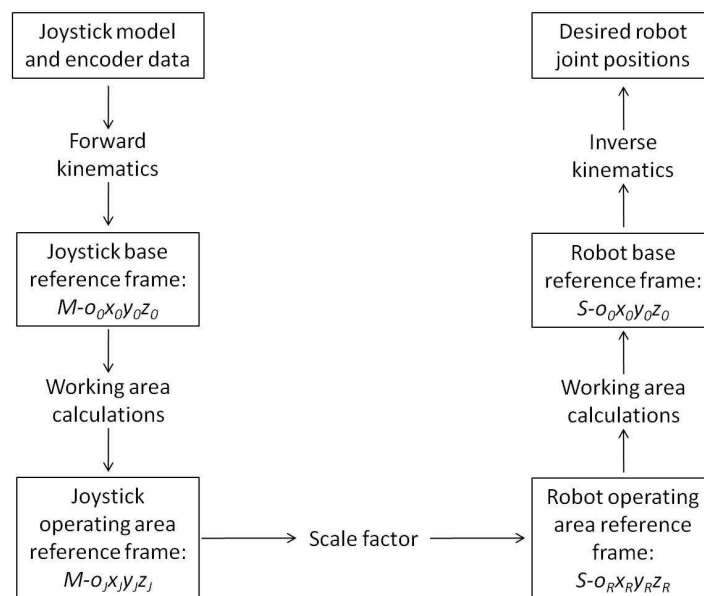
Figure 30: The master-slave control structure

On the left hand side of Figure 30, the master tasks are shown: at the end of a discrete time sample when the desired calculations have been performed, the joystick sends its end effector's position and orientation in the $M\text{-}o_J x_J y_J z_J$ reference frame to the slave side. Then, on the right hand side, the slave scales this information down, performs the necessary calculations to determine the required position and orientation of the robot end effector in the $S\text{-}o_0 x_0 y_0 z_0$ reference frame and calculates the necessary joint positions. This process provides the main control structure from master to slave. The next main step, not indicated in Figure 30, will then be to send velocity instructions to the robot motors to move to the required positions.

During the process of working with, and testing several movement options on, the PSM, the SSM and the surgical robot as a whole, many of the slave system shortcomings mentioned in Section 6.1 (especially those that could not be solved in Sections 6.2 and 6.3) caused problems. These included the extra weight and inertia added by the Parvalux and Linak motors of the SSM, the weight and inertia of the PSM assembly, the lack of robot stiffness, the limited range (due to physical constraints) of most joints and the fundamental flaws in the PSM's spherical wrist design, which caused multiple cables to break. These problems are mentioned here, because they led to several constraints being put on the control system, which resulted in the final control structure not adhering to the one specified by Figure 30.

In the sections below, the shortcomings responsible for deviations from the specified control structure will be given, but these are all subject to the main adjusting decision: in order to still be able to demonstrate control of the robot by the joystick and to satisfy the main objective despite all of the constraints, the control setup was divided into two main modes of operation. For the first operational mode all the aspects pertaining to the left hand side of Figure 30 was covered, but only position tracking of the robot end effector by moving the SSM motors was done, effectively incorporating variations of Equations 3 and 4. The orientation tracking was not executed in this operational mode, because complete functionality of the spherical wrist was needed, which was not the case. However, calculations to determine the required orientation could still be done.

The second operational mode was then used to show the ability to control those PSM joints that were in a working condition at the time of control execution. This control setup uses a DOF to DOF approach, where there is a direct relation between the movements of the respective spherical wrist joints on the joystick and their counterparts on the robot. This is where the spherical wrist design of Joints 4, 5 and 6 of the joystick is especially applicable. The effect of these two operational modes is that the first one can be used for larger movements, while the second one allows relatively smaller movements and orientation changes.

8.3   Joystick control aspects

This section describes how the joystick model and encoder data were interfaced to determine the correct encoder information for each joint, how the forward kinematics and working volume calculations were performed to obtain the desired position and orientation of the joystick's end effector in the necessary reference frames, and how the joystick operation was simulated and validated.

8.3.1   Model-encoder interfacing

To enable forward kinematic calculations to be executed correctly, the physical encoder operation had to be set up according to the joystick model. Firstly, the positive directions of rotation were checked for each joint – an increase in encoder position for the physical joystick joints had to agree with a positive rotation on the z-axes of the model's respective joints. If this was not the case,

the connections of the specific encoder's A and B quadrature lines to the Arduino board could just be swopped, which would result in a change in the positive rotation direction. The final connections of the different encoders to the Mega 2560 are shown in the pin configuration table in Appendix B.

Then, interfacing necessitated the inclusion of a zeroing operation for the joystick: if the joints are all situated in known positions (as determined according to the joystick model), a switch can be pressed to save these positions. Any resulting encoder value output from any joystick joint would then agree with the value for the same position on the model. This option was incorporated by fixing two opposing angle brackets to the side of the joystick base extension, shown in Figure 31 below.
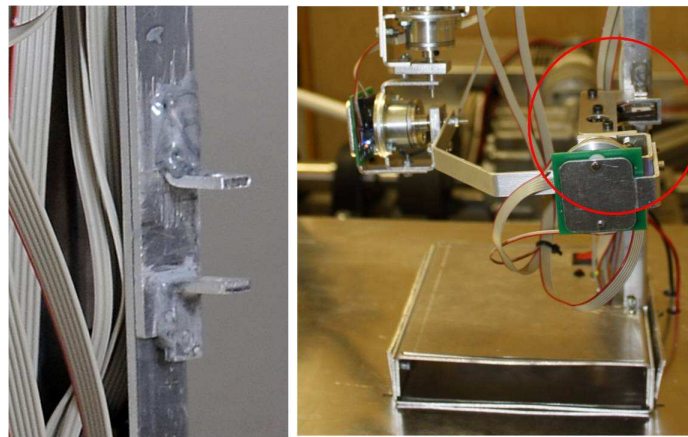
Figure 31: The joystick's zeroing brackets

The figure also shows how the gripper part fits neatly in between the brackets, ensuring that all of the joints are in a stationary and known position. These positions were determined by first orientating the respective joints of the joystick according to the model in Figure 27. These positions were then saved as the starting positions indicated in Table 8, after which the joystick gripper was positioned correctly in between the zeroing brackets. These new positions were noted as the zero positions – upon system startup, when the joystick is moved to its zero position and the safety switch is pressed for the first time, the zero positions are saved as the initial encoder values. These initial values are indicated in Table 21 in Appendix D. It has to be noted that, with 10 bit resolution, the joystick encoders output 1024 pulses per revolution, which converts to 2.844 pulses per degree. This factor was taken into account in the applicable calculations.

8.3.2 Control structure

With the model-encoder interfacing finished, the next step according to Figure 30 was to use forward kinematics to determine the position and orientation of the joystick end effector in the $M\text{-}o_0x_0y_0z_0$ reference frame, i.e. the joystick base reference frame. Equations 1 and 2 were used to set up functions for determining

the T_i and A_i matrices for each joint, with only the respective joint and link variables used as arguments. These equations are shown in Appendix D.

Then, with o_6 and R_6 in the M-$o_0x_0y_0z_0$ reference frame known, the position and orientation in the joystick working volume (which should be easily relatable to the working volume on the robot side) had to be determined. For this reason, the M-$o_Jx_Jy_Jz_J$ reference frame, indicated in Figure 32, was created.
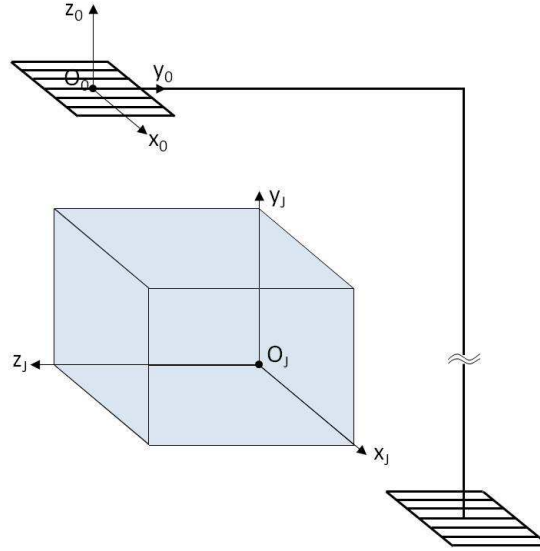
Figure 32: The joystick working volume

Figure 32 contains the joystick base frame and it indicates the orientation of M-$o_Jx_Jy_Jz_J$ with respect to this base frame. The end effector's position and orientation in the M-$o_Jx_Jy_Jz_J$ working volume are the important quantities to be sent to the robot. To calculate them, a basic rotational transformation matrix is necessary according to the equations,

$$p^j \; = \; \left(R_j^i \right)^{-1} p^i \; = \; \left(R_j^i \right)^T p^i \qquad\qquad [7]$$

and

$$R^j \; = \; \left(R_j^i \right)^{-1} R^i \, R_j^i \; = \; \left(R_j^i \right)^T R^i \, R_j^i \qquad\qquad [8]$$

Equation 7 implies that, to find the position of point p in reference frame j, the known position of p in reference frame i has to be multiplied by the transpose of the rotation matrix R that relates reference frame j to reference frame i. The same rotation matrix R is used in Equation 8 to convert the orientation with respect to reference frame i to the orientation with respect to reference frame j. For Figure 32, Equation 7 can be written as

$$p^J \; = \; \left(R_j^0 \right)^T p^0 \qquad\qquad [9]$$

where

$$\left(R_j^0 \right)^T \; = \; \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}. \qquad\qquad [10]$$

Equation 9, for determining a position, is valid when the origins of the base and working volume reference frames are located at the same point. Naturally, when this is not the case, the offset between the two origins has to be taken into account, but the rotational relation still plays an important role. The exact equations that are valid for any case are shown in Appendix D.

The final part of the joystick control structure, after the forward kinematics and the working volume calculations, is the safety aspect that had to be incorporated to satisfy control specifications. The requirement is that the robot end effector should only move when the safety switch on the joystick gripper is pressed; when it is released, nothing should happen. This implies that, although the joystick may move around and the position and orientation may change while the switch is released, once the switch is pressed again the end effector must carry out its instructed movement from that previous stationary position. To account for this, the M-$o_J x_J y_J z_J$ working volume was designed to be stationary with regards to the joystick base reference frame while the switch is pressed and fixed to the end effector position frame's origin (o_6) while the switch is released. In this way, no movement occurs with respect to the M-$o_J x_J y_J z_J$ working volume when the switch is released.

8.3.3   Simulation

To determine whether the modelling, encoder values, forward kinematics and working volume calculations were correctly executed and to serve as an error detection tool during the course of the control system design process, custom simulation software was created. This entailed creating a MATLAB Graphical user interface (GUI) that could display the position of the joystick joints (according to the encoder information) in real time. Figure 33 shows the general form of the GUI.

It has options to choose and connect to a computer's serial port, to start sampling data and to draw the joystick links in real time. For correct functioning, the joystick's Mega 2560 has to be connected to a computer via a Universal Serial Bus (USB) cable. The GUI takes the raw encoder data of the joystick's six joints as inputs and uses this information, together with forward kinematic calculations, to draw the simulated joystick. This GUI was tested successfully when the drawn joints showed the same positions and orientations as those of the physical joystick.

The GUI was also used to test if the mathematics and control for the safety switch were incorporated correctly. By outputting and checking the position of the joystick in the M-$o_J x_J y_J z_J$ working volume, the values could be seen to change constantly while the safety switch was pressed and to stay constant while it was released. These tests showed that the requirements of incorporating the safety switch and the joystick working volume had been satisfied. The completion of the remaining specifications, mostly referring to the aspects on the right hand side of Figure 30, is discussed as part of the robot control aspects.
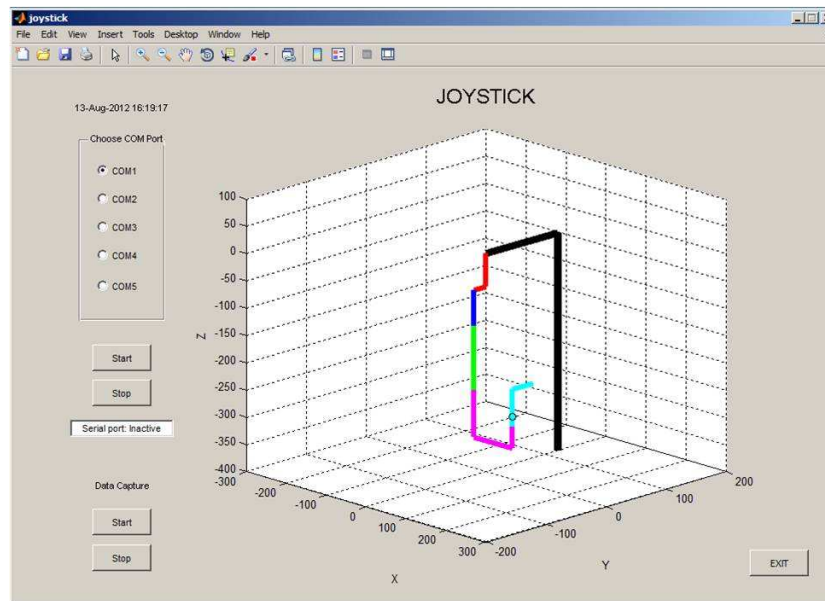
**Figure 33: The joystick simulation GUI**

8.4   Robot control aspects

This section describes how the robot model and its encoder data were interfaced to get the correct encoder information for each joint, how the working volume calculations were performed (based on information from the master) to get the desired position and orientation of the robot end effector in the robot's base reference frame, how the inverse kinematics were done to determine the required joint positions and how control was executed on the robot's motors to enable movement. The simulation tool was adjusted to include the robot as well.

8.4.1   Model-encoder interfacing

As was done for the joystick, the robot's physical encoder operation firstly had to be set up according to its model. The same method was used for detecting and adjusting the positive rotation axes of the joints, with the final connections again indicated in the pin configuration table in Appendix B. The SSM motor and PSM motors (where possible) were then actuated in order to orientate the robot according to the model, with the joint variables being equal to the values shown in Table 10. These angles were converted to the appropriate encoder values, depending on the type of encoder, by using the conversion factors of Table 22 in Appendix D.

Section 6.3.1 documented the addition of zeroing switches to the SSM system. Thus, for the encoder interfacing, the three SSM motors were moved from their model positions to the zero positions. Here, the new zero position encoder values were saved so that the SSM can be zeroed at each system startup in order to have known positions for all three SSM motors. Unfortunately, the PSM did not contain any zeroing methods and the addition of switches was improbable due to

60

the small scale and timing constraints. However, the decision was made to have the motors start in their known and saved model positions and then to return the motors to these positions before the system is switched off. This poses a problem if an unexpected power outage occurs, but except for that the correct encoder values could be calculated.

8.4.2   SSM control structure: Operational mode 1

Here, the emphasis is on the control structure for the first operational mode as described in Section 8.2.4, however still concentrating on the aspects on the right hand side of Figure 30. Firstly, the working volume calculations had to be done. The outputs from the joystick working volume calculations were the end effector's position and orientation in the $M\text{-}o_J x_J y_J z_J$ reference frame. The position variables then had to be scaled to account for the smaller movements at the robot end. According to literature (Lobontiu & Loisance, 2007) a size ratio of 3:1 (joystick volume to robot volume) was chosen. The robot working volume was then created as illustrated in Figure 34, with the x, y and z axes pointing in the same directions as those of the joystick reference frame.
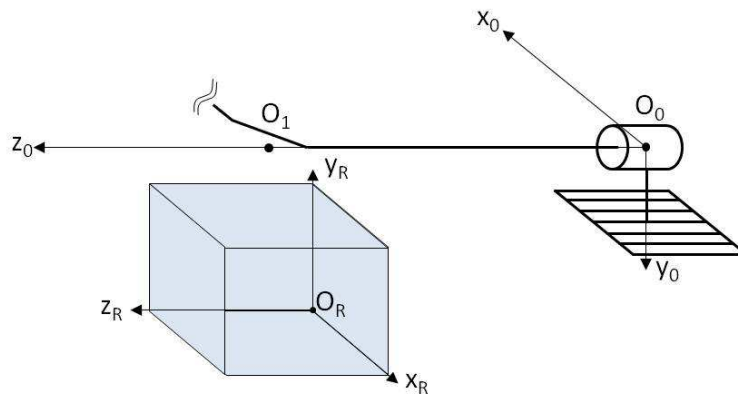
Figure 34: The robot working volume

As can be seen, the working volume is situated directly beneath o_1, the stationary pivot point of the robot. For the best operation, optimal distances between o_0 and o_R (in each dimension) were determined through a process of trial and error; they are indicated in Table 11 below. The three entities are defined in Appendix D.

Table 11: The working volume position values

Entity:	X_{r0}	Y_{r0}	Z_{r0}
Distance (mm):	40	153	610

After scaling, the required position and orientation of the robot end effector in the $S\text{-}o_R x_R y_R z_R$ reference frame was known. Then, to determine these quantities with respect to the robot's base frame, the same theory was used as for Equations 7 and 8, but with some variations. The equation for determining the position of a point known in $S\text{-}o_R x_R y_R z_R$, with respect to $S\text{-}o_0 x_0 y_0 z_0$, is

$$p^0 = (R_0^R)^T p^R \qquad\qquad [11]$$

where

$$(R_0^R)^T = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \qquad\qquad [12]$$

Again, this matrix is valid for the rotational relation as well. Then, with the required values known with respect to the *S-o₀x₀y₀z₀* reference frame, inverse kinematics (as described in Section 8.1) could be incorporated to determine the joint variables. As was mentioned previously, this operational mode is responsible for controlling the SSM motors in order to move the robot end effector to the required position. For this reason, inverse kinematics was only applied to determine the required positions of the SSM motors.

Without the orientation tracking, the required rotation matrix *M-R* (as specified by the joystick forward kinematics) was not necessary or used in this case, which implied that the wrist centre o_C could not be calculated with Equation 3 or 4. Rather, the inverse kinematics process was applied directly to the required end effector position in the *S-o₀x₀y₀z₀* reference frame. The detailed equations can be seen in Appendix D. The result of this process was having reference positions for all three SSM motors, which could then be compared to current positions of the motors and consequently allowed speed control to be executed.

Several problems hindered effective speed control of the motors, with the most prominent being the mass of the two Parvalux motors and the PSM exerting excessive torque and forces on the SSM joints. The result was that the Parvalux motors could only be moved once the opposing forces were overcame; this corresponded to a certain PWM duty cycle, typically around 8 %. Between zero and 8 %, the speed did not increase linearly and the joint was stationary. The joints only moved when enough torque was exerted (above 8 %) and this was at a relatively high speed. This limited how well control could be executed when small and slow movements were necessary. A trial and error process was used to find an algorithm where speed increases linearly with the error between reference and actual encoder values. It showed acceptable movement results.

If errors were registered at the SSM motors, the above movement algorithm would cause these motors to be actuated even if no movement was measured at the joystick end. This would lead to master-slave control complications and necessitated the inclusion of additional controlling methods: stopping all three SSM motors when no movement was picked up at the joystick side and stopping the SSM motors if their respective error values were below a certain level.

The final control aspects of Operational mode 1 pertained to safety measures and the physical constraints of the SSM motors. As mentioned in the safety specifications section, control had to ensure that the robot stays within the boundaries created by the physical joint constraints. The constraint values for the SSM motors, mentioned in Section 2.3.2, were noted and algorithms were created to stop the motors before any of these positions were reached.

8.4.3   PSM control structure: Operational mode 2

In this section the emphasis is on the control structure for the second operational mode as described in Section 8.2.4. The original idea was to use the PSM joints to provide the orientation of the robot end effector, according to Equations 4, 5 and 6. However, with the slave system shortcomings as they were, this could not be done. The decision was then made to use a DOF to DOF approach, where there is a direct relation between the movements of the respective spherical wrist joints on the joystick and their counterparts on the robot

The control structure for this type of movement was quite simple, with Joints 4, 5 and 6 on the joystick providing reference values that their counterparts on the robot had to follow. Here, the error between the reference and actual encoder values was not an important indicator of the necessary speed control, due to the fact that all the PSM motors were run at 100% duty cycle. This was because the large gear ratios of the PSM motors limited their maximum rotational speed to a very low value – anything lower would not be sensible to use. As was done for the SSM motors, the physical constraints were incorporated for the PSM motors as well. Software algorithms stopped the applicable motors if they neared the physical constraints agreeing with the values mentioned in Table 2 in Chapter 2.

During testing of this operational mode the design flaws of the PSM caused several problems. The joints of the spherical wrist had too great an effect on each other, e.g. if Joint 6 was actuated with the gripper closed the tension in the gripper cable increased greatly, causing the cable to break. There was also no way of determining the tension in the cables; an option was to use the current through the motors (indicating the amount of torque exerted), but the gear ratio of 1526:1 removed any linear relationship between current and torque. These problems caused the gripper and all the spherical wrist joint cables (except for Joint 4, which operated with a gear) to break. At the testing stage, only Joints 4 and 6 were functional and time constraints did not allow the others to be fixed.

Due to these problems it was decided that the second operational mode could be demonstrated by relating the movement of Joint 6 on the joystick to Joint 4 on the robot. This could show successful execution of the control structure for the second operational mode and, if the other PSM joints were functional, the same method could be incorporated in precisely the same way.

8.4.4   Simulation

The MATLAB GUI that was created for the joystick was extended to include a robot simulation option. With this new GUI the joystick encoder information could be sampled and the forward kinematics could be calculated to draw the joystick in real time. The working volume scaling could be done to determine the required robot end effector position and orientation, after which the inverse kinematics calculations could be executed to draw the simulated robot in real time. It provided an effective way to check calculations and test control options during the course of designing the two operational modes. It was also used as a safety

63

precaution to test if control structures were satisfactory before actually applying it to the physical robot. Figure 35 below shows a representation of the new GUI.
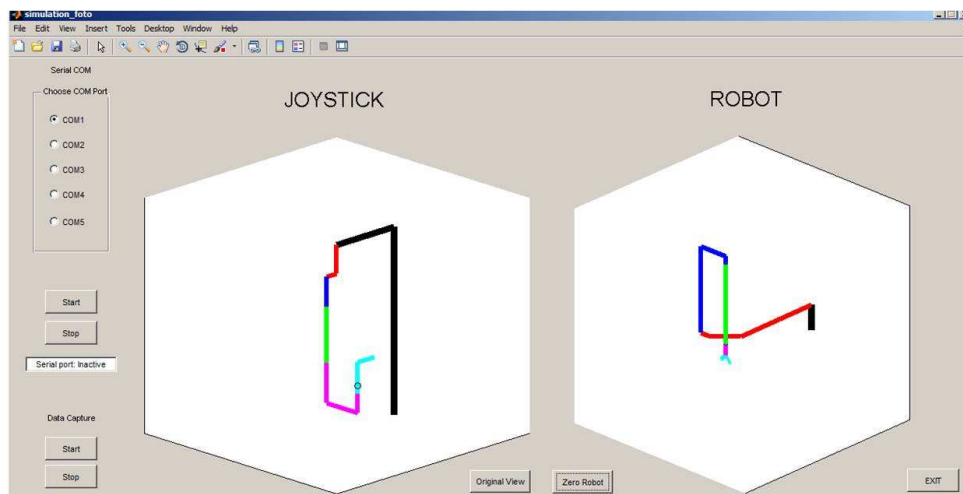
Figure 35: The final simulation GUI

With the SSM and PSM not functioning as required, the desired control system from master to slave could not be demonstrated on the robot. In this case, the GUI was of great help, because it could be used to demonstrate the required control structure according to the specifications of Figure 30 in Section 8.2.4. The new aspects of the GUI contain equations to incorporate the scaling factor and determine the $S\text{-}o_R x_R y_R z_R$ reference frame. From the joystick's forward kinematics equations it then finds the robot end effector's required position and orientation in this reference frame and consequently in the robot's base reference frame, $S\text{-}o_0 x_0 y_0 z_0$. For this, Equations 7 through 12 were incorporated. The next step was then to use kinematic decoupling and inverse kinematics (Equations 4, 5 and 6) to determine the required joint positions of the robot's spherical wrist. Once these variables were known, the robot could be drawn in real time. All equations to support these calculations are shown in Appendix D.

The GUI was used to effectively demonstrate control of the slave system by the joystick, showing that the objective of controlling the seven DOF robot was satisfied. If it were not for the problems with the SSM and PSM this control system could have been demonstrated physically as well.

8.5 Final operation

In this section a basic overview of the final operation setup is given. This information was sourced from the user's manual that was created to provide a full and detailed discussion of all the aspect necessary to operate and maintain the joystick-robot setup.

During operation, the master-slave setup functions autonomously without inputs from a computer: the two Arduino Mega 2560 control boards provide all the control and monitoring that is needed. At system start-up the power supply is

switched on, providing a 24 V (DC) supply voltage to the master and slave systems. There are two emergency stop buttons: one is part of the master setup and regulates voltage supply to the whole master-slave system, while the other one is part of the slave setup and only regulates power supply to the slave system. They can be pressed at any time to stop any activity.

When the power supply is switched on and both emergency stop buttons are released, the joystick's own power switch can be turned on, which provides power to the encoders, brakes and switches. The first task is then to zero the joystick encoders, as the joystick control board waits for this to happen before entering its main execution loop. The joystick gripper can be pushed into the bracket gap and held steady, after which the gripper safety switch is pressed. The joystick's red warning LED will flash twice, indicating that the zeroing method has been completed successfully.

The joystick is mounted on the joystick plate, which is in turn mounted on the robot base structure. This joystick plate also serves as a mounting place for several switches that are part of the master controls. Three of these switches, situated next to the joystick on the joystick plate, are important for master-slave control.

After the joystick is zeroed, the "Zero robot" switch can be used to send the SSM motors to their starting positions. Then the "Mode 1" switch can be turned on, which results in the SSM motors moving to predefined positions (-90°, 90° and 610 mm for Joints 1, 2 and 3 respectively, according to the robot model) from where the joystick can take over control. These values were chosen so as to position the robot end effector within the $S\text{-}o_Rx_Ry_Rz_R$ reference frame. In operational mode 1, the joystick can be moved but the movement is only executed by the robot if the safety switch on the joystick gripper is pressed. Any time during this mode, the "Mode 1" switch can be turned off and the "Mode 2" switch can be turned on. This will cause the three SSM motors to be stopped and the Joint 1, 2 and 3 brakes on the joystick to activate, which will make the second mode operational. In this mode, provided that the PSM is functional, small movements are possible and the PSM joints are controlled directly by the corresponding joints on the joystick.

The joystick plate contains six switches, each of which can be used to activate a certain joint's brake on the joystick (see Figure 23). In operational mode 2 these switches can be activated to stop movement in any DOF of the robot. To switch back to operational mode 1, the "Mode 2" switch can be turned off and the "Mode 1" switch can be turned on again.

At the end of the process, the robot can be zeroed again, the joystick can be turned off and the emergency stop button can stop all power supply to the master-slave system.

CHAPTER 9   SOFTWARE

Satisfying the thesis objective of controlling the robot end effector with the joystick went hand in hand with creating the necessary software. This chapter describes the software that was developed during this control structure development stage. Both Mega 2560 control boards needed extensive programming to be able to execute control and the simulation GUI was also based on MATLAB code. These three bases form the major divisions under which the software is described. Pseudo-code flow diagrams to indicate program execution are provided, the main functions are described and any libraries that were used are discussed.

9.1   Master software

The Arduino Mega 2560 that is part of the joystick system had to be programmed in order to perform all of its tasks and interface with the master system inputs and outputs (see Figure 19, Section 5.1). Programming was done with the Arduino integrated development environment (IDE) (*Arduino Software,* [S.a.]) and the Mega 2560 was loaded with the final program.

9.1.1   Libraries

Several Arduino libraries were needed in the master program to allow the necessary calculations. To enable quadrature encoder readout, the open-source *Encoder* library (*Aduino Encoder*, [S.a.]) was used. This library counts all four edges of a single cycle on a quadrature encoder line, allowing 10 bit resolution. For optimal use, one quadrature line of an encoder must be connected to an external interrupt pin on the Mega 2560, while the other can be connected to a digital input pin. When an *Encoder* object (e.g. "*enc1*") is declared using this library, a value can be read from or written to that encoder at any time by using functions *enc1.read()* and *enc1.write(value)* respectively.

To allow matrix manipulation, the open-source *MatrixMath* library was used (*Aduino MatrixMath*, [S.a.]). With this library, matrix multiplication was made possible, a task that was essential in the forward kinematic calculations. To calculate the matrices and send the necessary data to the robot at a constant rate, the open-source *Timer* library (*Aduino Timer*, [S.a.]) was used. The *Timer* library has several functions, but the one that was incorporated here calls a specified function at a constant rate supplied by the user. For correct functioning, the *Timer* object is updated during each execution of the program's main loop.

Finally, for data transfer purposes, the *EasyTransfer* library (*Aduino EasyTransfer*, [S.a.]) was used to transfer data over the Serial3 ports of both control boards. An identical structure, containing the same variable names and types, is declared in the program of each control board. The *sendData()* function can be used to send the values of all variables simultaneously, while the

66

receiveData() function is used to check if the variables have been received on the other end.

9.1.2   Program flow

Figure 36 below shows a simplified version of the main program flow.
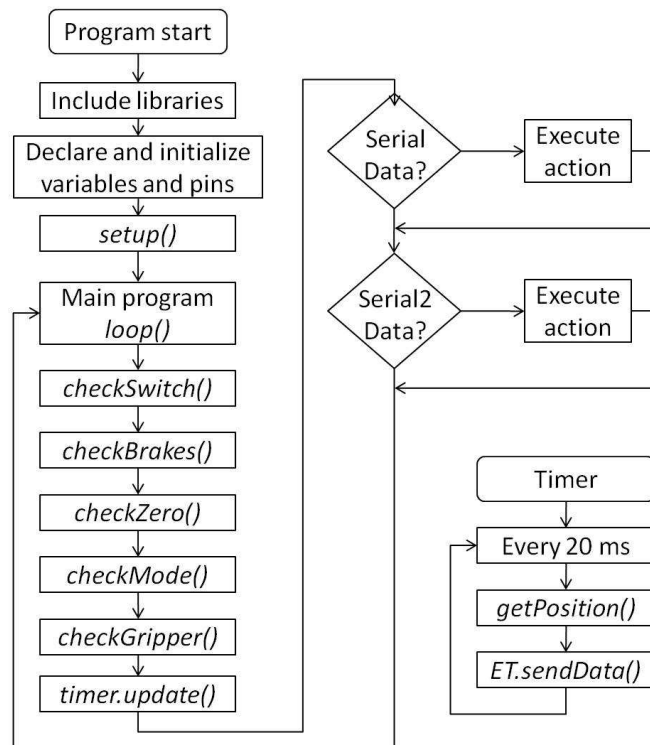
Figure 36: The master program flow diagram

When the program has started, the first action is to include all the necessary libraries. After this, the Mega 2560 pin numbers and names are declared and the variables pertaining to the main program are declared and initialized. Objects for the different libraries are also created; this includes *Encoder* objects for the six joystick encoders, a *Timer* object, a *MatrixMath* object and a variable structure to be used by the *EasyTransfer* library. This is followed by the *setup()* function, which is executed once before the main program starts.

The main loop is executed continuously while the Mega 2560 receives power. The *checkSwitch(), checkBrakes(), checkZero(), checkMode(), checkGripper()* and *timer.update()* functions are executed, after which the Serial and Serial2 ports are checked to see if data is available. Based on the data inputs, certain actions can then be executed.

An alternate control loop is also executed in the background. Based on the *Timer* output, the *getPosition()* and *ET.sendData()* functions are called every 20 ms. The *getPosition()* function is responsible for using the joystick encoder data to

67

calculate the required position and orientation of the robot end effector and to use the *ET.sendData()* function to send this data to the robot. The execution time for *getPosition()* was checked during program operation and it varied between 8 and 10 ms. A control loop of 20 ms was therefore deemed safe, thus preventing the next iteration from occurring while the function was still busy executing.

9.1.3 Main functions

The first main master program function is the *setup()* function. It is responsible for several initialisations and declarations that have to be in place when the main program loop starts executing.

Firstly, the Serial, Serial2 and Serial3 ports are all initialised at a baud rate of 9600 bits/s, the rate at which communication between the master and slave control boards, and between the control boards and the computer, was successfully tested. The Serial port is used for data transfer between the Mega 2560 and a computer; this port was implemented during testing and when using the MATLAB simulation GUI. The Serial2 and Serial3 ports are both used for master-slave data transfer: Serial3 is used with the *EasyTransfer* library to send data points continuously, while the Serial2 port is used for event based data transfer like button pushes.

The Mega 2560 pin modes are then set, e.g. encoder pins are set as input pins, while the brake pins are set as output pins. This is followed by the function *matrix_init()*, which is responsible for initialising the values of all matrices used in the forward kinematics calculations. The warning LED on the joystick assembly is then flashed to indicate the main initialisation is complete. After this, the program waits for an input by the user before it carries on with execution. This is when the joystick has to be put in its zeroing brackets and the safety switch has to be pressed. After this occurrence, the warning LED is flashed twice to indicate successful zeroing. The *setup()* function then ends when the *EasyTransfer* method is linked with Serial3 and when the *Timer* object is set and started.

The functions that form the bulk of the main program loop are as follows: *checkSwitch()* checks the state of the safety switch on the joystick gripper and executes the appropriate actions; *checkBrakes()* checks if any of the brake toggle switches on the joystick plates have been activated and it switches the appropriate brakes on or off; *checkZero()* looks if the zero toggle switch has been activated and, if so, it zeroes the robot; *checkMode()* monitors the operational mode toggle switches and executes the appropriate action based on the selected mode; *checkGripper()* checks the state of the gripper buttons to see if the gripper should be actuated; and *timer.update()* is necessary for correct functioning of the *Timer* object.

The final main function is *getPosition()*, which is executed every 20 ms. It uses encoder data, the *MatrixMath* library and forward kinematic equations to determine the position and orientation of the joystick's end effector with respect to the $M-o_J x_J y_J z_J$ reference frame, which it then sends to the slave side by using the *ET.sendData()* function.

9.2   Slave software

The Arduino Mega 2560 that is part of the robot system had to be programmed in order to perform all of its tasks and interface with the slave system inputs and outputs (see Figure 26, Section 6.3.2). Programming was done with the Arduino IDE (*Arduino Software,* [S.a.]) and the Mega 2560 was loaded with the final program.

9.2.1   Program flow

The slave software uses the same libraries as those specified for the master software. They will therefore not be explained again. Figure 37 below shows a simplified version of the main program flow.
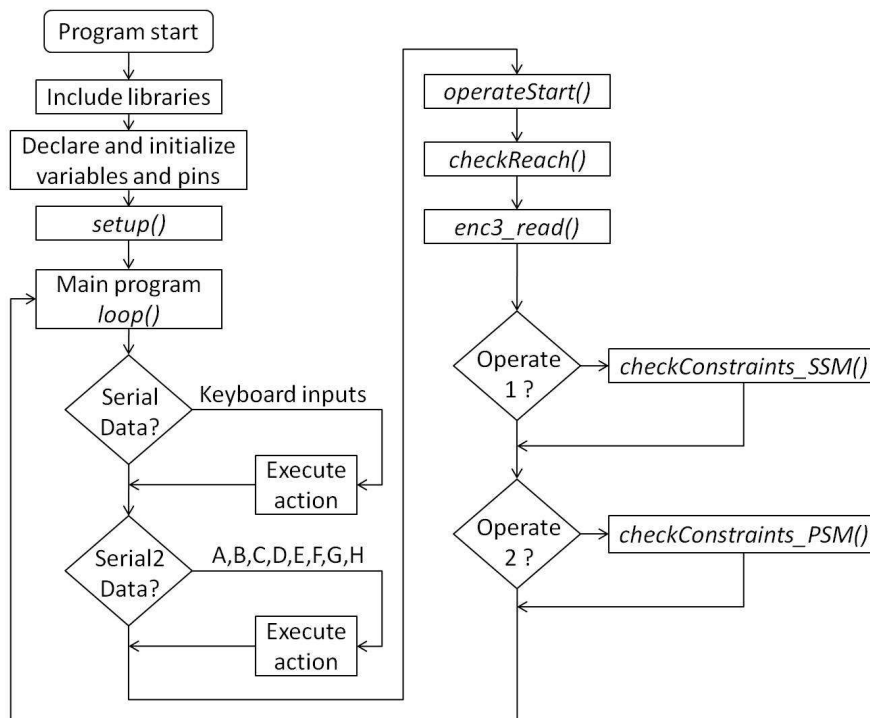
Figure 37: The slave program flow diagram

At program start-up the libraries are included, the Mega 2560 pin numbers and names are declared and the variables pertaining to the main program are declared and initialized. Objects are also created, including *Encoder* objects for the seven SSM and PSM encoders, a *MatrixMath* object and a variable structure to be used by the *EasyTransfer* library. The *setup()* function is executed once before the main program starts.

The main loop is executed continuously. Firstly the Serial and Serial2 ports are checked to see if data is available – based on the received data, the appropriate actions are executed. This is followed by functions *operateStart()*, *checkReach()* and *enc3_read()*. Finally, *checkConstraints_SSM()* or *checkConstraints_PSM()* is executed based on whether the robot is in the first or second operational mode.

9.2.2　　　　　Main functions

The first main function of the slave program is the *setup()* function. As for the case of the master *setup()* function, it is responsible here for several initialisations and declarations that have to be in place when the main program loop starts executing. The Serial, Serial2 and Serial3 ports are initialised at the same baud rates and for the same data transfer purposes as for the ports of the master control board. The input/output pin modes are also set and initialised.

In the main program loop, the first two main functions check to see if data is available on ports Serial and Serial2. The Serial port is used for data transfer between the Mega 2560 and a computer, typically for testing purposes and simulation GUI interaction. The Serial2 port receives instructions, in the form of characters, from the master. Table 12 below indicates the different instructions and the applicable actions that are taken.

Table 12: Slave functions based on master instructions

Character	Action
A	Safety ON
B	Safety OFF
C	Gripper ON
D	Gripper OFF
E	*control()*
F	*zero_func()*
G	*operate1_set()*
H	*operate2_set()*

Actions based on instructions from the master include setting the safety switch state based on whether the switch on the joystick is pressed, setting the gripper state based on whether the gripper buttons are pressed, calling the *control()* function to execute control, calling the *zero_func()* to zero the SSM motors and calling functions *operate1_set()* an *operate2_set()* based on the applicable operational mode that is active.

Every 20 ms when the master sends its end effector data to the slave, an "E" character is received on Serial2 and *control()* is called, which entails several steps. Firstly, the *ET.receiveData()* function is used set all the variable values, after which function *getAngles()* uses inverse kinematic equations to calculate the necessary positions of the SSM motors. Then, depending on the operating mode, function *move_SSM()* or *move_PSM()* is called to move the specific motors to their required reference positions and to stop any unnecessary movement.

Next, the final main functions of the main iteration loop are executed. The function *operateStart()* is used to check if the first "Move to operational mode 1"-

instruction has been received from the master. If this is the case, the SSM motors are moved from their zero positions to the predefined positions for the first operational mode. Then, *checkReach()* is just a safety precaution that stops the SSM motors if the zero positions are reached during operation execution. As mentioned in Section 6.4, the six external interrupt pins on the slave Mega 2560 were not enough for the seven encoders of the surgical robot. An extra function, *enc3_read()*, was then created for quadrature encoder readout of the linear motor encoder. Finally, the *checkConstraints_SSM()* and *checkConstraints_PSM()* functions are used to check whether the boundaries of the SSM and PSM motors have been reached and to stop the applicable motors if this is the case.

9.3   Simulation GUI

As Chapter 8 demonstrated, a MATLAB GUI was used successfully to incorporate joystick encoder data, to execute all control calculations and to simulate movement of the seven DOF surgical robot. It was created in MATLAB's GUIDE environment, a user-friendly IDE for the specific design of GUIs.

Functions for this software are linked to specific buttons on the GUI and are only executed if and when these buttons are pressed. There is therefore not a very specific program flow, although Figure 38 indicates the order in which instructions are given to the GUI (i.e. which buttons are pressed), as well as the resulting tasks that are executed.
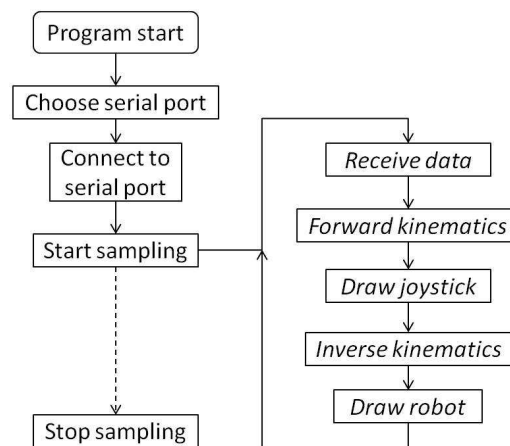
Figure 38: The GUI program flow diagram

When the serial port is connected and data sampling is started, MATLAB reads data (mainly the six encoder positions, set up according to the model) from the serial port in string format and converts it to the necessary variable types. These variables are used in forward kinematic equations, working volume calculations and finally inverse kinematic equations to determine the required robot joint positions. To draw the joystick and robot models, the *line* command is used to draw a specific link between its two binding reference frame origins. The loop that is executed once the "Start sampling" button is pressed occurs every 20 ms, based on the *Timer* function of the master Mega 2560. The models are therefore drawn at a constant sampling rate of 50 Hz. To stop the simulation, the "Stop sampling" button can be pressed and the GUI can be closed.

CHAPTER 10   TESTING

Chapters 7, 8 and 9 were all concerned with satisfying the objective of controlling the movement of the surgical robot. In order to validate the work that was performed in these sections – and also to determine whether the joystick and the robot adhere to the main requirements that were set in terms of safety, intuitiveness, movement, etc. – several tests had to be conducted. This chapter describes these tests.

Firstly, the main outcomes of this chapter, based on the system specifications, are described. This is followed by a description of the experimental setup and the applicable tests that were done. Lastly, the results of the tests are discussed and evaluated based on the desired outcomes.

10.1   Required testing outcomes

Throughout the course of the research, several outcomes were specified for the user interface due to the absence of detailed, client-specified requirements. These outcomes were mainly specified in terms of movement control, intuitiveness and safety and are elaborated on below. Although being the important aspects for which the testing section was intended, another important outcome is the technical description of the product, for which several tests were also conducted.

10.1.1   Movement control

According to one of the thesis objectives, the joystick should be able to control the movement of the seven DOF surgical robot. An important point to note is that this thesis concentrated on the user interface and not the surgical robot, implying that the movement accuracy already had to be set and specified for the robot. When adding the fact that several problems were evident with the PSM's and SSM's functionality, the user interface could not hope to improve the accuracy or resolution of the surgical robot's movement. Also, the presence of visual feedback implied that, if the robot did not initially execute the desired movement as provided by the operator, the error could merely be corrected by another input from the operator based on the visual position difference.

When considering these factors, it was evident that traditional testing methods of providing a reference input to the robot and measuring the actual output (and therefore the resulting error and accuracy) would not be applicable. It was decided that, in order to concentrate on how the robot follows the joystick's instructions, a test would be conducted by determining how well the robot end effector could be moved from point to point. The time it takes for this movement could also be documented, which could indicate the effectiveness of the control system.

The drawbacks of the slave system prevented high-resolution movement from being tested, for example in the setup of suture application. But the point to point test could show the successful control of the robot by the joystick, which would satisfy the main thesis objective. The time aspect could also indicate the degree to which successful control is executed.

10.1.2   Intuitiveness

The requirement of the joystick having an intuitive design stems from the need to decrease the complexity of the product and the time it takes for a new operator to learn how to use the joystick. Ultimately, a new operator should just receive a short introduction to using the product and execute a few practice runs, after which he/she should have enough experience to control the robot's movement. The intuitiveness of the joystick should simplify this process greatly.

To test for this, different subjects could be used as test cases. New operators can be taught how the joystick operates and what the best way of executing control is, after which they can also execute point to point tests. The desired outcome would be minimisation of the time necessary for the test subject to move the robot between points. Feedback from the subjects could also be useful in determining features that are positive or negative in the joystick's design.

10.1.3   Safety

Chapter 5 introduced several electronic safety features that were included in the joystick's design. These are mainly the safety switch on the joystick's gripper, the electromagnetic brakes included in the joystick's joints and the emergency stop button that controls the power to the whole master-slave setup. Chapters 6 and 9 also discussed the SSM and PSM safety features where software was used to keep movement of the robot joints within specified boundaries determined by their physical constraints.

The outcomes for safety testing would be to test the functionality of these safety aspects during point to point testing. User feedback on these features would also be valuable.

10.1.4   Technical specification

The final outcome of the testing section would be to determine a technical specification of the master-slave system. The joystick and robot can be tested for voltage supply, current rating, operating volume and similar technical aspects. This data can then be used to create a technical specification sheet for the product.

10.2   Testing setup and execution

The main objective of this test was to create an experimental setup that would serve as a basis for conducting the tests and achieving the desired outcomes mentioned above. The same setup was used for testing the different aspects.

For point to point movement tracking, a simple PCB containing two tactile switches (each with their own indicator LED) was designed and constructed. The objective is then to move the robot end effector from the one switch to the other and to document the time it takes for this movement. With a 5 V supply, the LEDs will light up when the switches are pressed. This PCB is shown in Figure 39 (the circuit design can be viewed in Appendix B).
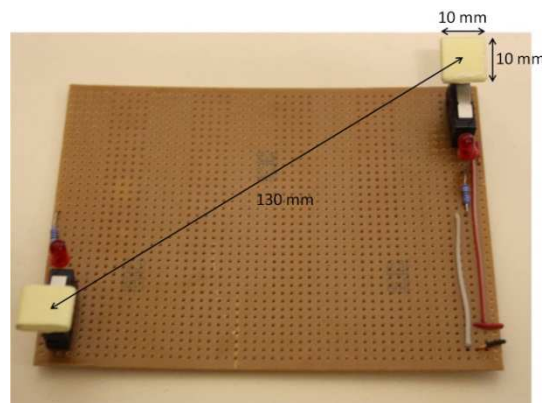
Figure 39: The point to point testing board

This PCB was placed inside the robot working volume so that both switches (with their 10 mm x 10 mm pads) would be reachable by the robot end effector. The following testing procedure was then executed for the set of movement control tests:

- The master-slave system was switched on.
- The joystick was switched on and zeroed and then the robot was zeroed.
- The robot was moved to its initial operational mode 1 starting position, as indicated in Figure 40 (a).
- The operator used the joystick to move the robot end effector 7 cm to the first and nearest switch, as shown in Figure 40 (b). The movement time from start to the point where the switch was pressed, Time_1, was documented when the LED flashed.
- The operator used the joystick to move the robot end effector 13 cm to the second switch, as shown in Figure 40 (c). The movement time from the previous switch to the point where the second switch was pressed, Time_2, was documented when the second LED flashed.
- The robot was zeroed and the master-slave system was switched off.

To prevent consecutive tests from influencing one another, the whole master-slave system was restarted and zeroed between each test. The above procedure was executed 20 times by an expert (the designer of the joystick), which provided enough data on which to base the results of the movement control tests.
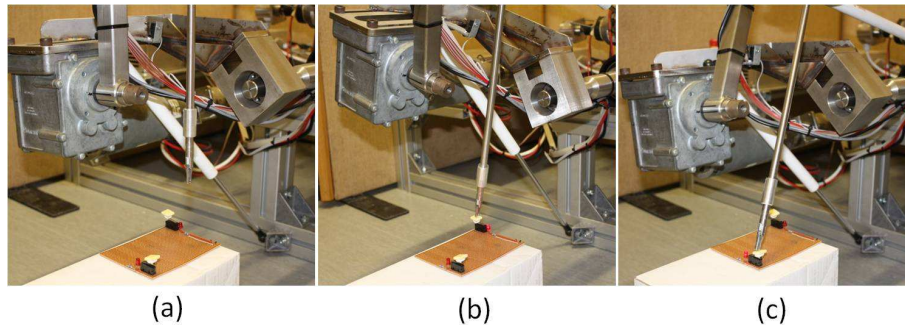
Figure 40: Point to point testing execution

For intuitiveness testing, four random subjects that have no experience with this system or similar master-slave controllers were asked to execute the same test as above. Before each set of tests, the subjects were given a 5-minute introduction on the functioning of the joystick and the specifics of movement control, which included the working volume orientation, the safety switch functionality and the SSM boundary constraints. They were then given a 5-minute practicing period to try out the joystick and to get accustomed to how it operates. After this, a set of five tests were conducted by each subject, using the same method as explained above and again documenting Time_1 and Time_2 as above. After the tests, the subjects gave feedback in terms of intuitiveness, positive aspects and possible hindrances.

The safety testing was fairly straightforward: during movement control testing exercises the functionality of the safety switch, the emergency stop and the boundary constraints was tested. This was done by pressing the switch/button or moving toward an applicable boundary and checking the response in each case. The technical specification testing was done similarly, with no specific test setup necessary. During operation, voltages and currents were measured by using a standard multi-meter or a bench power supply, while other specifications were derived from datasheets and the PSM and SSM constraints.

10.3   Testing results

The times (Time_1 and Time_2) that were documented for the set of 20 movement control tests are shown below in Figure 41, while the data for the four sets of intuitiveness subject tests are given in Figure 42 and Figure 43. The raw data can be viewed in Table 23 through Table 27 in Appendix E.

For the movement control tests, the average for Time_1 was 7.93 s (indicated by the blue line in Figure 41) and the average for Time_2 was 14.15 s (indicated by the red line in Figure 41). When considering the distances that the end effector was moved (7 cm and 13 cm respectively), this amounted to a movement speed of about 1 mm/s. The actual speed can be considered to be even higher due to the fact that the tests were usually not executed in one smooth movement, but rather with several 'stop-go' efforts. This was because movement errors, derived from visual feedback, had to be accounted for. These errors were ascribed to the limited functionality of the SSM motors, but they did not result in unsuccessful

testing. The two testing board switches could be pressed successfully, within the indicated times, for each of the 20 tests. This showed that the joystick could successfully control the movement of the robot between two specified points and within relatively fast times.
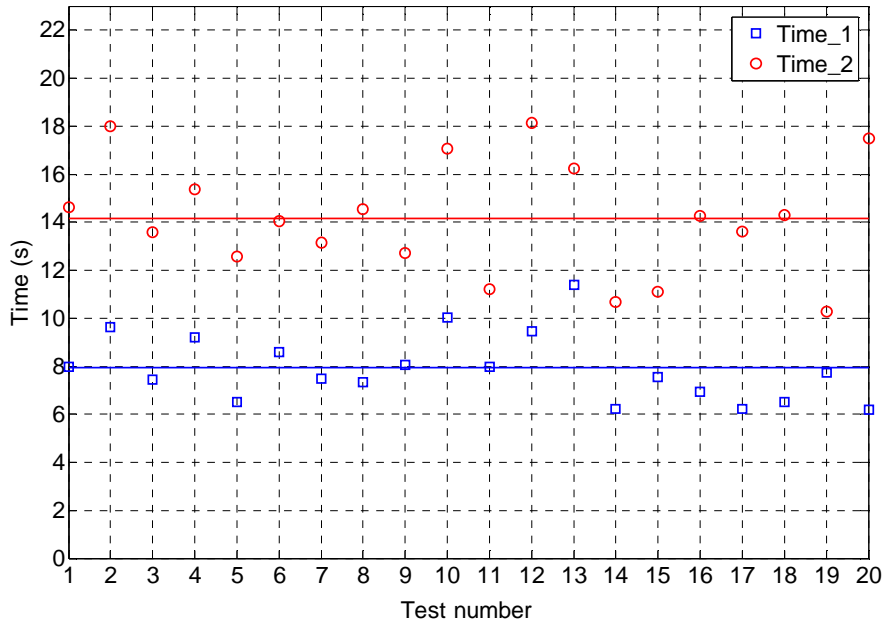
Figure 41: Time_1 and Time_2 data for the 20 movement control tests
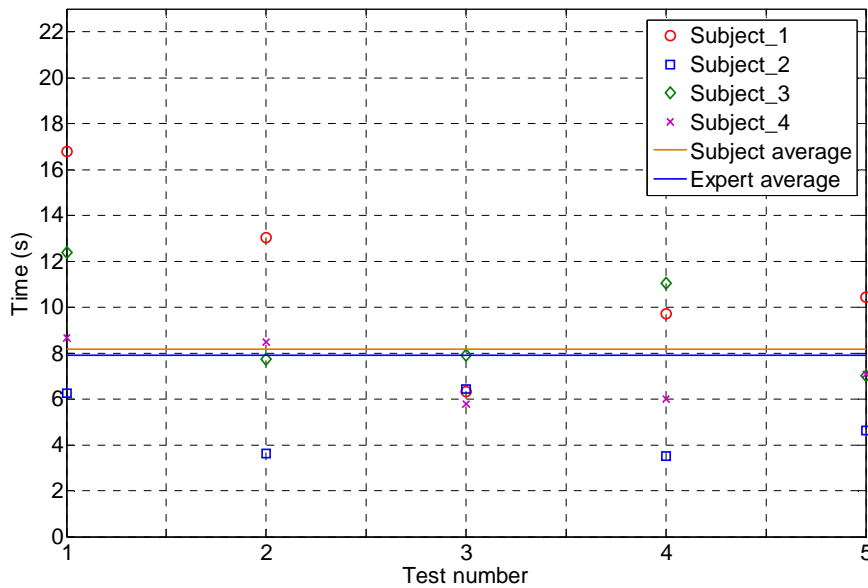
Figure 42: Time_1 data for the five intuitiveness tests conducted by four subjects

Figure 42 shows the Time_1 data for the four sets of intuitiveness subject tests, as well as the subject average of 8.15 s and the expert average for Time_1 taken from Figure 41. Similarly, Figure 43 shows the Time_2 data for the four sets of intuitiveness subject tests, as well as the subject average of 15.6 s and the

76

expert average for Time_2 taken from Figure 41. The subject averages are given in Table 28 (Appendix E).
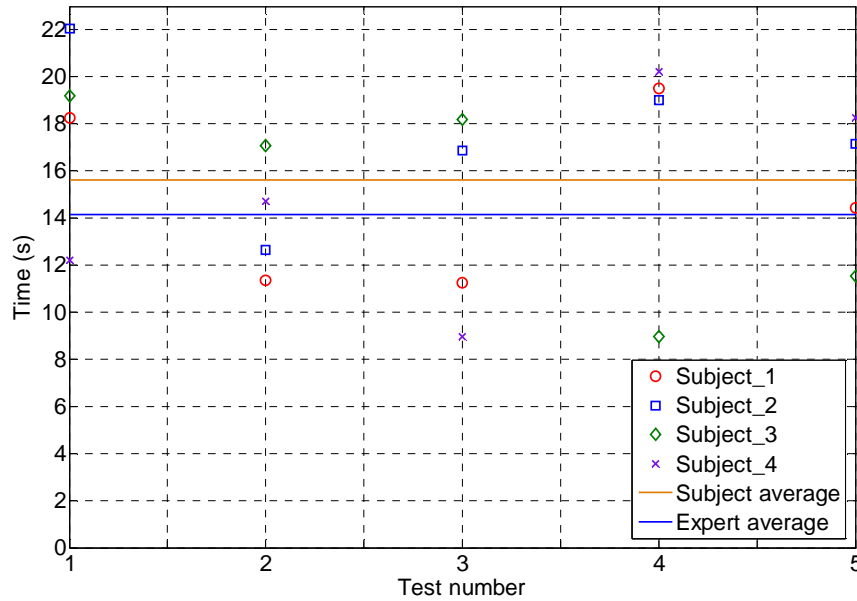
Figure 43: Time_2 data for the five intuitiveness tests conducted by four subjects

These subject averages compare very well with the respective expert averages of 7.93 s and 14.15 s from the movement control tests, which points to a very intuitive system. With little knowledge of the joystick, only a short introduction and minimal practice, the subjects executed movement control within times that are highly comparable with the first 20 tests, sometimes even outperforming the designer of the system. This indicates successful intuitiveness tests.

Subjects were asked to give feedback on how user-friendly the joystick is, with the main positive feedback points being that the joystick is easy to operate, the movement scale from joystick to robot is well defined and that the safety button functionality is effective. On the other hand, subjects mentioned that at some places within the joystick working volume, the movement was constrained by some joystick joints interfering with each other, although this factor did not affect the outcomes. It was also noted that the joystick setup was optimised for right-handed users and that orientation shifts and some changes to the working volume calculations would be necessary to reconfigure it for left-handed users.

Finally, the safety and technical specification tests were executed. The safety switch functionality proved that movement on the robot-end only occurred when the joystick's safety switch was pressed; when it was released, nothing happened. The emergency stop button could also be pressed any time during operation, which resulted in the whole master-slave system turning off and all movement being stopped. The boundary constraints were also tested: all SSM motors stopped moving when the specified boundary on each side was reached. During these tests, the technical aspects were also recorded and entered into the technical specification sheet that can be viewed in Appendix E.

CHAPTER 11 CONCLUSIONS AND RECOMMENDATIONS

Chapters 1 through 10 covered the work that was necessary to satisfy the main thesis objectives that were set out in Chapter 3. This final chapter looks at these objectives and discusses where and how well they were satisfied. The direct result of the work executed under each objective is given, as well as the problems that were encountered in each case. Conclusions are then made from these observations. After this, and based on the problems that were noted, recommendations for continuation of the surgical robot project are made.

11.1 Conclusions

Chapter 3 contained a list of eight main objectives that had to be satisfied in order for this thesis to be considered successful. These objectives are discussed below in terms of the manner and degree to which they were satisfied.

The first objective was to *"obtain a full understanding of the field of robotic surgery, specifically with regards to user interface systems, by means of researching current surgical robots and their specific designs".* This work was covered in Chapter 2 where the current state of MIRS, current examples of surgical robot user interfaces and necessary conventions and notation for robot mathematics were discussed. A detailed overview of the PSM and SSM systems were also given. Literature showed that designs incorporating current MIS tools together with haptic devices are used, but that the fulcrum effect is still present and the use of haptic feedback increases the complexity of the system. It also revealed that user interfaces based on the designs of their surgical robots are easy to control, but that the master-slave control system is limited to the physical constraints of the specific robot. Independent designs provide enough versatility of movement while not constrained in any way. This information provided a basis for understanding the rest of the thesis and was sufficient for carrying out the necessary work, thereby satisfying the first objective.

The next objective was to *"design and construct a mechanical user interface system ... that the surgeon can use to control the movement of the seven DOF surgical tool"*, for which Chapter 4 was responsible. The iterative concept development process led to a final chosen design that, upon assembly, adhered to all of the engineering specifications that were set for it: it provided unhindered and intuitive movement (controlled by one hand) in a 30 cm x 30 cm x 30 cm working volume and its design was not constrained specifically to the SSM-PSM system. It allowed, with downscaling, a 10 cm x 10 cm x 10 cm 'safe space' of operation at the robot end. Calculations confirmed that less than 1 mm deflection was experienced by the joystick's base frame, while the maximum allowable kinematic chain deflection of 0.6 mm was also not reached. The electromagnetic brakes allowed each of the joystick joints to be actuated independently – allowing direct control of the PSM motors – and the safety switch was incorporated successfully.

The testing done in Chapter 10 supported the conclusion of an intuitive system by showing that subjects needed minimal preparation time and minimal understanding of the system's functional operation, before being able to control the robot optimally. It did, however, also draw attention to the following subject feedback points: at some places (mainly outside the optimal working volume) the joystick joints interfered with each other, thus hindering movement; and the joystick was optimised only for right-handed users. The interfering of the joints is one negative effect of the existence of seven degrees of freedom, although it did not affect the joystick's ability to control the robot. With the regards to intuitiveness, the multiple joints are not considered redundant as they provide the operator with enough space and manoeuvrability to reach the required positions and orientations. Fewer joints would most likely have interfered with this ability and thus hindered control and intuitiveness. On the robot side, the fundamental problems with the PSM and SSM hindered effective speed control of the robot and prevented smooth movement of the end effector.

It was also noted that, since the degrees of freedom of the master and slave systems do not correspond to each other directly, a small motion on the master side may cause relatively large movements of the slave system joints, which would be non-intuitive and cause performance issues. Spong *et al.* (2006) outlined the existence and effects of singularities (different configurations of the manipulator joints for the same end effector position and orientation) and how to handle them. This process was however not incorporated into the control system design of the user interface as this would have required total functionality of the slave system and more time for software development. It was decided in the thesis that only enough control system development would be done to demonstrate control of the slave by the master. However, several methods were still implemented to prevent such occurrences and to keep the system safe. These include the safe operating space that was created and putting maximum limits on the speed of the joints. Also, the design of the slave system is such that mostly only one configuration for a required output is possible within the joints' physical constraints.

Objective three, to *"design ... the encoder system that will be used to track the movement of the surgeon's hands"*, was covered partly in Chapter 4 and partly in Chapter 5. An easy, low cost and accurate solution was found in the AS5040 rotary magnetic encoder chip and its accompanying magnet. The electronic circuitry was successfully implemented, PCBs were created and mounted on each joint of the joystick and the software allowed 1024 positions per revolution to be sampled, which is more than the 1005 positions needed to satisfy the surgical robot resolution requirement. Although not fulfilling the suture resolution requirement of 10050 positions per revolution, the AS5040 chip provided enough resolution to demonstrate the functionality of the joystick accurately.

The objective of designing and implementing *"the necessary electronics that will allow the mechanical, electronic and encoder components of the joystick to work together as one system"* was executed in Chapter 5. Circuitry was designed and PCBs were created for power supply, encoder functioning, brake functioning and connecting all of the inputs and outputs to the Arduino Mega 2560 control board. The designs and pin configurations were thoroughly documented. Upon

completion of this objective, the result was a functioning joystick with successfully implemented mechanical and electronic aspects. The safety switch and emergency stop button satisfied the safety specifications that had to be met, as the testing section indicated.

In order to control the robot with the functioning user interface, the master and slave systems had to be interfaced, which addressed the objective of implementing *"an effective communication system between the master console and the collective PSM and SSM system"*. Chapter 6 described this process, as well as the problems that were encountered with the PSM and SSM systems and the subsequent adjustments that were made. It was noted that fundamental flaws in the mechanical design of both the PSM (with regards to the spherical wrist design, the cables and the strong motors) and SSM (with regards to the system's inertia and lack of stiffness) would influence the joystick's ability to control the robot's movement. Still, this objective was satisfied with successful serial communication, in the form of a four-wire cable, being set up between the master and slave systems.

Objective six was then to *"design the control system that will regulate how the joystick is able to control the surgical tool"*, which proved to be a considerably more complicated task than anticipated. Chapters 7, 8 and 9 contain all of the work that was done for this objective. Firstly, both the joystick and the robot were mathematically modelled according to the DH convention. Forward kinematic calculations were then applied to the joystick to determine its end effector position and orientation with respect to the joystick's base reference frame. These quantities were recalculated with respect to the joystick working volume, which could be scaled down to the robot working volume. From there, the desired position and orientation of the robot end effector was calculated with respect to the robot's base reference frame, which finally enabled calculation of the robot's required joint variables through inverse kinematics. This process was successfully demonstrated with the MATLAB GUI that was created for simulation and testing purposes.

A novel aspect that emerged from this control section is the functioning of the safety switch and its joystick working volume. To keep the robot stationary while the safety switch is released and to move the robot from that precise stationary point when the safety switch is pressed, a new algorithm was developed. It keeps the joystick working volume locked to the joystick's end effector while the safety switch is pressed and it locks the joystick working volume with respect to the base frame while the switch is released. This implies that movement inside the joystick working volume (which directly translates to the required movement in the robot working volume) is only registered when the safety switch is pressed. No precedence of such an algorithm was found during the literature review, which pointed to its novelty.

Unfortunately, due to the mentioned problems with the SSM and PSM, the control system implementation could not be successfully demonstrated on the robot. The PSM's only reliable joint was Joint 4, the one using no cables; and only limited speed control was possible with the SSM. For this reason, two operational modes, one where inverse kinematics were just applied to the SSM

motors and one where the PSM could be controlled independently, were created and demonstrated. The safety aspect of staying within the allowable volume by constraining the SSM motor movement was also accomplished here. Chapter 9 documented the successful implementation of Arduino and MATLAB software in order to accomplish this objective.

The seventh objective was to *"design and conduct experiments to test the working of the control system and the accuracy with which the robot's movements are controlled"*. Chapter 10 described the testing procedure and results that were acquired for this objective. By using the joystick to move the robot end effector between two points, and doing this multiple times at relatively high speeds, it was shown that the joystick could successfully control the surgical robot's movement. Through similar tests, done by random subjects achieving highly comparable results, the system proved to be very intuitive. Furthermore, safety aspects were confirmed and a full technical specification of the master-slave system was acquired.

The final main objective was to *"write a full technical report on all of the above sections and the findings of the thesis"*, of which this thesis is the result. It was created together with all of supporting documents – the technical drawings, the program listing and the user manual. Several sample videos of the master-slave system were also taken to demonstrate the successful control of the surgical robot by the user interface.

It is also important to note that, while execution of the main objectives occurred, adherence to the budgeting and timing constraints also played a role in the final success of the project. Appendix F contains information on the total budgeted and actual expenses. The total expenses amounted to R10000, a value much lower than what was anticipated.

11.2   Recommendations

Based on the research, several recommendations can be made for future work.

Firstly, for the master system, the conclusions pertaining to the joystick's mechanical and electronic aspects indicated successful implementation with minor problems, although some improvements can be suggested. The test subject feedback mentioned the problem of interfering joints – this can be corrected by increasing link lengths, as well as making the joystick base extension higher, which would decrease the chance of joints interfering with each other. Another solution might be to add another joint, giving the system more versatility of movement. Then, to create an even lighter product that puts less weight on the operator's hands, a type of ABS plastic material can be used instead of steel or aluminium. Through more design iterations the size of the encoder PCB and holder can also be decreased, which will also decrease the mass.

On the electronic side, the rotary magnetic encoders provided enough resolution according to the surgical robot resolution requirement, but not enough for when

the flaws of the SSM and PSM are removed and the suture resolution requirement becomes important. To account for this, higher resolution encoders must be researched and incorporated into the design. Another possible drawback of the encoder system is the need to zero the joystick with each system start-up procedure. To counter this, absolute encoders can be investigated. The large amount of wires and cables connecting the encoders and brakes to the control board is also a factor worth investigating, as it could also have contributed to the joystick's movement being hindered. Wireless communication or better cable layout should be considered. Currently, encoders and brakes all plug in separately on the main joystick PCB. To simplify this aspect, and to prevent cables from being plugged into the wrong slots, a single connection port can be implemented.

In terms of controller use and software, some improvements are possible. The 16 MHz speed of the Arduino's microprocessor allowed control to be executed every 20 ms. To increase this value for better control, faster microprocessors (possibly even embedded PCs) could be investigated. Then, the software implementation of the safety switch and joystick working volume was limited to position control – when the button was pressed, the joystick's position could be sent to the robot; when it was released, no data was sent and no error occurred upon pressing the button again. But this was not implemented for orientation control, a factor which must be investigated further.

Several recommendations can be made for the slave system, mostly with respect to the PSM and SSM mechanical designs. It is recommended that both systems undergo more design iterations and that care should be given to these designs. The problems to address for the SSM would be the lack of stiffness and rigidity and the wrong motor specification. Encoder placement is also important, with the best mounting position for encoders being directly on the rotating joint that emerges from the gearbox and not at the back of the motor. For the PSM, the spherical wrist design should be reiterated and the effect of different joints on each other should be eliminated. Again, the over-specification of motors should be addressed and methods should be put in place to enable robot zeroing and detecting when the specific joint's boundaries have been reached. Only once these improvements have been made and when the combined SSM and PSM system is functional can the joystick's control be demonstrated efficiently on the robot.

Provided the above recommendations have been followed and the adjustments have been made, new tests can also be conducted to further demonstrate the abilities of the master system. This can include a fatigue test to see the effect of the joystick on surgeon fatigue after a prolonged operating period, which can determine the effectiveness and user-friendliness of the joystick. Also, once the slave system is in a working state, new tests can be conducted to determine the overall resolution and accuracy of the master-slave system. For these tests, standard reference position and error tracking protocol can be used.

APPENDIX A: MECHANICAL ASPECTS

This appendix contains all necessary data that is complimentary to the mechanical development of the joystick and the rest of the mechanical aspects presented in this thesis. It discusses the mechanical concept development process and it contains the assembly drawings for the final joystick design.

A.1   Concept development

A.1.1   Concept 1

Before the PSM and SSM were properly inspected and their shortcomings realised, the main idea was to design a joystick with more or less the same joints and geometry as the surgical robot, as researched showed this option to be a simple solution. Thought was only given to how the different joints should fit together to form a full kinematic chain and not to the more detailed specifications (provided in Chapter 3) that were unknown at this stage. The result is shown in Figure 44, with the numbered DOF joints and their movement (as they relate to the respective DOF of the robot) also indicated.
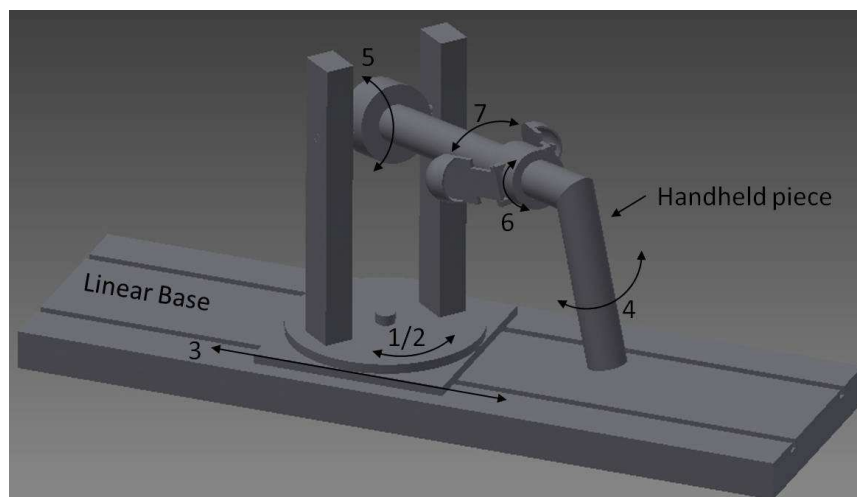
Figure 44: Concept 1

The idea is that the operator's hand closes around the part that is marked as DOF 4, with the thumb and index finger fitting into the two gripper parts at DOF 7. Then the gripper can be actuated as indicated by the arrows at DOF 7, while the thumb and index finger can also be used to rotate the same part around the horizontal shaft for DOF 6. DOF 5, the elbow joint, is operated by lifting the hand up or down; and DOF 4, the long rotating axis of the PSM, is actuated by turning the part in the operator's hand CW or CCW. The linear movement is brought about by pushing the handheld part forward or backward as the whole system

slides on the rectangular base (for DOF 3). The circular pad can rotate in order to provide a combined DOF 1 and 2 movement.

Concept 1 was used as a basis for further design, based on certain adjustments. One shortcoming was the fact that Concept 1 would not be intuitive and easy to handle. To hold the hand piece stationary while only moving Joint 6 with the thumb and index finger would be difficult and using the whole hand to actuate Joint 5, instead of just a small shift in orientation, would not be optimal. Furthermore, the combination of DOF 1 and 2 of the SSM into one joint puts constraints on the joystick's, and therefore the robot's, movements. These aspects led to the design of Concept 2.

A.1.2   Concept 2

The main adjustments from the previous concept came in terms of the handheld part of the joystick and the separation of Joints 1 and 2. Concept 2, with the indicated DOF, is shown in Figure 45 below.
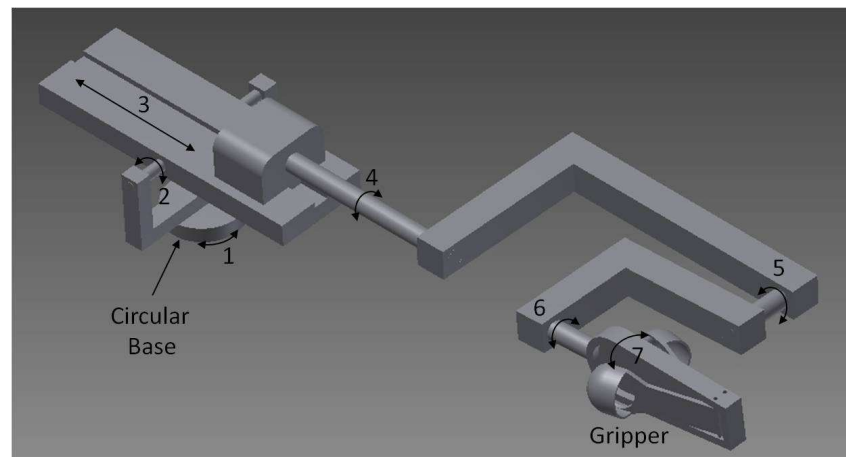
Figure 45: Concept 2

This joystick is held at Joint 7 by the thumb and index finger, where the gripper is also actuated. Joint 6 is actuated by turning the handheld piece CW or CCW, while the elbow joint is actuated by rotating the joint at DOF 5. If Joints 5, 6 and 7 are held together they can all be rotated about the shaft at Joint 4, in order to actuate the long shaft of the PSM. The linear movement is the same as for Concept 1. The two DOF at the base of the joystick are separated here, with the linear base being able to rotate about the Joint 2 shaft and the circular base at the bottom also able to rotate.

The joints of this concept corresponding to the PSM's joints are already much more intuitive than that of Concept 1. Holding the gripper between thumb and index finger, turning that part to get a Joint 6 rotation and easily shifting the orientation at the elbow joint will be much easier for the operator. Keeping Joints 5, 6 and 7 locked together in order to rotate Joint 4, however, proves more difficult. The extra joint at the base allows the joystick a better movement reach

84

than Concept 1 and also keeps the DOF in the same order as the combined PSM and SSM.

Concept 2 was an improvement on Concept 1. Consequently, a more detailed design was created, with the result shown in Figure 46. The DOF are not indicated, because they agree with those in Figure 45. The only main change from the conceptual design is the extra DOF 1 joint. This new joint agrees with the base joint of the SSM, making every joint of this joystick comparable to the same joint on the robot.
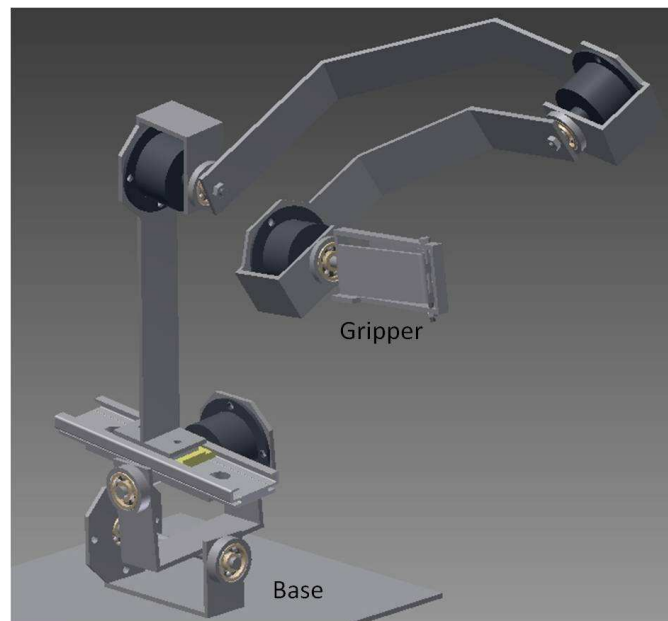
Figure 46: Concept 2 adjusted

The links were designed to be cut from thin sheet metal (either stainless steel or aluminium), making it lightweight and easy to acquire. In order to create revolute joints, the sheet metal can be bent into a U-shape, housings for bearings can be fixed to both sides of the U-shape, bearings can be fitted into the housings and a shaft can be inserted through the bearings. This creates a free rotating shaft (to which the next link in the kinematic chain can be fixed) with two bearing-supports to keep the system rigid. For the prismatic joint a linear sliding bearing was used in the design.

The four dark, cylindrical, flanged parts in Figure 46 are CAD models of an electronically actuated magnetic brake. The stator part is flanged and fixed to a stationary wall, while the rotor is fixed to the shaft. When a voltage is applied, a magnetic field is created and the rotor and stator snap together, thereby locking the shaft's movement and braking the joint. These parts were included because of two reasons: firstly to account for the problem where Joints 5, 6 and 7 have to be locked together in order for Joint 4 to be rotated; and then also to provide the safety feature of locking the joystick's links whenever necessary. Brakes are not indicated for Joints 1 and 3, but they could be implemented on any joint of the

joystick, although a different option would be necessary for the linear joint. Specification of added and designed parts will only be done for the final design.

Concept 2 provided a more detailed look at how the joystick would function if it were designed according to the joints and geometry of the surgical robot. The concept is much more intuitive than Concept 1, as it resembles the robot and would therefore be easy to handle. The 'same geometry' factor would also make the controlling aspects simpler. The last four joints (including the gripper) are based on the design of a spherical wrist fitted with a gripper, which makes this part also general to many industrial robots. The rotating shafts simplify the application of rotary encoders, while a linear encoder could be used on the prismatic joint. The gripper handle is designed from two plates pivoting at their ends around the same shaft, which resembles the pivoting point of the gripper teeth at the end of the PSM. The operator can press the plates together or open them up in order to actuate the gripper in a similar manner.

Still, these positive points could not account for the problems that arose from modelling the joystick on the geometry of the robot. At this stage a better understanding of the PSM and SSM showed that the robot had problems with stiffness, rigidity, operation, encoder placement and other aspects (please refer to Chapter 6 for a more elaborate explanation). It was evident that both the PSM and SSM would have to undergo several more design iterations before they would be perfectly operational, which implied that Concept 2 (or at least the SSM part of it) would then be obsolete. This motivated the design of a next concept with its own kinematic chain configuration.

A.1.3   Concept 3

For Concept 3 the parts of the previous concept that fitted design specifications were kept as is, which included the use of sheet metal, the design of the revolute joints using bearings and shafts and the use of brakes. However, an improved kinematic chain that provided enough movement versatility and was still intuitive, was necessary. Figure 47 shows the concept, with the previously designed gripper handle at the end. The base, gripper and all other DOF are indicated. As can be seen, all joints are revolute joints. Also evident is the fact that the design for Joints 4, 5, 6 and 7 is very similar to that of Concept 2. The 'Joint 4 - Joint 5' design may look very different, but they are still two revolute joints situated at 90° with respect to each other, as was the case for Concept 2.

This specific kinematic chain was not based on any precedent designs, but rather chosen because of its freedom of movement and intuitiveness. No understanding of the geometry of the robot is necessary to operate this joystick; the gripper can simply be held and moved unhindered to different positions and orientations.

This design simplified the use of encoders, as only rotary encoders were now necessary for all the revolute joints. The safety aspect could also be addressed by putting brakes on each joint.
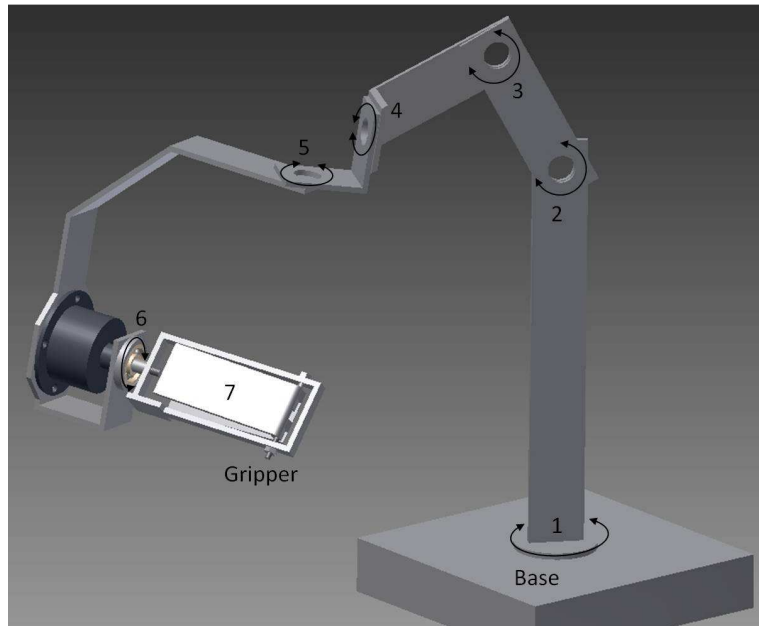
Figure 47: Concept 3

In order to further define this concept, a more detailed design stage was executed, with Figure 48 (gripper not shown) being the result. One can see the sheet metal, brakes, bearings and shafts being implemented again.
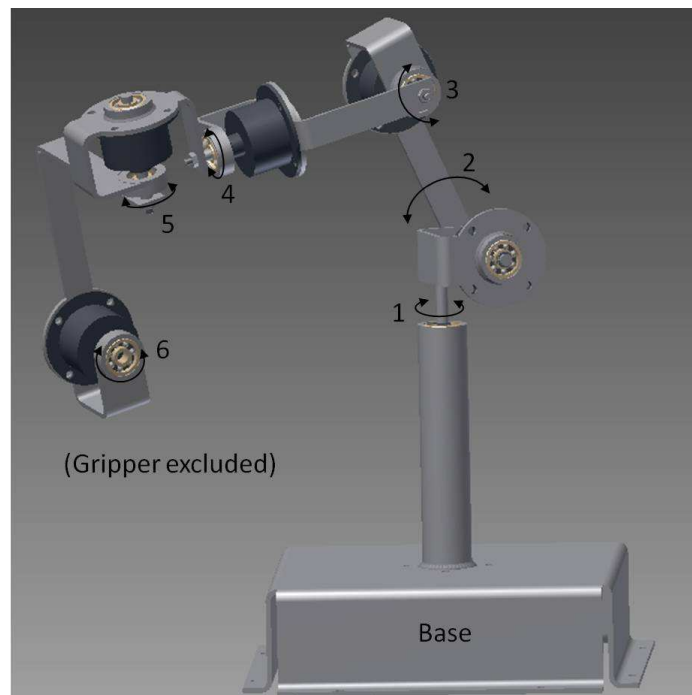
Figure 48: Concept 3 adjusted

2 mm thick sheet metal that could be laser cut and bent was chosen as the main material for the links of this concept. 5 mm shafts were chosen for the joints, with

their ends threaded so that following links in the chain could be fastened to the shaft with thread and nuts. It was decided that the bearing housings could also be laser cut from thicker material and it would then be fixed to the links at the appropriate positions.

A.1.4   Concept 4

One problem that was apparent from Concept 3 was the fact that the operator would have to carry the weight of most of the joints because, if the gripper was released, the design of joints 2 and 3 would cause the whole chain to hang from the attachment at joint 1. This meant putting more strain on the operator's hand which would constitute a safety threat during a surgical procedure. The first option was then to use a kind of torque limiter on Joints 2 and 3, a feature that would allow movement of these joints when forces are applied (i.e. the operator moves the gripper) but would keep the joint stationary once these forces are removed. Research showed that torque limiters, for example the SK5 backlash free model from RW-America (*RW-America*, [S.a.]), were specifically designed for these problems and could be used in this case.

A.1.5   Concept 5

Although a good solution, the use of torque limiters implied the need for a different design of Joints 2 and 3, more mass added to the system and greater expenses. The other option was to redesign the base frame and keep Joint 1 in the same orientation, but to let the whole kinematic chain hang from it instead of connecting them in the upward direction as Figure 48 shows. The base frame of the joystick would then counter the gravitational forces of the links, which meant that the surgeon would not have to. This solution proved to be more efficient than the torque limiter option, which led to the final design iteration. Every joint of the joystick was kept identical; only the design of the base frame was altered to enable all the joints to hang down.

Other requirements and engineering specifications motivated the need to consider more aspects of this concept, including the effect of the robot's physical constraints on the joystick and the safety considerations.

In order to keep movement within the physical constraints of the SSM and PSM, two main initial methods were considered: specific physical construction on the joystick and program implementation. Specific physical construction implies the use of physical stop points on the joystick resembling the physical constraints on the robot. The problem with this method is that the design of the joystick will then be constrained by the physical aspects of the robot – an option already discarded when Concept 3 was introduced. Control of the positions of the robot's 7 joints, with specific consideration of the physical constraints, will therefore have to be done by software implementation. This method would entail stopping the robot's actuators by means of the controlling program if the encoders sense that the links have nearly reached their respective constraints. The angles provided by Table 1 and Table 2 have to be incorporated here.

In terms of safety, first option for regulating this has already been mentioned: software implementation. The controlling software of the user interface can limit the movement of the robot, not only with regards to staying within the physical constraints of its links, but also by limiting the reach and velocity of the end effector in the patient. By creating a certain 'safe space' within which the surgical tool is allowed to operate, the software can keep the robot from moving past set limits, even if the joystick itself moves further. The velocity of any or all of the robot joints can also be limited through software control.

Another probable necessity in terms of safety would be the ability of the robot to hold its position at times when the surgeon does not want the joystick's movements to be carried out at the robot end. An obvious example of such a situation is when the operator needs to let go of the controls in order to rest his/her hand or arm or for any other probable reason. To counter this, a type of safety switch can be incorporated: only when a switch is activated will movement be transferred from the joystick to the robot; otherwise, nothing will happen.

A further extension of the above method may be the use of an emergency stop button, as is already incorporated in the PSM-SSM design (see Chapter 6), to either stop all movement of the robot by means of software, or to stop all movement of the joystick by means of brakes.

A.1.6 Concept evaluation

The five concepts were evaluated against the set of engineering specifications shown in Chapter 3 and an evaluation matrix was created to compare the evaluation results. It should be noted that many of the specification criteria aspects were derived throughout the concept development process, which implies that several concepts might not satisfy some of the specifications at all. Where applicable, these concepts were awarded a score of zero at the specific points. Concepts were then awarded a score from zero to five based on how well they satisfy each specification in comparison with the other concepts, with zero being the worst and five being the best. The results of the evaluation process are shown in Table 13.

As the table indicates, several concepts scored zero for criteria points 2, 3, 4 and 9. This was because encoders were not yet implemented with the early concepts, implying that resolution tests were not yet executed, and because the safety switch specification was only derived during the development of Concept 5. Table 13 shows Concept 5 scoring 46 marks, the highest of all the concepts, which led to it being chosen as the final concept for further development.

Table 13: Concept evaluation matrix

	Specification Criteria	Concept scores				
		1	**2**	**3**	**4**	**5**
1	Controllable with only one hand.	5	5	5	5	5
2	Provide unhindered movement space of 30 cm x 30 cm x 30 cm.	2	3	4	4	5
3	Encoders should be small and lightweight	0	0	0	0	0
4	Provide 9 mm resolution for surgical robot resolution requirement	0	0	0	0	5
5	Provide 0.9 mm resolution for suture resolution requirement	0	0	0	0	0
6	Allow a 'safe space' volume of 10 cm x 10 cm x 10 cm for the robot end effector.	2	2	3	4	4
7	The joystick's base frame should not experience more than 1 mm deflection	3	3	3	5	5
8	The combined link deflection of the kinematic chain should be less than 0.6 mm.	3	3	3	5	5
9	It must be possible to actuate the joystick joints separately from each other.	0	3	5	5	5
10	A safety switch should be incorporated.	0	0	0	0	3
11	Minimal mass should be carried by the operator.	2	2	3	4	5
12	The joystick's end effector should directly translate its movement to that of the robot end effector.	0	0	0	0	5
13	Detailed knowledge of how the joystick functions should not be a prerequisite to being able to operate it.	2	2	4	4	4
	Totals	**19**	**23**	**30**	**36**	**51**

A.2 Assembly

With the detail design of Concept 5, technical part and assembly drawings were created for the whole joystick. The main assembly drawing, indicating the bill of materials and the main joystick dimensions, is shown below in Figure 49 and Figure 50.

Figure 49: Assembly drawing 1

ITEM	BESKRYWING	AANTAL	MATERIAAL / SPESIFIKASIES
11	BASE SIDE PLATE	2	DRW_03
10	GRIPPER ASSEMBLY	1	ASS_09
9	JOINT 6 ASSEMBLY	1	ASS_08
8	JOINT 5 ASSEMBLY	1	ASS_07
7	JOINT 4 ASSEMBLY	1	ASS_06
6	SHAFT AND MAGNET 2	5	ASS_11
5	JOINT 2 ASSEMBLY	1	ASS_04
4	SHAFT AND MAGNET 1	1	ASS_10
3	JOINT 1&3 ASSEMBLY	2	ASS_02
2	BASE TWO	1	DRW_02
1	BASE ONE	1	DRW_01 a&b

TITEL: FINAL ASSEMBLY

SKAAL OP A 3 = 1:2
MATE IN mm
DATUM JAN 2012

VEL No. **VAN** **VELLE** **No.** ASS_01

UNIVERSITEIT VAN STELLENBOSCH

UNLESS OTHERWISE STATED
TOLERANCES ± 0,1
ANGLES 1°

STUDENTE No. 15016161 | TEKENAAR JS HEUNIS | NAGESIEN

91

Figure 50: Assembly drawing 2

APPENDIX B: ELECTRONIC ASPECTS

This appendix contains all necessary data that is complimentary to the electronic development of the joystick and the rest of the electronic aspects presented in this thesis. Circuit diagrams and PCB designs are shown and master and slave control board pin configurations are given.

B.1   Circuit diagrams and PCB designs

This section provides all necessary circuit diagrams, created with Eagle Layout Editor, that were design mainly in Chapter 5, but also in other sections of the thesis. Where applicable, the PCB designs are provided. In some cases, for example the main joystick PCB, changes were made to the printed PCB's with the addition of more components. Where applicable, these additions will be indicated on the circuit/PCB designs. If specific, physical changes were made to the PCB layout or connections, it will be noted where applicable. The designs shown in this section are therefore the final versions (not necessarily physically but for all functional purposes) of the circuits as they form part of the master or slave system.

B.1.1   Encoder circuit

Figure 51 below shows the AS5040 encoder circuit diagram, while Figure 52 shows the corresponding PCB design. Red indicates top layer tracks, while blue indicates bottom layer tracks.

Figure 51: The encoder circuit

Figure 52: The encoder PCB design

Figure 52 shows that all pins of the 10-pin cable connector are connected to the encoder circuit, each pertaining to a certain output mode or voltage supply. However, only pins 7, 8, 9 and 10 (5V, ground and the A and B quadrature lines) of the connector were connected with wires to the joystick's main PCB, as the other output modes of the encoder chip were not incorporated. Furthermore, after PCB printing and construction, wires were soldered to pins 1 and 2 of the AS5040 chip (MagINCn and MagDECn) and also connected to the joystick's main PCB. This was to enable magnet position failure detection mode.

B.1.2   Gripper circuit

The joystick's gripper contains one switch (the safety switch) and two buttons (the gripper buttons). The combined circuit diagram is shown in Figure 53 below.
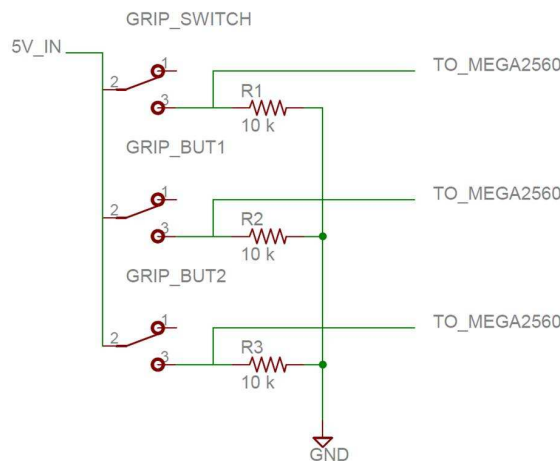
Figure 53: The gripper circuit

Outputs of this circuit are connected via wires to the joystick's main PCB, from where it is connected to digital input pins on the Mega 2560.

B.1.3 Brakes circuit

Figure 54 below shows the combined circuit diagram for the six electromagnetic brakes, the included TNR9G820K varistors and the controlling Reed relays. The varistors are connected in parallel with the brake ports. The relay coils are controlled by digital output pins on the Mega 2560: when a 5 V level is registered, the relay switch is turned on, which in turn provides a 24 V supply the appropriate brake. The full PCB design for this circuit is contained in the joystick's main PCB.
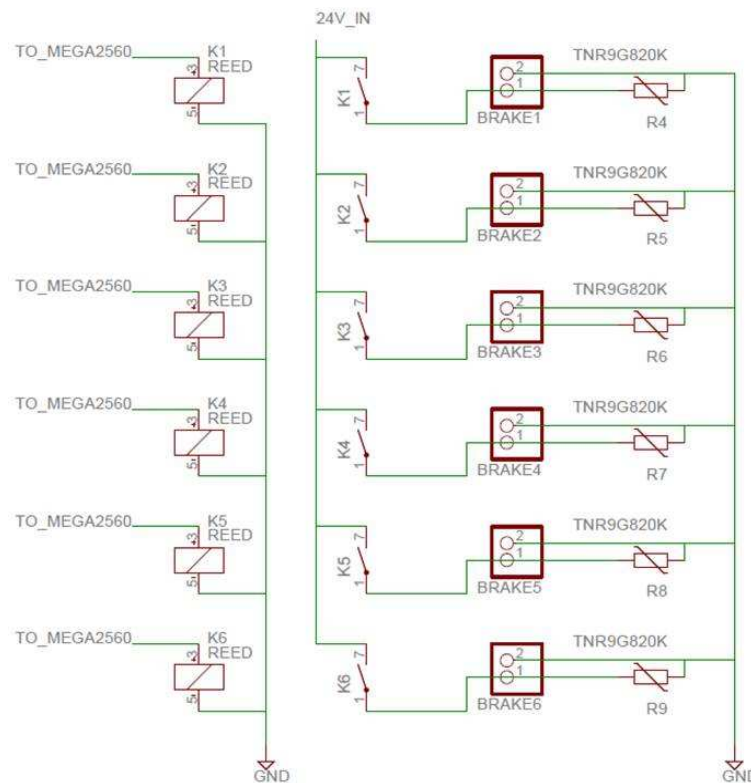
Figure 54: The brakes circuit

B.1.4 Brake switches circuit

Upon final assembly of the master system, the joystick plate contained nine toggle switches (with accompanying indicator LEDs) that form part of the user interface controls. These circuits are very similar to the one shown in Figure 53, just with LEDs added. The circuit for the six brake switches is shown below in Figure 55; the other three switch circuits were designed in precisely the same way. Custom, prototype PCBs were constructed for these circuits, as the designs were very simple.

95
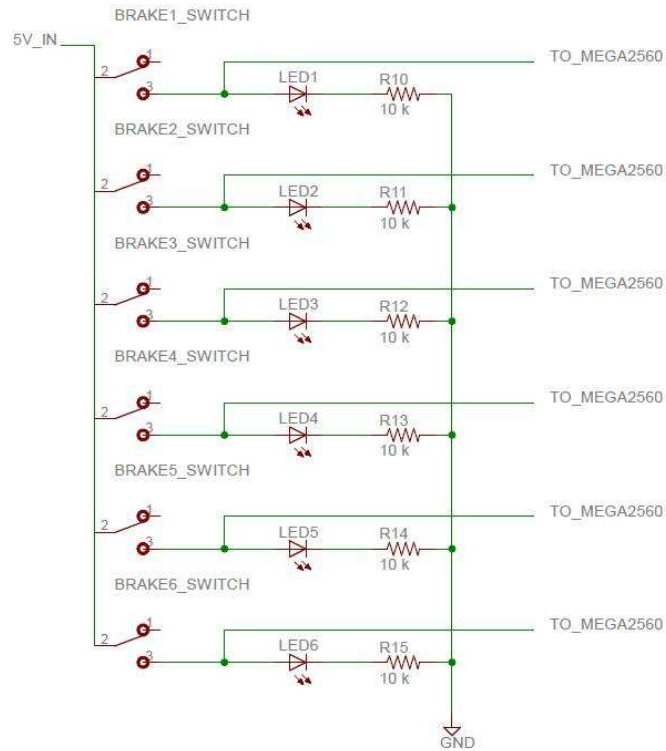
Figure 55: The brake switches circuit

B.1.5   Voltage regulator circuit

The voltage regulator circuit, where the L7812C and L7805C chips were used to respectively change the voltage level from 24 V to 12 V and from 12 V to 5 V, is shown below in Figure 56.
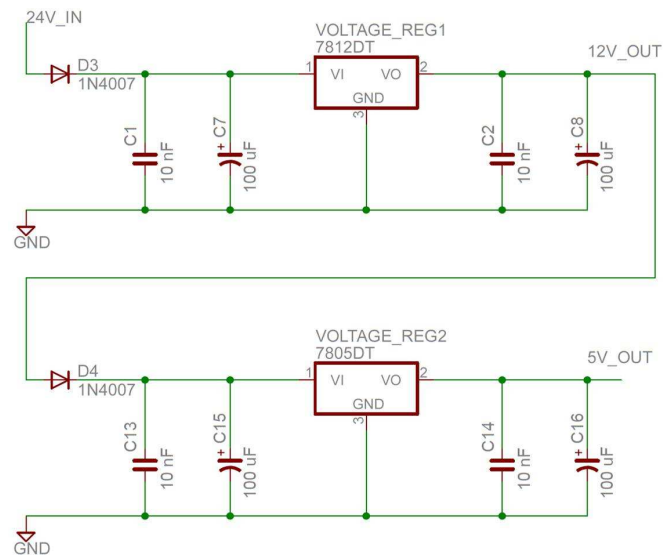
Figure 56: The voltage regulator circuit

B.1.6   Main joystick PCB

The circuits shown above for the gripper, the brakes and the voltage regulators, as well as a few prototyping pads and several connection ports for the joystick encoders, were all combined into one PCB design for the joystick. This main joystick PCB design is shown in Figure 57 below.
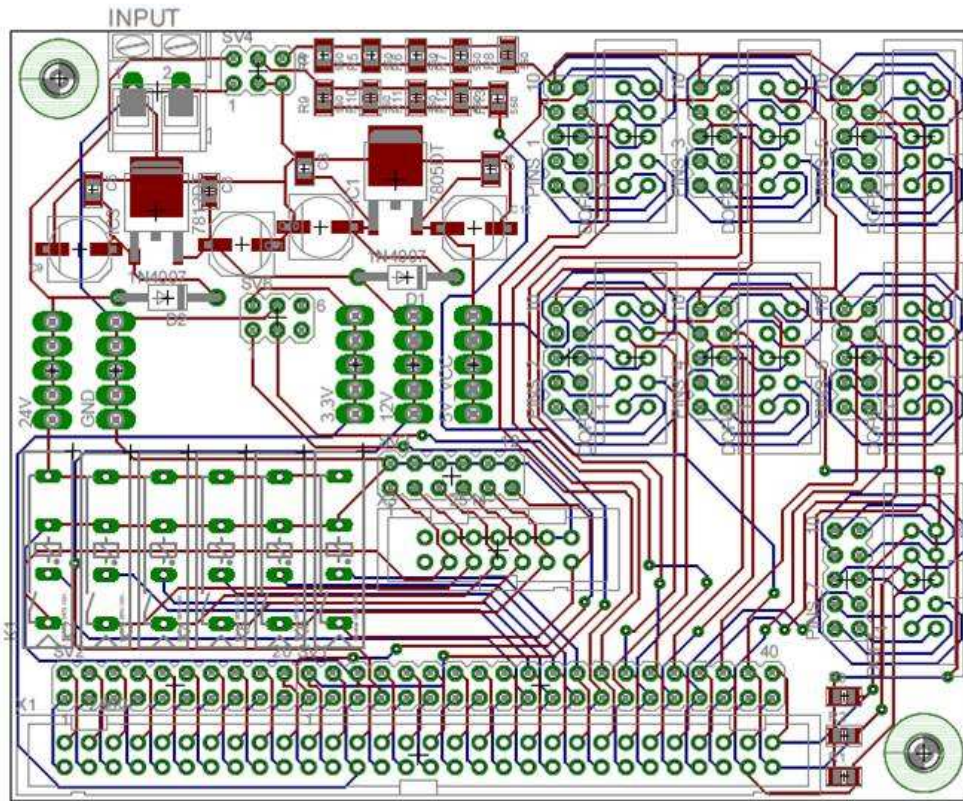
Figure 57: The main joystick PCB design

The 24 V input port is at the top left of the PCB, with the regulator circuitry situated beneath that. The green pads are for prototyping purposes and most are connected to 24 V, 12 V, 5 V or ground levels. Beneath that are the circuitry for the brakes, varistors and relays. On the right hand side of the PCB are several connection ports: six for the joystick encoders and one for the gripper circuitry. It was mentioned in Section B.1.1 that only 6-wire cables were used for each encoder to connect it to the above PCB. These wires (for 5 V, ground, line A, line B, MagINCn and MagDECn) were connected to the corresponding pins on the connection port plugs. All important inputs and outputs were then connected to the 60-way connection port at the bottom of the PCB, which was in turn connected to the Arduino Mega 2560 via the shield PCB mentioned in Section 5.8 and shown in Figure 22. The PCB design for this shield circuit is shown in Figure 58 below.
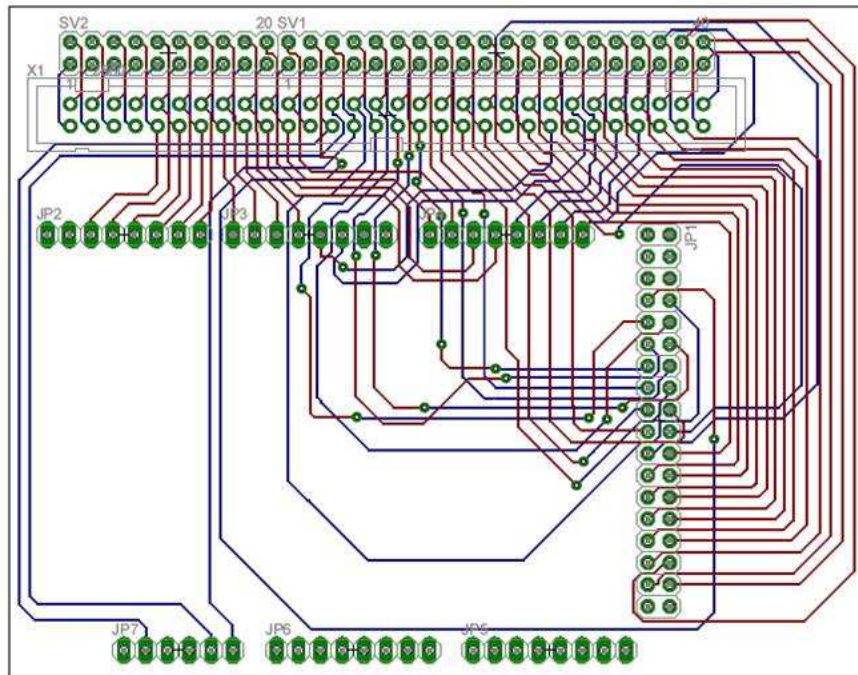
Figure 58: The joystick shield PCB

B.1.7 Main robot PCB

As Section 6.3.2 mentioned, a whole new PCB was designed to connect the SSM and PSM circuitry to the robot's controlling Mega 2560. The design is shown below.
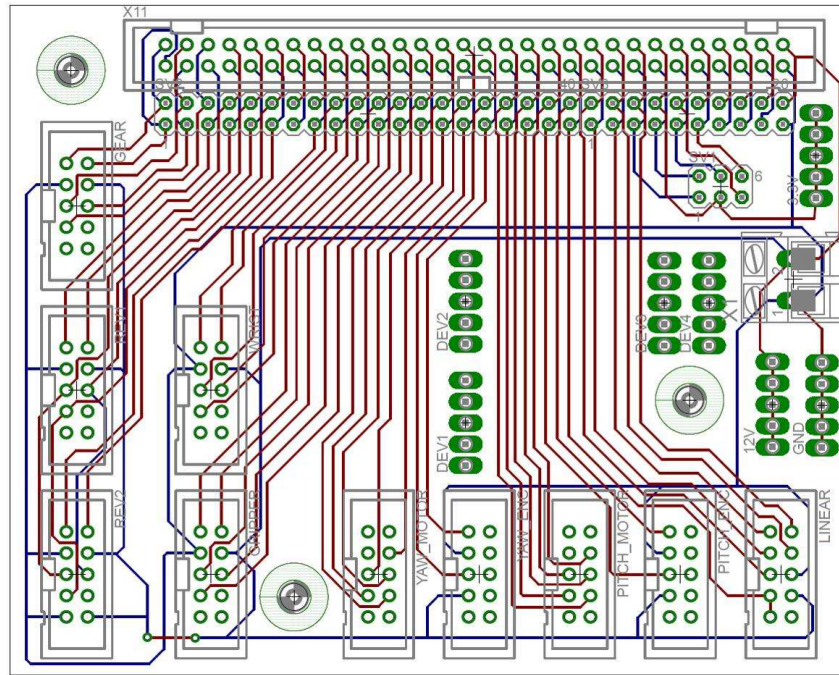
Figure 59: The robot PCB

As with the main joystick PCB, the one in Figure 59 contains several ports: one for each of the PSM motors' wires (for control and encoding), two for the Joint 1 motor control and encoder wiring, two for the Joint 2 motor control and encoder wiring and one for the linear motor's control and encoder wires. All of the necessary inputs and outputs are then connected to the 60-way cable connection port, which is in turn connected to the robot's Mega 2560 via the robot shield PCB. This PCB is shown below in Figure 60.
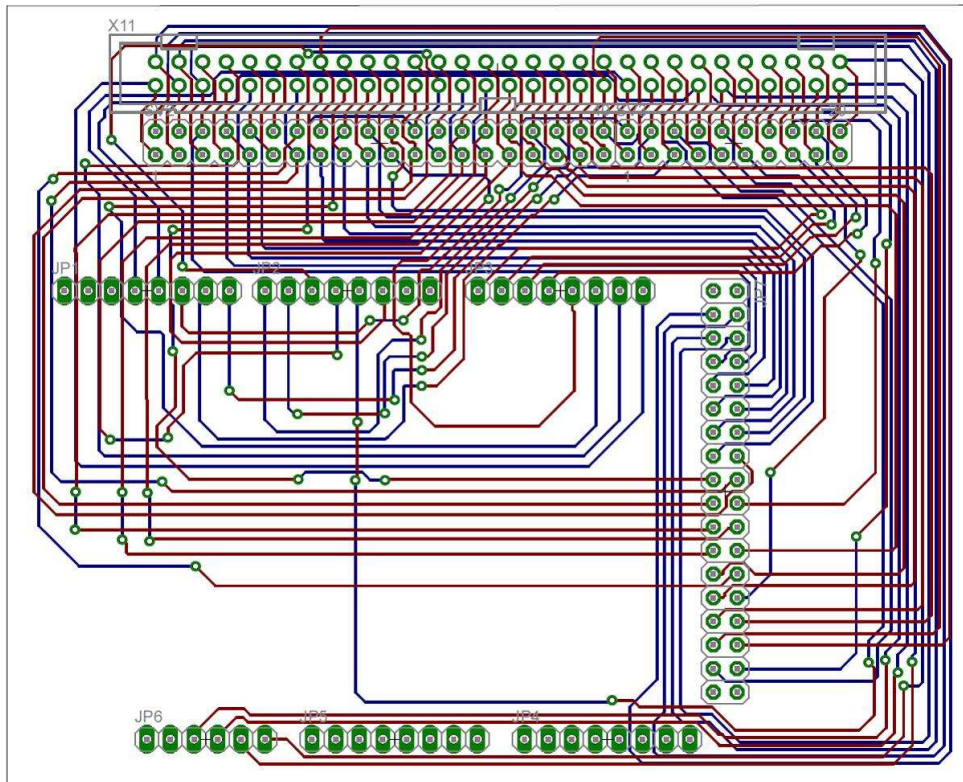
Figure 60: The robot shield PCB

B.2 Pin configurations

Section B.1 showed the circuitry for the many different connections between system inputs and outputs and the control boards. To enable direct correlation between specific pins on the Mega 2560 and its inputs or outputs, pin configuration tables were created for the main joystick and robot PCBs. On Table 14 and Table 15, the first column indicates the applicable pin number of the 60-way connection port on the joystick or robot PCB, the second and third columns refer to the Mega pin it is connected to and the last columns shows which joystick or robot pin is connected to it. The variable names are the same as those used in the Arduino programs.

Table 14: The joystick pin configuration

Connection Port Pin	Mega Pin Name	Pin Description	Joystick Pin Name
1	-	-	-
2	-	-	-
3	-	-	-
4	-	-	-
5	-	-	-
6	-	-	-
7	-	-	-
8	-	-	-
9	Digital 13	Digital input	NC (test)

100

10	Digital 12	Digital input	toggle8 (mode1)
11	Digital 11	Digital input	toggle9 (mode2)
12	Digital 10	Digital input	toggle7 (zero robot)
13	Digital 9	Digital input	toggle5
14	Digital 8	Digital input	toggle6
15	Digital 7	Digital input	toggle3
16	Digital 6	Digital input	toggle4
17	Digital 5	Digital input	toggle1
18	Digital 4	Digital input	toggle2
19	RX2 (Digital 17)	Serial 2 receive	RX2
20	TX2(Digital 16)	Serial 2 transmit	TX2
21	TX3 (Digital 14)	Serial 3 transmit	TX3
22	RX3 (Digital 15)	Serial 3 receive	RX3
23	-	-	-
24	Digital 26	Digital input	mag1_plus
25	3.3V	3.3V	3.3V
26	Ground	Ground supply	Ground
27	Vin	12V supply	Vin
28	Digital 27	Digital input	mag1_min
29	Digital 28	Digital input	mag2_plus
30	Digital 29	Digital input	mag2_min
31	Digital 30	Digital input	mag3_plus
32	Digital 31	Digital input	mag3_min
33	Digital 32	Digital input	mag4_plus
34	Digital 33	Digital input	mag4_min
35	Digital 34	Digital input	mag5_plus
36	Digital 35	Digital input	mag5_min
37	Digital 36	Digital input	mag6_plus
38	Digital 37	Digital input	mag6_min
39	Digital 38	Digital output	brake6
40	Digital 39	Digital output	brake2
41	Digital 40	Digital output	brake1
42	Digital 41	Digital output	brake4
43	Digital 42	Digital output	brake5
44	Digital 43	Digital output	brake3
45	INT1 (Digital 3)	External interrupt	Enc1_b
46	Digital 44	Digital input	Enc1_a
47	Digital 45	Digital input	Enc2_b
48	INT0 (Digital 2)	External interrupt	Enc2_a
49	INT5 (Digital 18)	External interrupt	Enc3_b
50	Digital 46	Digital input	Enc3_a
51	INT4 (Digital 19)	External interrupt	Enc4_b
52	Digital 47	Digital input	Enc4_a
53	INT3 (Digital 20)	External interrupt	Enc5_b
54	Digital 48	Digital input	Enc5_a
55	Digital 49	Digital input	Enc6_b
56	INT2 (Digital 21)	External interrupt	Enc6_a
57	Digital 50	Digital input	switch1
58	Digital 51	Digital output	warning
59	Digital 52	Digital input	button2
60	Digital 53	Digital input	button1

Table 15: The robot pin configuration

Connection Port Pin	Mega Pin Name	Pin Description	Robot Pin Name
1	INT2 (Digital 21)	External interrupt	Enc4_a
2	Digital 22	Digital output	Enc4_b
3	Digital 23	Digital output	m4d1
4	Digital 24	Digital output	m4e
5	Digital 25	Digital input	Enc5_b
6	Digital 26	Digital output	m4d2
7	Digital 27	Digital output	m5e
8	INT3 (Digital 20)	External interrupt	Enc5_a
9	Digital 28	Digital input	m5d2
10	Digital 29	Digital output	m5d1
11	-	-	Enc5_a (2nd motor)
12	-	-	Enc5_b (2nd motor)
13	Digital 37	-	zero_1
14	Digital 45	-	zero_2
15	Digital 30	Digital input	Enc6_b
16	Digital 51	-	zero_3
17	Digital 31	Digital output	m6e
18	INT4 (Digital 19)	External interrupt	Enc6_a
19	Digital 32	Digital output	m6d2
20	Digital 33	Digital output	m6d1
21	INT5 (Digital 18)	External interrupt	Enc7_a
22	Digital 34	Digital input	Enc7_b
23	Digital 35	Digital output	m7d1
24	Digital 36	Digital output	m7e
25	Digital 5	Digital output	m1pwm2
26	Digital 38	Digital output	m7d2
27	Digital 4	Digital output	m1pwm1
28	Digital 40	Digital output	m1d2
29	Digital 41	Digital output	m1e
30	Digital 42	Digital output	m1d1
31	INT0 (Digital 2)	External interrupt	Enc1_a
32	Digital 43	Digital input	Enc1_b
33	Digital 44	Digital output	m2d2
34	Digital 7	Digital output	m2pwm2
35	Digital 46	Digital output	m2d1
36	Digital 6	Digital output	m2pwm1
37	Digital 48	Digital input	Enc2_b
38	Digital 49	Digital output	m2e
39	Digital 50	Digital output	m3e
40	INT1 (Digital 3)	External interrupt	Enc2_a
41	Digital 12	Digital output	m3d1
42	Digital 10	Digital output	m3d2
43	-	-	NC (test)
44	Digital 8	Digital input	Enc3_a
45	Digital 9	Digital output	m1b
46	Digital 52	Digital input	Enc3_b
47	Digital 11	Digital output	m2b
48	5V	5V	5V
49	Ground	Ground	Ground
50	TX3 (Digital 14)	Serial 3 transmit	TX3

51	RX3 (Digital 15)	Serial 3 receive	RX3
52	-	-	-
53	-	-	-
54	Digital 53	-	NC (test)
55	Digital 47	-	NC (test)
56	Digital 39	-	NC (test)
57	TX2 (Digital 16)	Serial 2 transmit	TX2
58	RX2 (Digital 17)	Serial 2 receive	RX2
59	Ground	Ground supply	Ground
60	Vin	12V supply	Vin

APPENDIX C: CALCULATIONS

This appendix contains all of the calculations that were done in order to support certain design decisions that were made for this thesis. There are two main sections: mechanical design calculations and electronic design calculations.

C.1 Mechanical design calculations

C.1.1 Material choice

The calculations that are applicable here all relate to the strain experienced by different positions on the joystick due to the weight of the joints and links. The process evolved together with the concept development and was started when the desired material was necessary at Concept 3. A decision had to be made between stainless steel and aluminium; aluminium was preferable due to it being lightweight, but it first had to be shown that it would be strong enough to experience minimal strain, otherwise stainless steel would be the chosen material. A design requirement of less than 0.6 mm strain at the joystick end was deemed adequate.

Investigation was started into the necessary quantities that would help determine the strain experienced by different joints of the design (at that point Concept 3). For the calculations, the density (ρ) and modulus of elasticity (E) of both stainless steel and aluminium were necessary, as well as the sizes of the different links and joints. Certain simplifications also had to be made with regards to the mechanical design model in order to enable the use of the necessary equations. Table 16 below contains the material properties as given by Hibbeler (2005).

Table 16: Material properties

Material	Density (kg/m^3)	Modulus of elasticity (GPa)
Stainless steel	7800	200
Aluminium	2712	69

Then, when the moments or forces acting on the ends of the different joystick links are known, the following equations can be used to calculate the standard deflection of a cantilever beam, according to which the links will be approximated (Hibbeler, 2005):

$$v = \frac{M L^2}{2 E I} \qquad [13]$$

or

$$v = \frac{P L^3}{3 E I} \qquad [14]$$

where v is deflection (mm), M is the moment (Nm), P is the force (N), L is the moment- or force-arm (m), E is the modulus of elasticity (GPa) and I is the moment of inertia (mm^4).

To enable the use of Equations 13 and 14, some force and moment approximations were necessary. Firstly, the Concept 3 design was orientated as shown in Figure 61 (a), after which the approximations of Figure 61 (b) were applicable. The weights of every part to the left of Joint 2 cause a moment, M_1, about the top of the first link, which is modelled as a cylindrical pole. Then, the weights of everything to the left of Joint 3 act as force P_3 on the end of the Joint 3 link, which is modelled as a cantilever beam with width 2 mm and height 20 mm. the same approximation holds for the Joint 4 link, where the weights of everything to the left of Joint 4 act as force P_4. Finally, the same cantilever approximation is used at the Joint 6 link, with force P_6 (the weights of Joint 6 and the gripper) acting down on the corner of the L-shaped link. Here, however, the cantilever beam's height is 2 mm and the width is 20 mm. These approximations were seen as worst case scenarios and would therefore deliver the greatest deflections.

(a)
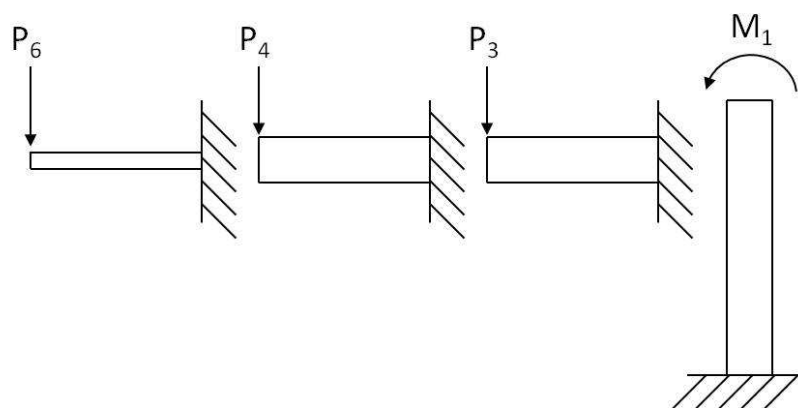
(b)

Figure 61: Concept 3 deflection approximations

Equations 13 and 14 could then be applied to the four cases in Figure 61 (b), but first the respective moments, forces, weights and moments of inertia had to be calculated. The sizes and lengths of the parts were taken from the CAD drawings of Concept 3 and the mass of a brake was taken from its datasheet to be 60 g (*Miki-Pulley*, [S.a.]). The mass for a part was calculated by multiplying the part volume by the applicable material's density, the moment of inertia was calculated for beams with circular or rectangular cross sections with equations from Hibbeler (2005) and the forces could be determined from the masses. With this information, the resulting deflections at the four positions could be determined. As a safety precaution, all of the mass calculations were multiplied by a scaling factor of 1.5 in order to account for any extra possible deflections.

The values were then calculated for both materials and the results are shown in Table 17 below.

Table 17: Deflection values for Concept 3

Case	Material	Force/Moment	Deflection (mm)
Case 1 (M_1)	Stainless steel	0.796 Nm	0.0782
	Aluminium	0.480 Nm	0.1367
Case 2 (P_3)	Stainless steel	6.466 N	0.0096
	Aluminium	3.783 N	0.0163
Case 3 (P_4)	Stainless steel	5.102 N	0.0038
	Aluminium	2.925 N	0.0063
Case 4 (P_6)	Stainless steel	2.502 N	0.2516
	Aluminium	1.254 N	0.3654

As expected, one can see in all of the cases that the force/moment due to the stainless steel is considerably higher (due to its higher mass) and its deflection is lower than that of the aluminium. When adding the deflections of the separate points together, the stainless steel amounts to 0.3432 mm while the aluminium is 0.5247 mm. As both were less than the design requirement of 0.6 mm, and aluminium was the lighter material, aluminium was chosen as the final material for joystick construction.

C.1.2 Base deflection

To determine whether the new base frame of the final design would carry the weight of all the links and joints without showing apparent deflection, similar calculations as above were necessary. Firstly, some approximations were made in terms of the forces and moments applied on the base frame. Figure 62 shows the final assembly and the force, *P*, and moment, *M*, that are exerted on the base frame due to the weight of the links and brakes. *P* is taken to be the resultant force due to the weight of six brakes and all the aluminium of the joints, as well as a safety factor inclusion for extra weight. *M* is taken as the moment

experienced at the base frame corner as a result of force *P*, with the moment arm being 120 mm.
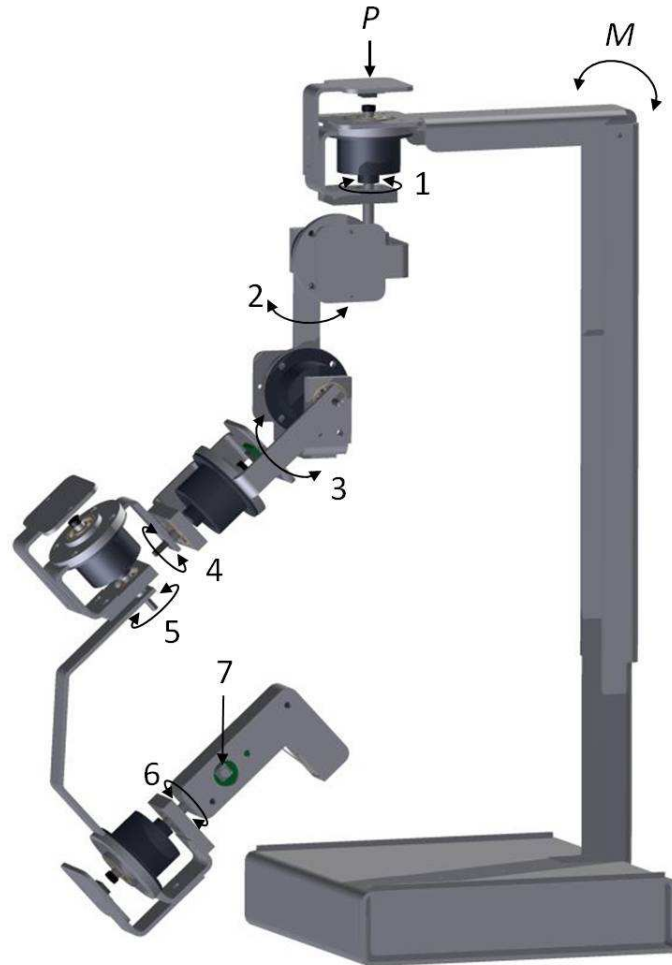
Figure 62: Base frame deflection approximations

Again, the combined weight of all the parts was calculated, as well as the resultant force and moment. The moment of inertia was calculated based on a U-shaped frame with 20 mm width and a side height of 10 mm. The lengths of the two base frame parts were taken as 120 mm and 400 mm from technical drawings. Then, Equations 13 and 14 could be used again to calculate the deflections. Table 18 below contains the calculated data.

Table 18: Deflection values for the base frame

Case	Force/Moment	Deflection (mm)
Case 1 (*M*)	0.647 Nm	0.073
Case 2 (*P*)	5.389 N	0.814

Thus, a total deflection of 0.887 mm occurs as a result of the joystick's weight. At this large scale, a maximum deflection of 1 mm was considered acceptable,

which shows that the resulting deflection of the base frame was within limits and the design was able to carry the joystick's weight.

C.1.3   Brake torque calculation

To see if the specified brake torque of 0.4 Nm (*Miki-Pulley*, [S.a.]) was enough to hold the joystick stationary, the necessary holding torque had to be calculated. It was noted that the brake which was likely to experience the largest amount of torque would be the one at Joint 2 (see Figure 62), as it would have to carry the weight of Joints 3 through 7. An approximation was made that the weights of the joints result in a force acting downward on Joint 3, with its link situated horizontally. The moment arm, the length of Link 3 and the resultant force then produce a moment about Joint 2, which is designated as the necessary holding torque of the brake. The resulting force was calculated from the weights (again with the safety factor included) as 3.783 N, the moment arm was taken from technical drawings as 100 mm and the holding torque was calculated as follows:

$$T_{holding} \; = \; P\,L \; = \; (3.783)(0.1) \; = 0.378 \; \text{Nm}. \qquad [15]$$

This showed that the specified brake torque of 0.4 Nm would be enough for the joystick's purposes.

C.2   Electronic design calculations

The only calculation applicable here is the one that was done for the L7812C voltage regulator to support the addition of a heat sink. According to its datasheet, the values for the L7812C's absolute maximum operating junction temperature (T_{OP}) and its junction-ambient thermal resistance (R_{thJA}) are respectively 125 °C and 50 °C/W (*L78xx Voltage Regulator*, [S.a.]). When the joystick is fully operational, tests confirmed that the current through the L7812C component reaches a maximum of 200 mA, while the regulator brings the voltage level down from 24 V to 12 V. The power dissipated in the regulator can then be calculated as:

$$P_{dissipated} \; = \; \Delta V\,I \; = \; (24 - 12)(0.2) = 2.4 \; \text{W}. \qquad [16]$$

With a 2.4 W power dissipation, the temperature rise is

$$T_{rise} \; = \; P_{dissipated}\,R_{thJA} = \; (2.4)(50) = 120 \; °\text{C}. \qquad [17]$$

With an assumed ambient temperature of 25 °C, this brings the total operating temperature to 145 °C, which is more than the absol ute maximum rating. This then motivated the need for the use of a heat sink and a cooling fan, which made the voltage regulator functional.

APPENDIX D: CONTROL EQUATIONS

This appendix contains the main equations that describe the forward and inverse kinematic processes as they relate to the models of the joystick and the robot that were given in Chapter 7. In addition to this, information on the conversion between raw encoder data and the values according to the models are given.

D.1   Forward kinematics

The A_i and T^i_j matrices specific to each joystick joint are given here, with the only variables being the variable joint angles that are received from the encoder data. The matrices were calculated according to Equations 1 and 2 in Chapter 7.

$$A_1 = \begin{bmatrix} \cos\theta_1 & 0 & -\sin\theta_1 & -20\cos\theta_1 \\ \sin\theta_1 & 0 & \cos\theta_1 & -20\sin\theta_1 \\ 0 & -1 & 0 & -60.8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [18]$$

$$A_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & -65\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & -65\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [19]$$

$$A_3 = \begin{bmatrix} \cos\theta_3 & 0 & \sin\theta_3 & 0 \\ \sin\theta_3 & 0 & -\cos\theta_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [20]$$

$$A_4 = \begin{bmatrix} \cos\theta_4 & 0 & -\sin\theta_4 & 0 \\ \sin\theta_4 & 0 & \cos\theta_4 & 0 \\ 0 & -1 & 0 & -118.5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [21]$$

$$A_5 = \begin{bmatrix} \cos\theta_5 & 0 & -\sin\theta_5 & 0 \\ \sin\theta_5 & 0 & \cos\theta_5 & 0 \\ 0 & -1 & 0 & -95.3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [22]$$

$$A_6 = \begin{bmatrix} \cos\theta_6 & -\sin\theta_6 & 0 & 0 \\ \sin\theta_6 & \cos\theta_6 & 0 & 0 \\ 0 & 0 & 1 & -27.6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [23]$$

$$T_1^0 = A_1 \qquad\qquad [24]$$

$$T_2^0 = T_1^0 A_2 \qquad\qquad [25]$$

$$T_3^0 = T_2^0 A_3 \qquad\qquad [26]$$

$$T_4^0 = T_3^0 A_4 \qquad\qquad [27]$$

$$T_5^0 = T_4^0 A_5 \qquad\qquad [28]$$

$$T_6^0 = T_5^0 A_6 \qquad\qquad [29]$$

The two important quantities, $M\text{-}R^0{}_6$ and $M\text{-}o^0{}_6$, can then be derived from Equation 29.

D.2 Working volume calculations

Before the inverse kinematics can be done, the position and orientation of the robot end effector are necessary in the robot's base reference frame. For this, the working volume calculations must be done first. This entails using the rotation matrix, $M\text{-}R_6$, and the position vector, $M\text{-}o_6$, to determine their counterparts in the joystick working volume, scaling this down to the robot working volume and finding the important values with respect to the robot's base reference frame.

D.2.1 Position

With the coordinates of the joystick working volume's origin defines as follows,

- X_{j0} is the position of the joystick working volume's origin on the X-axis of the $M\text{-}o_0x_0y_0z_0$ reference frame,
- Y_{j0} is the position of the joystick working volume's origin on the Y-axis of the $M\text{-}o_0x_0y_0z_0$ reference frame,
- Z_{j0} is the position of the joystick working volume's origin on the Z-axis of the $M\text{-}o_0x_0y_0z_0$ reference frame,

the joystick end effector position in the joystick working volume can be calculated with the following equations (refer to Figure 32):

$$X_J = o_6^0(X) - X_{j0} \qquad\qquad [30]$$

$$Y_J = o_6^0(Z) - Z_{j0} \qquad\qquad [31]$$

$$Z_J = Y_{j0} - o_6^0(Y) \qquad\qquad [32]$$

With these values known in the $M\text{-}o_Jx_Jy_Jz_J$ reference frame, the scaling could be done to produce X_R, Y_R and Z_R in the $S\text{-}o_Rx_Ry_Rz_R$ reference frame. Then the position had to be determined in the robot's base reference frame.

With X_{r0}, Y_{r0} and Z_{r0} described in the same way as X_{j0}, Y_{j0} and Z_{j0} above – but this time with regards to the robot working volume and the S-$o_0x_0y_0z_0$ reference frame – the position of the robot end effector in the S-$o_0x_0y_0z_0$ reference frame can be calculated as (refer to Figure 34):

$$X = X_{r0} - X_R \tag{33}$$

$$Y = Y_{r0} - Y_R \tag{34}$$

$$Z = Z_R + Z_{r0} \tag{35}$$

D.2.2 Orientation

According to Equations 7 through 12 (Sections 8.3 and 8.4), particular rotation matrices where necessary to transform M-R_6 in the joystick's base reference frame to R_J in the M-$o_Jx_Jy_Jz_J$ reference frame. There is no change from to R_J to R_R (in the S-$o_Rx_Ry_Rz_R$ reference frame) as these working volumes are orientated identically. Then, a set of rotation matrices are necessary to transform R_R to the required S-R_6 in the S-$o_0x_0y_0z_0$ reference frame. The equations for this are shown below:

$$R_J = \left(R_J^0\right)^T R_6\, R_J^0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} R_6 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \tag{36}$$

$$R_R = R_J \tag{37}$$

$$R_6 = \left(R_0^R\right)^T R_J\, R_0^R = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} R_J \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{38}$$

Here, the values of the matrices $R^0{}_J$ and $R^R{}_0$ were given by Equations 10 and 12 respectively.

D.3 Inverse kinematics

The inverse kinematics calculations were based on Equations 3 through 6 from Section 8.1. From the required position and orientation, S-R_6 and S-o_6, of the robot end effector calculated above, the wrist centre position is calculated as follows:

$$X_C = X - d_6\, r_{13} \tag{39}$$

$$Y_C = Y - d_6\, r_{23} \tag{40}$$

$$Z_C = Z - d_6\, r_{33} \tag{41}$$

Here, the value of d_6 was given in Table 9 and the r_{13} to r_{33} values refer to the specific indices of the required $S\text{-}R_6$ matrix calculated with Equation 38. Then, using the methods prescribed by Spong *et al.* (2006), the first three joint variables are calculated as follows:

$$\theta_1 = \begin{cases} -\left|\tan^{-1}\left(\frac{X_C}{Y_C}\right)\right|, & if\ X_C \leq 0 \\[2mm] \left|\tan^{-1}\left(\frac{X_C}{Y_C}\right)\right|, & if\ X_C > 0 \end{cases} \qquad [42]$$

$$r = \sqrt{(X_C)^2 + (Y_C)^2} \qquad [43]$$

$$s = Z_C - d_1 \qquad [44]$$

$$\theta_2 = \begin{cases} \left|\tan^{-1}\left(\frac{s}{r}\right)\right|, & if\ s < 0 \\[2mm] -\left|\tan^{-1}\left(\frac{s}{r}\right)\right|, & if\ s \geq 0 \end{cases} \qquad [45]$$

$$d_R = \sqrt{(r)^2 + (s)^2} \qquad [46]$$

$$d_3 = d_4 - d_R \qquad [47]$$

These variables are then used to calculate the first three transformation matrices of the robot:

$$S\_A_1 = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & 0 \\ \sin\theta_1 & 0 & -\cos\theta_1 & 0 \\ 0 & 1 & 0 & 682 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [48]$$

$$S\_A_2 = \begin{bmatrix} \cos\theta_2 & 0 & \sin\theta_2 & 0 \\ \sin\theta_2 & 0 & -\cos\theta_2 & 0 \\ 0 & 1 & 0 & -176 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [49]$$

$$S\_A_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 162 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [50]$$

$$S\_T_1^0 = S\_A_1 \qquad [51]$$

$$S\_T_2^0 = S\_T_1^0\, S\_A_2 \qquad [52]$$

$$S\_T_3^0 = S\_T_2^0\, S\_A_3 \qquad [53]$$

The rotation matrix $S\text{-}R^0_3$ can then be derived from $S\text{-}T^0_3$ and can be used according to Equation 6 as follows:

$$S\_R^3_6 \;=\; (S\_R^0_3)^T \, S\_R_6 \qquad\qquad [54]$$

From this, the final robot joint variables can be calculated as the following set of Euler angles:

$$\theta_4 = \begin{cases} \tan 2^{-1}\left(r_{33}, \sqrt{1-(r_{33})^2}\ \right)\ \ or \\ \tan 2^{-1}\left(r_{33}, -\sqrt{1-(r_{33})^2}\ \right) \end{cases} \qquad\qquad [55]$$

$$\theta_5 = \begin{cases} \tan 2^{-1}(\,r_{13},\ r_{23})\ \ or \\ \tan 2^{-1}(-\,r_{13},\ -\,r_{23}) \end{cases} \qquad\qquad [56]$$

$$\theta_6 = \begin{cases} \tan 2^{-1}(-\,r_{31},\ r_{32})\ \ or \\ \tan 2^{-1}(r_{31},\ -\,r_{32}) \end{cases} \qquad\qquad [57]$$

In the above calculations, the r_{13} to r_{33} values refer to the specific indices of the $S\text{-}R^3_6$ matrix. With these equations, the inverse kinematic calculations are concluded.

D.4   Encoder value conversions

In this section, information with regards to the joystick and robot encoders is given. Their specific resolutions and how this information relates to the encoder values at the joystick and robot zero positions are of importance. For this reason, Table 19 and Table 20 below were created. Table 19 and Table 20 each contains data for each DOF encoder on the joystick and the robot. The specific encoder names are given, as well as their respective resolution ratings. If there is a ratio between the encoder shaft and the specific joint's output shaft, this is given, along with the final resolution of the combined system.

Then Table 21 and Table 22 are applicable to the joystick and robot joint variable values at different states. The joint variables as they appear on the joystick and robot models are given, together with their conversions into encoder units. The joint variable values of the respective physical zero positions are also given, again along with the corresponding value in encoder units. One can see a difference between the units of pulses/rev and lines/rev, which is explained as follows: at 10 bit resolution, 1 line equals 4 pulses, as all four edges (rising and falling) of a full quadrature line cycle are counted; at 9 bit resolution, 1 line equals 2 pulses, as only the rising or only the falling edges of a full quadrature line cycle are counted.

Table 19: The joystick encoder resolution data

DOF	Encoder	Resolution	Ratio	Final resolution
1	AS5040	1024 pulses/rev	1:1	1024 pulses/rev
2	AS5040	1024 pulses/rev	1:1	1024 pulses/rev
3	AS5040	1024 pulses/rev	1:1	1024 pulses/rev
4	AS5040	1024 pulses/rev	1:1	1024 pulses/rev
5	AS5040	1024 pulses/rev	1:1	1024 pulses/rev
6	AS5040	1024 pulses/rev	1:1	1024 pulses/rev

Table 20: The robot encoder resolution data

DOF	Encoder	Resolution	Ratio	Final resolution
1	Hengstler	3600 lines/rev	1:154	2217600 pulses/rev
2	Hengstler	3600 lines/rev	1:154	2217600 pulses/rev
3	UniMeasure	9.7 lines/mm	1:1	19.4 pulses/mm
4	Faulhaber	16 lines/rev	1:1526	97664 pulses/rev
5	Faulhaber	16 lines/rev	1:1526	97664 pulses/rev
6	Faulhaber	16 lines/rev	1:1526	97664 pulses/rev
7	Faulhaber	16 lines/rev	1:1526	97664 pulses/rev

Table 21: Conversions for joystick encoder data

DOF	Model position joint variable		Zero position joint variable	
	Angle	Encoder units	Angle	Encoder units
1	90°	256	170.2°	484
2	-90°	-256	86.5°	-246
3	90°	256	92.1°	262
4	0°	0	4.2°	12
5	180°	512	260.9°	742
6	0°	0	-87.9°	-250

Table 22: Conversions for robot encoder data

DOF	Model position joint variable		Zero position joint variable	
	Angle	Encoder units	Angle/Distance	Encoder units
1	-90°	-554400	-78.7°	-484670
2	90°	554400	66.5°	409640
3	-90°	-554400	610 mm	11834
4	90°	554400	0	0
5	0°	0	0	0
6	0°	0	0	0

APPENDIX E: TESTING ASPECTS

Appendix E contains the data that was gathered during the testing procedures outlined in Chapter 10. A technical specification sheet for the final product is also shown.

E.1   Testing results

This section contains the data measured for the point to point movement control tests as well as the intuitiveness tests done by the four subjects. Please see Table 23 through Table 27 below.

Table 23: Point to point test times

Test #	Time_1 (s)	Time_2 (s)
1	8	14.63
2	9.63	17.99
3	7.45	13.57
4	9.22	15.38
5	6.53	12.56
6	8.59	14.05
7	7.47	13.16
8	7.35	14.55
9	8.05	12.71
10	10.03	17.05
11	7.97	11.21
12	9.45	18.15
13	11.4	16.22
14	6.24	10.68
15	7.54	11.1
16	6.96	14.25
17	6.23	13.62
18	6.5	14.29
19	7.72	10.27
20	6.2	17.51
Average	7.9265	14.1475

Table 24: Intuitiveness test - subject 1

Test #	Time_1 (s)	Time_2 (s)
1	16.78	18.25
2	13.03	11.37
3	6.33	11.25
4	9.7	19.5
5	10.45	14.45
Average	11.258	14.964

Table 25: Intuitiveness test - subject 2

Test #	Time_1 (s)	Time_2 (s)
1	6.25	22.04
2	3.64	12.66
3	6.43	16.88
4	3.5	19.02
5	4.65	17.14
Average	4.894	17.548

Table 26: Intuitiveness test - subject 3

Test #	Time_1 (s)	Time_2 (s)
1	12.38	19.19
2	7.75	17.07
3	7.93	18.19
4	11.05	8.95
5	7	11.53
Average	9.222	14.986

Table 27: Intuitiveness test - subject 4

Test #	Time_1 (s)	Time_2 (s)
1	8.66	12.21
2	8.48	14.71
3	5.8	8.97
4	5.99	20.24
5	7.1	18.27
Average	7.206	14.88

Table 28: The average intuitiveness test times

Subject #	Time_1 (s)	Time_2 (s)
1	11.26	14.96
2	4.89	17.55
3	9.22	14.99
4	7.21	14.88
Average	8.15	15.6

E.2   Technical specification sheet

In Chapter 10, the testing procedure delivered several technical specifications that describe the master-slave systems. They are described in Table 29 below.

116

Table 29: Technical specification sheet

1. PHYSICAL PROPERTIES OF OVERALL STRUCTURE		
Property		**Value/Specification**
Size	Length	1.9 m
	Width	1 m
	Height	1.7 m
Estimated mass		200 kg
Materials used		Steel, stainless steel, aluminium sheet metal, aluminium extrusions, motors, encoders, electrical wiring.
Main subsystems		Slave electronics platform SSM and PSM systems User interface platform

2. MAIN MECHANICAL COMPONENTS		
Subsystem	**Component**	**Quantity**
SSM	Parvalux PM95GWS Brushless DC motor	2
	Hengstler RI59-3600 rotary encoder	2
	Linak LA30 self-locking linear actuator	1
	UniMeasure JX-EP-20 linear encoder	1
	Zeroing microswitches	3
	Shafts, links and brackets	-
PSM	Faulhaber brushed DC motor (with gearhead)	5
	Faulhaber IE2-16 encoder	5
	Shafts, links, cables and brackets	-
Joystick	Miki-Pulley 112-02-11 electromagnetic brake	6
	Xiros® B180 deep grooved radial ball bearing	12
	Aluminium shafts, sheet metal, base parts.	-
Platforms and other	SSM electronics holder plate	1
	Joystick holder plate	1
	Joystick PCB holder plate	1
	SSM electronics box	1
	24 V / 3 kW power supply	1

3. ELECTRICAL AND ELECTRONIC COMPONENT SPECIFICATION				
Component	**Current rating/ Wattage**	**Speed rating**	**Torque/ Force rating**	**Supply voltage**
Parvalux DC motor	450 W	26 rpm	112 Nm	24 V
Hengstler rotary encoder	60 mA	10000 rpm	-	12 V
Linak linear actuator	14 A	60 mm/s	50 N	12 V
UniMeasure linear encoder	85 mA	-	-	12 V
Faulhaber DC motor	24.7 W	1 rpm	7 Nm	12 V
Faulhaber encoder	6 mA	-	-	12 V
Miki-Pulley brake	250 mA	10000 rpm	0.4 Nm	24 V

AS5040 rotary encoder	20 mA	30000 rpm	-	5 V
Joystick system (without brakes)	200 mA	-	-	24 V
Robot system (without motors or brakes)	200 mA	-	-	24 V

4. COMMUNICATION CAPABILITIES

Type	Between master and slave	Serial; direct 4-wire cable connection (between two Arduino boards)
	Between master or slave and PC	Serial; direct USB cable connection (between applicable Arduino board and PC)
Serial port setup	Data type	8-bit
	Parity	None
	Stop bits	1
	Flow control	None
	Baud rate	9600 bits/s

5. FUNCTIONALITY

Power supply	Input	Standard 220 V (AC, 50 Hz) wall socket
	Emergency stop	Two emergency stop buttons, one for master system and one for slave system.
User inputs	Toggle switches	Six switches to control joystick brakes
		One switch to zero robot
		One switch to put robot in operational mode 1
		One switch to put robot in operational mode 2
	Safety switch	Controls movement of robot end effector
	Gripper buttons	Controls open/close movement of gripper

APPENDIX F: PROJECT MANAGEMENT ASPECTS

This appendix describes the project management aspects that played an important role during the execution of this project. The budgeting aspects, referring mainly to the operation expenses, are considered.

Table 30 below is a summary of the total expenses for the user interface stage of the overall project. Expenses were divided into sections for mechanical and electronic parts.

Table 30: Project expenses (in ZAR)

Mechanical parts						
Joystick (material, bending, cutting)						964
Joystick plate (material, bending, cutting)						280
SSM electronics box and plate (material, bending, cutting)						500
New damping shocks						200
Electronic parts						
Mikipulley brakes						2800
Arduino Mega 2560						500
Arduino Mega shields						200
AS5040 chips and magnets						755
Encoder PCBs						462
Joystick PCB						470
Robot PCB						480
Joystick shield PCB						470
Robot shield PCB						480
Components						1412
(resistors, capacitors, LEDs, switches, relays,						
voltage regulators, ribbon cables, connectors)						
TOTAL						9973

This shows that, rounded up, the expenses totalled R10000. During the project proposal stage the total budgeted expenses for the above sections were R23600 and the allowed, client-specified maximum was R30000 (as determined during the project proposal stage of this project). It should be noted that, due to limited knowledge of the joystick's specific mechanical and electronic design at that time, the budgeted values were too high. But even if the project proposal over-budgeted by up to 100%, this would still result in the actual expenses being less than the budgeted value and less than the client-specified maximum. This shows successful budget management of the user-interface stage of the overall project.

When looking at the overall project budgeting, the R10000 can be added to the total expenses. The R125000 total for the PSM and SSM project stages can, for safety reasons, be subject to an inflation rate of approximately 5 % per year over five years, which gives an approximate total of R160000. This brings the overall

project's expenses up to R170000. This value is solely based on actual expenses for material and components and does not include engineering and workshop labour, which would increase the value considerably. Although labour is an important factor, a true indication of the cost of the product is still the actual expenses, which is evidently considerably lower than other commercial systems like the da Vinci robot. This shows that, excluding at this stage any future design iterations that might occur, the overall project is still achieving its objective of keeping the costs as low as possible.

REFERENCES

Arduino Mega 2560. Datasheet. [S.a.]. [Online]. Available: http://arduino.cc/it/Main/ArduinoBoardMega2560. [2012, July 20].

Arduino EasyTransfer. [S.a.]. [Online]. Available: http://www.billporter.info/easytransfer-arduino-library/. [2012, July 20].

Arduino Encoder. [S.a.]. [Online]. Available: http://www.pjrc.com/teensy/td_libs_Encoder.html. [2012, July 20].

Arduino MatrixMath. [S.a.]. [Online]. Available: http://www.arduino.cc/playground/Main/MatrixMath. [2012, July 20].

Arduino Software. [S.a.]. [Online]. Available: http://arduino.cc/en/Main/Software. [2012, July 20].

Arduino Timer. [S.a.]. [Online]. Available: http://arduino.cc/playground/Code/Timer. [2012, July 20].

AS5040. Datasheet. [S.a.]. [Online]. Available: http://www.ams.com/eng/Products/Magnetic-Encoders/Rotary-Encoders/AS5040. [2012, July 3].

Ascension. [S.a.]. [Online]. Available: http://www.ascension-tech.com/medical/trakSTAR.php. [2012, July 3].

Camarillo, D.B., Krummel, T.M. & Salisbury, J.K. 2004. Robotic technology in surgery: past, present and future. *The American Journal of Surgery* 188: S2-S15.

Childress, V.W. 2007. Robotic surgery. *The Technology Teacher* 66.5: 9-13.

Christiane, P-J. 2008. *Development of a Minimally Invasive Robotic Surgical Manipulator.* Published MSc Eng Thesis. Stellenbosch University.

Christiane, P-J, Schreve, K & Scheffer, C. 2010. Development of a minimally invasive robotic surgical manipulator. *R & D Journal of the South African Institution of Mechanical Engineering* 26: 27-33.

da Vinci system. [S.a.]. [Online]. Available: http://thecoolgadgets.com/wp-content/uploads/2010/03/Da-Vinci-Surgical-Robot-Surgical-practice.jpg. [2011, March 5]

Dai, J.S. 2010. Surgical robotics and its development and progress. *Robotica* 28: 161.

EndoWrist design. [S.a.]. [Online]. Available: http://www.urologyspecialistskc.com/images/robotgif.gif. [2011, March 5].

Fastrak. [S.a.]. [Online]. Available:
http://www.inition.co.uk/3D-Technologies/polhemus-fastrak. [2012, July 3].
Hibbeler, R.C. 2005. *Mechanics of Materials.* SI Second Edition. Singapore:
Prentice-Hall, Inc.

Igus. [S.a.]. [Online]. Available:
http://www.igus.co.za/wpck/default.aspx?Pagename=xiros_B180&CL=GB-en.
[2012, July 2].

Intuitive Surgical Inc. Official website. [S.a.]. [Online]. Available:
http://www.intuitivesurgical.com/. [2011, February 24].

L78xx Voltage Regulator. [S.a.]. [Online]. Available:
http://docs-europe.electrocomponents.com/webdocs/0daf/0900766b80daf4ad.pdf
[2012, July 5].

Li, K, Chen, I, Yeo, S.H., Lim, C.K. 2011. Development of finger-motion capturing
device based on optical linear encoder. *Journal of Rehabilitation Research &
Development* 48: 69-82.

Lobontiu, A & Loisance, D. 2007. Robotic surgery and tele-surgery: basic
principles and description of a novel concept. *Jurnalul de Chirurgie* 3.3: 208-214.

Mack, M.J. 2001. Minimally Invasive and Robotic Surgery. *The Journal of the
American Medical Association* 285.5: 568-572.

Magnetic Rotary Encoder 1.0. [S.a.]. [Online]. Available:
http://www.reprap.org/wiki/Magnetic_Rotary_Encoder_1.0. [2012, February 2].

Miki-Pulley. [S.a.]. [Online]. Available:
http://www.mikipulley.co.jp/en/product/model.php?id=pcbmi&code=003.
[2012, July 2].

Phee, S.J.*, Low, S.C., Huynh, V.A., Kencana, A.P., Sun, Z.L., Yang ,K. 2009.
*Master And Slave Transluminal Endoscopic Robot (MASTER) for Natural Orifice
Transluminal Endoscopic Surgery (NOTES).* 31st Annual International
Conference of the IEEE EMBS. Minneapolis, Minnesota, USA, September 2-6.
pp.1192-1195.

Polhemus. [S.a.]. [Online]. Available:
http://www.polhemus.com/?page=Motion_Fastrak. [2012, July 3].

Queirós, P, Cortesão, C &, Sousa C. 2010. *Haptic Tele-Manipulation for Robotic-
Assisted Minimally Invasive Surgery with Explicit Posture Control.* 18th
Mediterranean Conference on Control & Automation. Marrakech, Morocco, June
23-25. pp.808-814.

Reed. [S.a.]. [Online]. Available:
http://docs-europe.electrocomponents.com/webdocs/0033/0900766b8003309.pdf
[2012, July 5].

RW-America. [S.a.]. [Online]. Available:
http://www.rw-america.com/products/torque_limiters/classic/sk5/. [2012, July 8].

Sensable technologies. [S.a.]. [Online]. Available:
http://www.sensable.com/. [2012, June 19].

Spong, M.W., Hutchinson, S, Vidyasagar, M. 2006. *Robot modeling and control*.
1st Edition. Hoboken: John Wiley & Sons, Inc.

Surgeon's console. [S.a.]. [Online]. Available:
http://www.lakelandhealth.org/images/davinci_images/InSite.jpg. [2012, June 19].

Tavakoli, M, Patel, R.V. & Moallem, M. 2003. *A Force Reflective Master-Slave
System for Minimally Invasive Surgery*. Proceedings of the IEEEIRSI International
Conference on Intelligent Robots and Systems. Las Vegas, Nevada, October.
pp.3077-3082.

TraX. [S.a.]. [Online]. Available: http://www.trax.co.za/. [2012, June 19].

Van den Bedem, L, Hendrix, R, Rosielle, N, Steinbuch, M &Nijmeijer, H. 2009.
*Design of a Minimally Invasive Surgical Teleoperated Master-Slave System with
Haptic Feedback*. Proceedings of the IEEE International Conference on
Mechatronics and Automation. Changchun, China. August 9 - 12. pp.60-65.

Van der Meijden, O.A.J. & Schijven, M.P. 2009. The value of haptic feedback in
conventional and robot-assisted minimal invasive surgery and virtual reality
training: a current review. *Surgical endoscopy* 23: 1180-1190.

Worst, S.C. 2012. *Development of a Low Cost Secondary Slave Manipulator for
a Minimally Invasive Robotic Surgical System*. Unpublished MSc Eng Thesis.
Stellenbosch University.

Yash, S. 2008. *The DaVinci Robot: Reducing Recovery Time and Medical Costs
in the Operating Room*. [Online]. Available:
http://www.jyi.org/news/nb.php?id=1499. [2011, March 10].