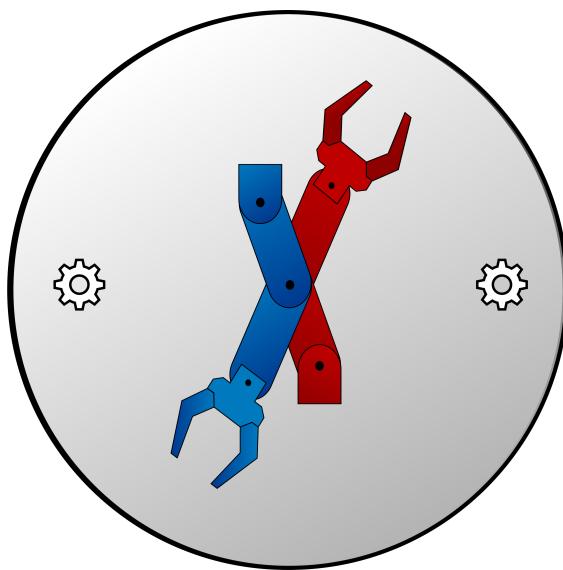


GROUP CONCEPTUAL DESIGN

Trey Dufrene, Alan Wallingford, David Orcutt, Ryan Warner, Zack Johnson



ME 407

Preliminary Design of Robotic Systems

Embry-Riddle Aeronautical University



Meiosis

Contents

1	Introduction	1
2	Physical System Overview	1
2.1	Base	2
2.2	Links	3
3	System Functions	4
3.1	Electrical	4
3.2	Software	5
4	Parts List and Manufacturing	6
5	Decision Matrices	8

List of Figures

1	Overall System Conceptual Design	1
2	Manipulator Base with Call-outs	2
3	Drawing Showing Key Features of Design	3
4	Drawing Showing Link Cross Section	3
5	Electrical System Block Diagram	4
6	Software Flowchart	5

List of Tables

1	Predicted List of Parts	6
2	Gantt Chart	7
3	End Effector Attachment Design Decision Matrix	8
4	Material Choice Decision Matrix	8
5	Computing Choice Decision Matrix	9

1 Introduction

The terminator T-2000 is a science-fiction spectacle of a robot – until you see the price. Channelling the inspiration many high school students may have for robotics, MEIOSIS robotics aims to provide an affordable manipulator to educators and enthusiasts. MEIOSIS uses primarily 3-D printed components and easily accessible materials. Among these materials are a Raspberry PI, smart servos and metal tubing. These features create an open-source manipulator accessible to the public to further robotics education.

2 Physical System Overview

The physical design of the robotic manipulator will be shown through Figures 1, 2, 3, and 4.

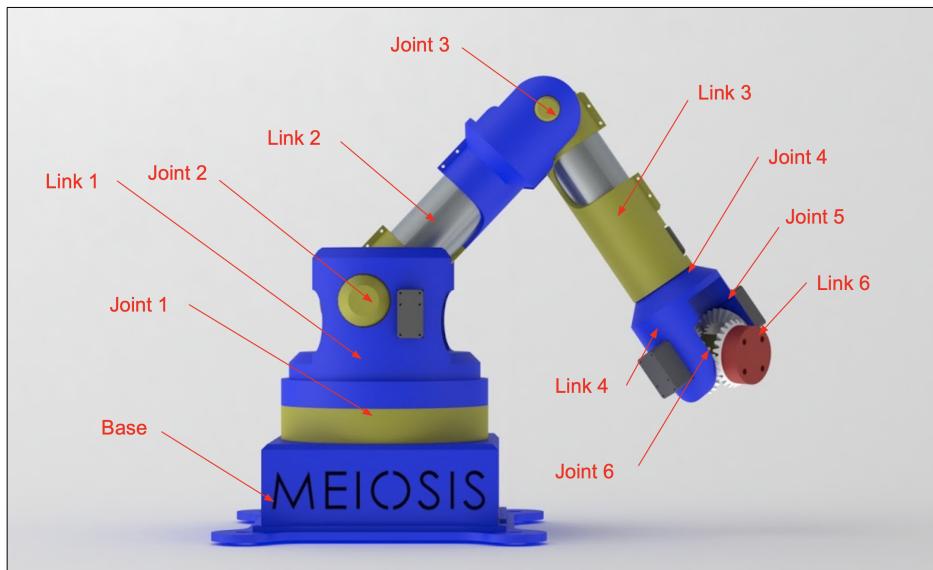


Figure 1: Overall System Conceptual Design

The colored links in *Figure 1* distinguish the different joints and links of the manipulator. The overall reach of the robot will be 582.5 mm. This length was chosen to decrease material cost and weight while still satisfying requirement 2.1.2 and 2.1.5, allowing the manipulating to pick and place objects to perform basic tasks. The base of the robot will be made to contain the Raspberry Pi and other electrical components.

2.1 Base

The base of the manipulator will house several of the electronic components, such as the computational system, power supply, and motor controller. A cross section of the base can be seen in *Figure 2*.

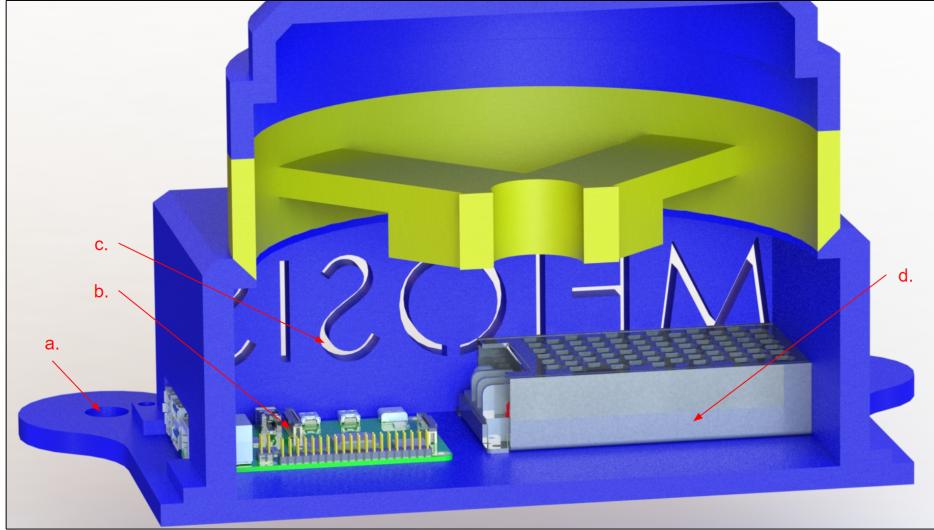


Figure 2: Manipulator Base with Call-outs

From *Figure 2*,

- a. *Base Supports*: The base supports are located at each corner of the base and will allow the base of the manipulator to be securely attached to a variety of surfaces with either standard bolt/fastener hardware or suction cups.
- b. *Computational System*: The computational system will be a Raspberry Pi; it will be housed in the base, which allows the Raspberry Pi to be more easily accessible. The primary reason for this system being chosen is to fulfill the budget requirement, 2.1.1. The Raspberry Pi will compute the manipulator's kinematics and command the motors accordingly.
- c. *Airflow Cutouts*: The side of the base will have cutouts to allow for airflow through the base; since the power supply is housed inside of the base as well as the computational system, the temperature must be regulated to prevent overheating.
- d. *Power Supply*: The power supply will be housed in the base as well; this allows the power supply to be more accessible and therefore more modifiable, so the end-user can easily expand the system to fulfill their needs.

2.2 Links

Figure 3 is an image of the robot that shows the links and their key features.

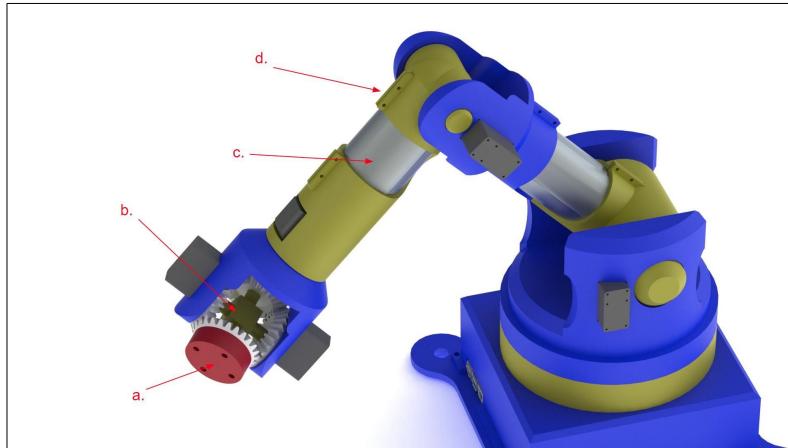


Figure 3: Drawing Showing Key Features of Design

Figure 3 highlights a few of the key features of our design. Call-out a shows the connection point for the end effector. The mountings are the standard used by the Sawyer manipulator. This may be adjusted to accommodate lower cost, more accessible end effectors. Call-out (b) shows the differential gearbox that will be used in the manipulator's wrist, saving space and weight. The manipulator will have aluminum tubing as support in the links (c) and will be attached to the 3D printed portion of the robot using clamp joints (d) tightened by screws.

Figure 4 is an image of the cross section of link 2 for the manipulator.

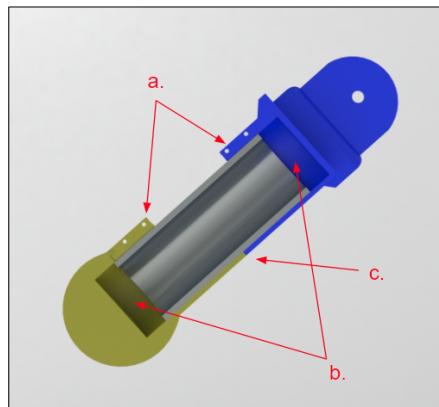


Figure 4: Drawing Showing Link Cross Section

The cross section seen in *Figure 4* shows the internal design for links two and three. It features two clamps that hold a hollow aluminum bar in place (a) and allows for gaps between the aluminum tube and the 3D printed call-out (b). The proper length will be dictated by the 3D printed guides lining up at call-out (c). This allows for imprecision in the manufacturing of the aluminum tube.

3 System Functions

The system can be divided into two subsystems: the electrical and software systems. The electrical subsystem includes the wiring and hardware computational components, power system, actuators with drivers, and sensors. The software subsystem includes the algorithm for the computational system.

3.1 Electrical

Figure 5 is the block diagram for the electrical system of the manipulator.

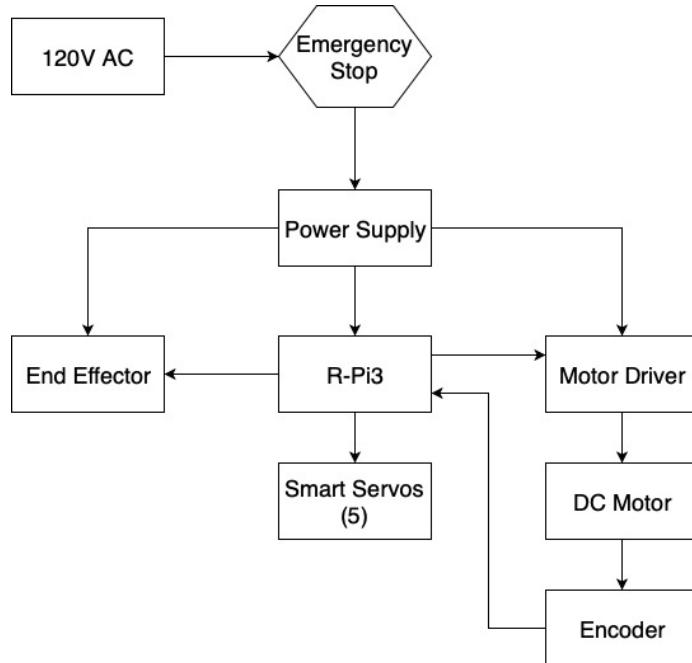


Figure 5: Electrical System Block Diagram

Figure 5 shows that the electrical systems of the manipulator will be relatively simple. Power

is supplied by the 120V AC from standard wall outlets. A power supply will adapt the AC voltage to the required voltages for each component. One component is the Raspberry Pi, which will perform calculations for motor control (described below in software). It will send signals to the DC motor driver and the five smart servos. The smart servos have an on-board controller, so no feedback will be necessary. However, the first joint, between the base and the first link, will be actuated by a DC motor with an encoder to minimize cost.

3.2 Software

Figure 6 shows the software flowchart for the system.

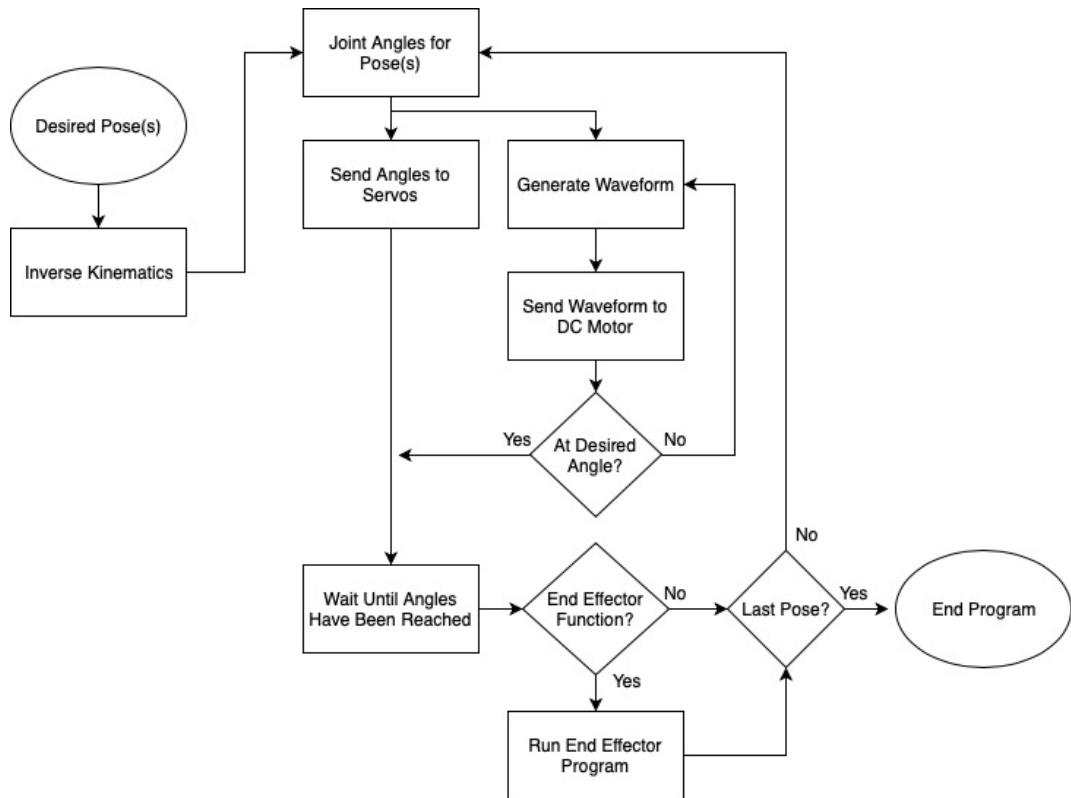


Figure 6: Software Flowchart

Similar to the electrical system, the software is also simple. *Figure 6* shows that the software will receive the desired pose or poses the user would like the manipulator to reach. Then the Raspberry Pi will use inverse kinematics to calculate the necessary joint angles. The waveforms/desired angles will be sent to the respective drivers/motors, and positional information will be sent back to the Raspberry Pi to adjust the DC motor angle. When the motors have

reached their desired pose, the Raspberry Pi will actuate the end effector if it is specified by the user. The system will then check to see if there are any more poses to reach and either repeat the motor control section given the desired angles of the new pose or end program if the last pose has been reached.

4 Parts List and Manufacturing

Our costs are estimated from all specific components we envision requiring with a factor of safety. The 20% factor of safety is the sum of 20% of the cost for each item listed in Table 4.1 and is intended to account for shipping, taxes, and replacement/additional parts. Including this factor of safety, we satisfy requirement 2.1.1 with a predicted budget of \$798.80.

Table 1: Predicted List of Parts

Part	Unit Cost (USD)	Qty	Cost (USD)	Source
2kg PLA	32	1	32	Amazon
Smart Servo	140	2	280	Trossen Robotics
Lower Torque Smart Servo (6 pack)	225	1	225	Trossen Robotics
DC Motor w/ Encoder	30	1	30	Robot Shop
Power Supply	16	1	16	Compt. Dist. Inc.
Raspberry Pi 3	35	1	35	Adafruit
3 Pin Cables	9	1	9	Amazon
4 Pin Cables	13	1	13	Amazon
Gripper	25	1	25	Trossen Robotics
Factor of Safety (20%)			133	
Total			\$798	

Our motor costs are approximated from three motor types. For use in joint one is one stepper motor [3]. It should experience minimal torque from gravity acting on the other links and does not require a continuous voltage to maintain position. The built-in encoder provides feedback for accurate motion and limiting the rotation to 2π radians. For use in joints 2 and 3 are two smart servos with holding torque of $1.6\text{N}\cdot\text{m}$. These feature a built in PD control, which means no overshooting, that can be precise to 0.29° with tuning [1]. So

long as the links, servos, and payload after the base do not exceed 0.65kg acting through the horizontally outstretched arm's centroid, 250mm from the base, the torque is sufficient. Although the servos are continuous rotation, joints 2 and 3 are not free to rotate 360° and will be limited by software. Finally, for the spherical wrist are three lower cost, lower torque, smart servos which also have PD control and a 0.29° resolution [2]. The gripper will also be actuated with one of these servos.

Since each motor has the same voltage requirement of 12V and our manipulator is powered by 120V AC, we also account for the cost of a power supply. The power and communication buses are handled by four pin wires for the smart servos and three pin wires for the lower-cost smart servos. 2kg of PLA filament allows all components of the manipulator to be 3D printed with 100% infill, making it an overestimate. With a 20% factor of safety to account for shipping, taxes, and unexpected part purchases, we are under our team budget of \$800 and design budget of \$1000.

Table 2 shows the Gantt chart for the remainder of the fall semester.

Table 2: Gantt Chart

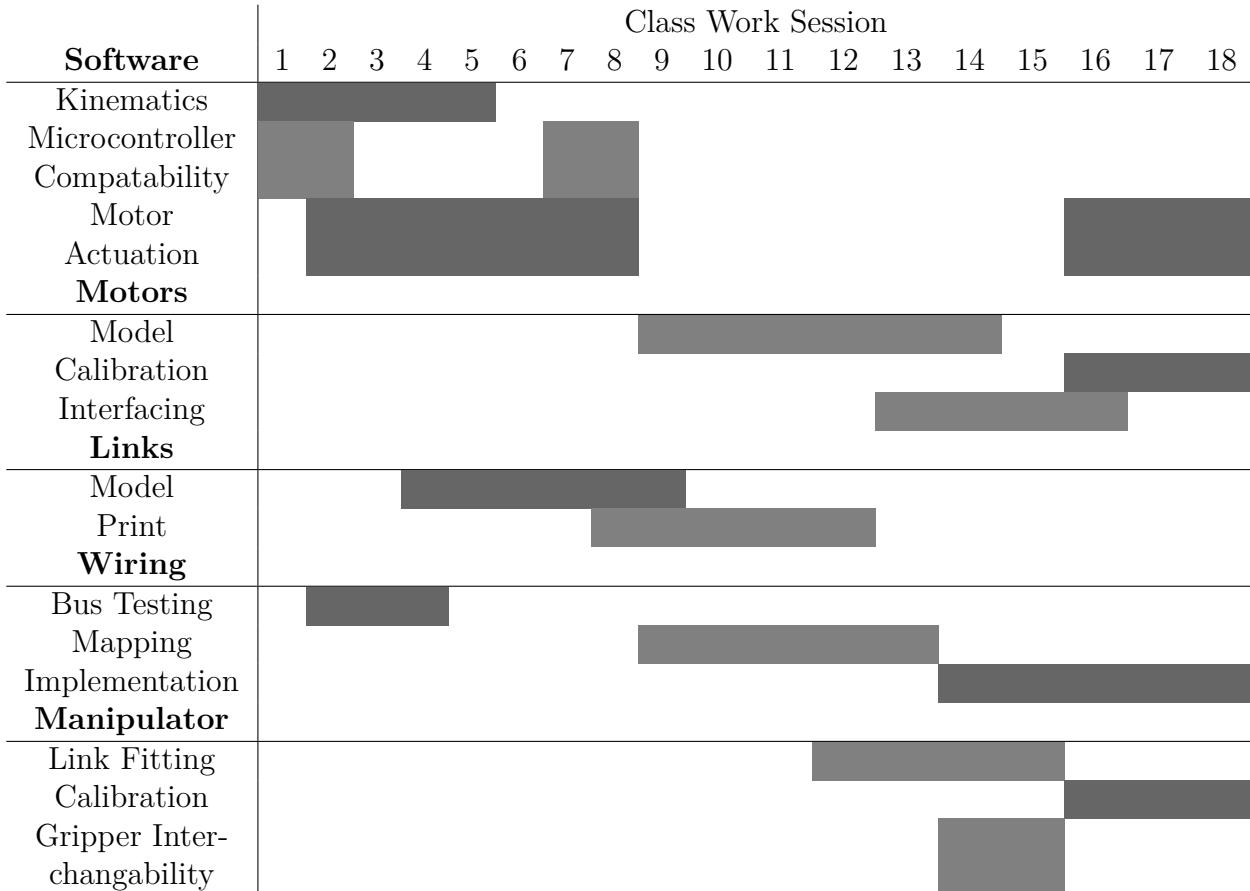


Table 2 shows approximately half our time for Fall semester will be used modeling the manipulator's kinematics, motor dynamics, and physical links with the second half dedicated to interfacing parts, wiring, and testing components. Nonetheless, the manipulator should have a functional guise by the semester's end, however, we anticipate our design will require further iterations before it satisfies all our requirements. For example, a refined control interface usable by novice robotics students is not accounted for.

5 Decision Matrices

The design for the end effector attachment varied across each of the team member's original conceptual designs, therefore a decision matrix was constructed to objectively decide on the best design for the manipulator.

Table 3: End Effector Attachment Design Decision Matrix

EE Attachment Weighting	Ease of Use	Manufacturability	Durability	Total
	3.6	5	7	
Screw Connections	6.8	8.8	9.8	137.08
Snap Fit Joint	8.6	5	2.2	71.36
Threaded End Effector	6.3	5.4	7.3	100.78

As shown in *Table 3*, the optimal design must be somewhat easy to use, easily manufacturable, and the most durable. This is the best represented by the screw connections design.

Table 4: Material Choice Decision Matrix

Material Weighting	Cost	Weight	Accessibility	Manufacturability	Durability	Total
	9.2	6.6	8	6.4	5.4	
3D Printed	8	9	8	9.4	5	284.16
Aluminum	4	5.8	5.8	6.6	9.2	213.4
Combination	8	7.8	7	9	9.2	288.36

As seen in *Table 4*, the primary material chosen to create the manipulator is PLA / 3D printed material, this is due to its low cost and accessibility; additionally, it is lightweight, manufacturable, and durable. Further, by using primarily 3D printed parts, the end-user's cost is potentially lower and their ability to modify parts is enhanced.

Table 5: Computing Choice Decision Matrix

Computing System Weights	Cost	Capabilities	Ease of Use	Independance	Total
	10	4.2	7.6	4.2	
RPi	7.9	6	9	10	214.6
Jetson	1	7	7	9	130.4
RPi + Arduino	4.8	7	8	10	180.2
Arduino	9	4	9	3	187.8
B Black	1.2	7	7	10	136.6
B Green	5	7	6	9	162.8

Table 5 shows the decision matrix used to determine the computing system. The most important attributes are cost and ease of use, being that the target end-user will be programming of system and the highest priority requirement is the system's final cost. Functionality beyond motor control and inverse kinematic calculations are not eminently important and operating independent of another computer is only desirable.

References

- [1] Nema 17 stepper motor 1.8. URL https://www.matterhackers.com/store/1/nema-17-stepper-motor/sk/M1C4A52S?rcode=GAT9HR&gclid=EAIAIQobChMIkeK0-fr25AIVeP7jBx1RXw0VEAkYASABEgLPzvD_BWE. Last accessed 30 September 2019.
- [2] Smart robot servo. URL <http://www.revrobotics.com/rev-41-1097/>. Last accessed 30 September 2019.
- [3] Dynamixel rx-24f robot actuator. URL <https://www.trossenrobotics.com/dynamixel-rx-24F-robot-actuator.aspx>. Last accessed 30 September 2019.