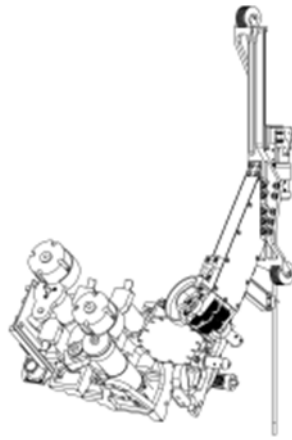


# EE 449 Project Milestone Report IV

---

## Automatic Cable Winding For Surgical Robot Arms



May 21, 2010

Group 5: Imam Tjung and Kiran Thomas

## **Table of Contents**

- 1. Introduction**
- 2. Project Description**
- 3. Project Update**
- 4. Demo Goals**
- 5. Symbols and units**
- 6. Updated System Diagram**
- 7. Hardware**
- 8. Software**
- 9. Experiment Result and Stability Issue**
- 10. Technical Obstacles**
- 11. Team Management**
- 12. Conclusion**
- 13. Bibliography**

## 1. Introduction

In designing a control system the general flow of tasks are system modeling, simulation, control design, controller performance and robustness testing. This milestone report covers the progress of the Automatic Cable Winding for Surgical Robot Arms project after seven weeks. The purpose of this report is to show the progress of the project leading to the fourth milestone. The objective of the fourth milestone is to begin implementation of the system and finalize the hardware. The position control on the motor with the cable on has been successfully coded. The next challenge is to code in the velocity control. A new hardware has been added to the project as requested by the customer. The final stage of the project will involve collecting the data from the various motors on the robot and entering a schedule into the code.

## 2. Project Description

The project originated from the problem of winding cable drivers on the Bio Robotics Lab's surgical robot RAVEN [6]. The cable drivers' capstan has to be wound by hand when it is built and also when it is re-cabled after maintenance. The drivers are located at hard to reach places on the arm and the existing method of cabling requires two people.

The solution was to create a controller for the motor connected to the driver capstan. The controller should be able to rotate the capstan at a specified velocity for a certain number of turns. Since the cable has to be wound taught, the other end of the cable will be under constant tension by a person holding the cable. Since the motor does not have to be wound under a specific tension. The system should be able to wind the cable in both clockwise and counter clockwise direction because the cable system of the robotic arm uses cables wound in opposite directions. The user should also be able to specify the velocity and number of turns within a range.

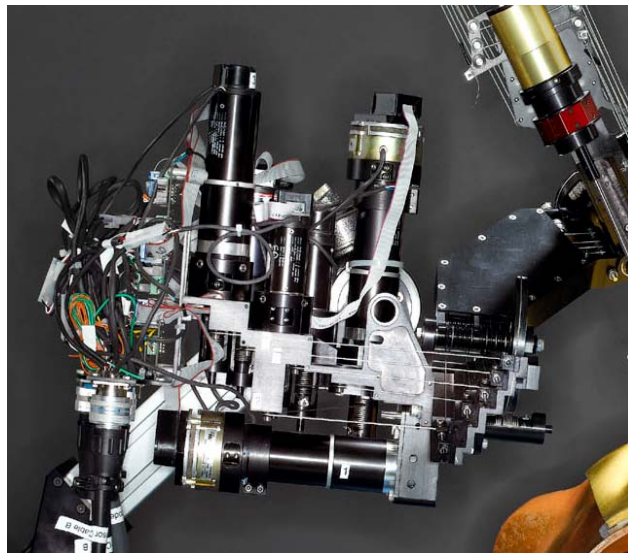


Figure 1 Raven Surgical Robot Close up

### 3. Project Update

After communicating with the customer the following changes were made to the project:

- Since this system will be mainly used by a student in UC Santa Cruz who is building seven robots, having specific hardware for this system is impractical.
- Design the best possible controller assuming no torque and make sure the resistance provided by a human is within the range of the motor capability.
- Use a foot pedal with three pedals to control the working of the controller. This will replace three signals from the keyboard.

### 4. Demo Goals

For the final demo of the project our goal is the following:

1. Demonstrate the cabling of a motor capstan on the pulley board.
2. Submit the code along with a user manual.
3. Include information for implementing the controller on all the surgical arm's capstans.

### 5. Symbols and units

The symbols and their corresponding units used in this report are:

Input voltage  $V_a(t)$  (V)  
Current  $i$  (Amps)  
Load torque  $T_L(t)$  (N-m)  
Torque constant  $K_T$  (N-m/A)  
Speed constant  $K_v$  (V/(rad/sec))  
Back emf voltage  $e(t)$  (volts)  
Viscous friction  $B_m$  (N-m)  
Motor terminal resistance  $R_m$  ( $\Omega$ )  
Motor terminal inductance  $L_a$  (H)  
Motor torque  $T(t)$  (N-m)  
Motor angle  $\theta$  (rad)  
Angular velocity  $\omega$  (rad/sec)  
Amplifier Gain  $K_A$   
Rotor + capstan inertia  $J_m$  (kg-m<sup>2</sup>)

## 6. Updated System Diagram

The figure below shows the updated diagram of all the hardware parts of the system. The initial design involved a source of constant torque to keep the cable taut. After the customer's recommendation we removed the extra hardware and decided to design a system that will be able to handle the force of a person when they hold the cable.

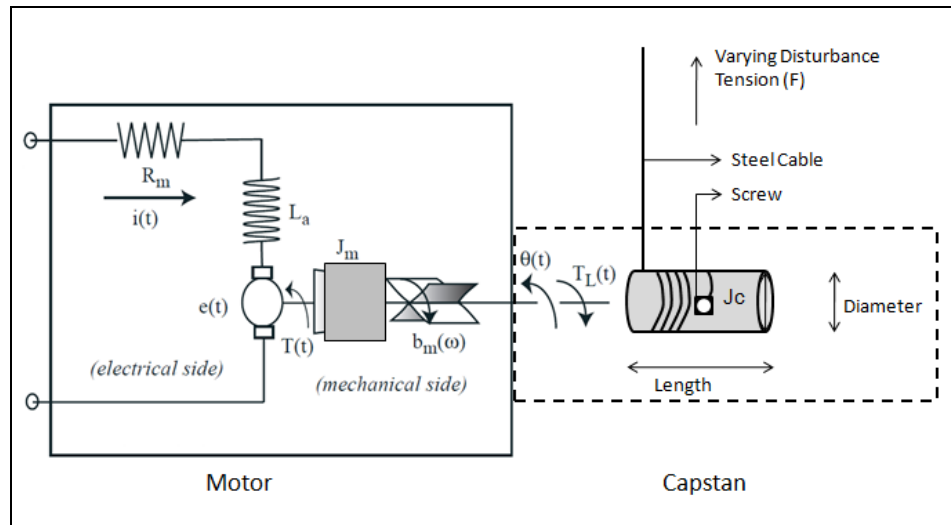


Figure 2. System parts mechanical diagram

## 7. Hardware

The hardware components of the project are from the actual robot and the controller is being tested on a pulley board which has the same hardware. The descriptions of each are given below:

Motor:

The motor is a MAXON brushless DC motor. It works with a motor controller which can also acts as an amplifier. The motor has a power rating of 120W with a max stall torque of 0.7 Nm. The remaining motor data is available in the data sheet attached in the appendix.

Encoder:

The encoder is built onto the motor and is also powered by the motor controller.

Capstan and Cable:

The capstan is a steel cylinder that is welded onto the motor shaft. The cable is a 4mm thick steel cable. The inertia of the capstan and cable combination was calculated based on the dimensions and included in the system model.

Figure 3 below shows the close up of the parts mentioned above.

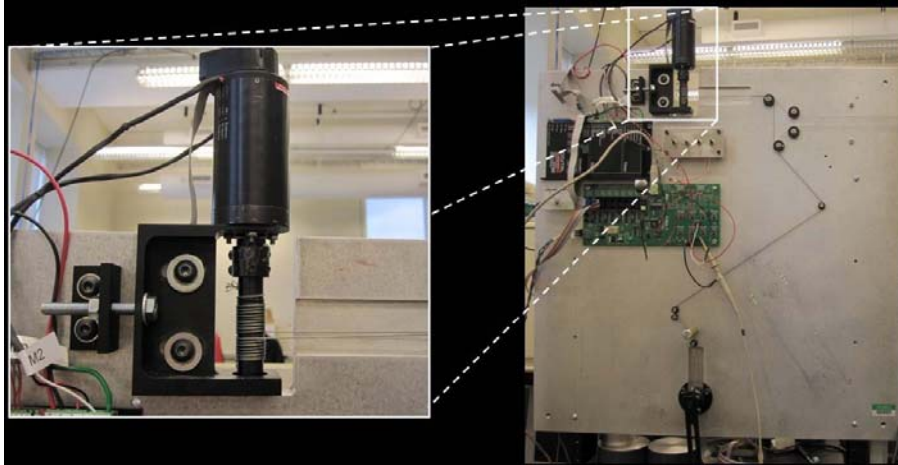


Figure 3. Photo showing a closeup of the motor and capstan on the pulley board

#### Foot pedal:

Since under the new design the user will be cabling the capstan by hand, typing commands or using the keyboard during cabling is impractical. Since haptic devices used in the BRL lab use a USB foot pedal, we are including that into our system. The pedal is an extension of the keyboard and has three pedals. Each can be used for a different purpose. For example one can be used to start rotating in one direction, second for the opposite direction and third for emergency stop. Figure 4 below shows a photograph of the foot pedal.



Figure 4. Foot pedal

## 8. Software

The software used for this project is run on a Fedora three Linux machine. The code has been written in C. Below shows the portions of the C code that needed to rotate with position control implementation.

```
#ifdef P_CONTROL
```

```
j1->tau_d = KP * (j1->mpos - j1->mpos_d); //- P-control
```

The position control gets implemented by changing the current torque (j1->tau\_d) that proportional to the position. The equation for current torque is (actual position (j1->mpos) – desired position (j1->mpos\_d) times the position gain.

```
#ifndef ROTATE
#define PI_TO_ANGLE PI / 180
#define ROTATION_ANGLE 5
#define NUMBER_OF_ROTATION 1 * (360 / ROTATION_ANGLE)
#endif
```

PI\_TO\_ANGLE convert the calculation from PI to angle. The user can specify the rotation each second (in degree) by changing the ROTATION\_ANGLE. And, the NUMBER\_OF\_ROTATION allows the user to put the desire number of rotation. The code above shows that the number of rotation is one.

```
#ifndef ROTATE
// rotate from 0 to desire position
j1->mpos_d = 0;
double lastSec = 0;
if (j1->mpos_d <= PI){
    j1->mpos_d = rotAngle * piToAngle * secs;
    lastSec = secs; }
else{
    j1->mpos_d = - (PI - (rotAngle * piToAngle * (secs-lastSec))); }
if (secs > numbRotate){
    if (j1->mpos_d <= PI){
        j1->mpos_d = rotAngle * piToAngle * numbRotate;
    }
    else{
        j1->mpos_d = - (PI - (rotAngle * piToAngle * numbRotate));
    }
}
}
```

The rotation function gets implemented by changing the desire position (j1->mpos\_d). The first if statement is to check if the position is in positive (right side) or negative (left side). There is a double lastSec that used to remember the time it hits PI. The variable is needed for negative side calculation. The second if statement checks whether the time is already reach the number of rotation that the user want. If the condition is satisfied, then it should maintain the current position.

## 9. Experiment Result and Stability Issue

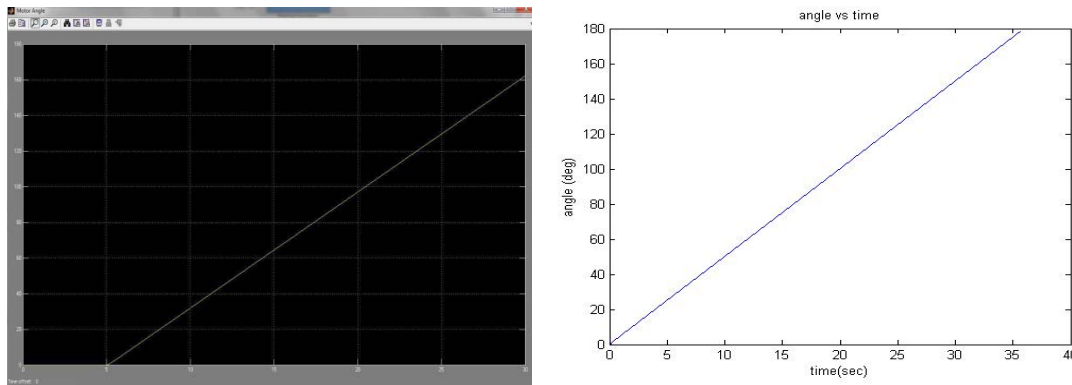


Figure 3. Simulation result (left) and experiment result (right).

The right figure shows the result of running the above code for about 35 second. The angle position is directly proportional to the time elapsed. Since the experiment result match with the simulation result, we believe that the position control is successfully implemented.

However, there is stability issue for just having position control. When we try to hold the capstan from rotating and suddenly release it, the capstan rotates so fast to the desire position. This will cause a problem if the person holds the cable too tight and suddenly loosen his grip. Thus, we need to have velocity control that will make sure that the capstan rotates at the constant velocity.

## 10. Technical Obstacles

The initial idea of having a source of constant torque for holding the cable taught during winding was rejected by the customer due to the extra hardware that was required to use that method. The customer informed us that the existing method to cable a capstan used a person's finger to press tightly on the cable while it wound up onto the capstan. This friction can be considered a disturbance and an approximate value can be measured for simulation purposes. Since the motor's maximum torque output is well above the frictional force, the controller will be able to continue winding.

## 11. Team Management

As the project gets closer to the final stages, we have noticed that the initial division of labor has become less clear and both the team members are working on each others areas. The next phase of implementation is to fine tune the system by testing and changing necessary code until it performs according to specification.

## 12. Conclusion

Since the quarter is nearing the end, the project will have to be completed and wrapped up soon. All the milestones for the project have been successfully completed so far. Even though the



project has undergone changes, the effect on the work done so far is minimal. The fourth milestone's purpose is to clearly define and justify hardware and software choices made for the final design. The final design choices which will be implemented are sure to change but only in a small way. Over the next two weeks we will be working to implement the system as specified by the customer and attain the best possible performance beyond the performance criteria.

### **13.Bibliography**

[1] Norman S. Nise, “Control Systems Engineering”, Wiley 2008

- The techniques for assessing controllability and observability were obtained from this book

[2] Uy-Loi Ly, “DC Motor Control”, 2010, [https://courses.washington.edu/aa448/DCMotorControl\\_LabDescription.pdf](https://courses.washington.edu/aa448/DCMotorControl_LabDescription.pdf)

- Used to determine the simulink simulation.

[3] “dcmmodel”, 2010. <https://courses.washington.edu/aa448/dcmmodel.pdf>

- Used to determine the State Space equation.

[4] “Maxon EC Motor Data Sheet”, 2010.  
<http://shop.maxonmotor.com/ishop/article/article/118898.xml>

- Used to get the model parameters

[5] Atef Saleh Othman Al Mashakbeh, “Proportional Integral and Derivative Control of Brushless DC Motor” European Journal of Scientific Research. ISSN 1450-216X Vol.35 No.2 (2009), pp.198-203

- Used the author’s method to model the BLDC motor.

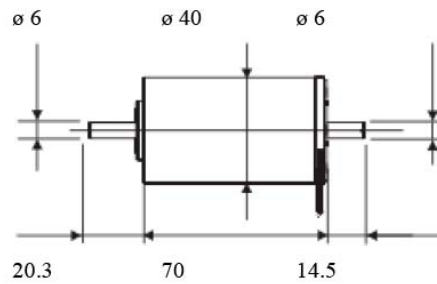
[6] J. Rosen, B. Hannaford, 'Doc at a Distance,' IEEE Spectrum, pp. 34-39, October 2006.

- The paper contained detailed information about the RAVEN projects and photographs

## Appendix A

Motor data sheet:

EC 40 Ø40 mm, brushless, 120 Watt, CE approved, with Hall sensors



Dimensions in mm

This schematic is not drawn to scale.

| Order No.                                 | Price in pc(s) excl. VAT              | Quantity |
|-------------------------------------------|---------------------------------------|----------|
| 118898                                    | USD 462.82                            | 1        |
| <b>Motor data</b>                         |                                       |          |
| Assigned power rating                     | W                                     | 120      |
| Nominal voltage                           | V                                     | 48       |
| No load speed                             | min <sup>-1</sup>                     | 5930     |
| Stall torque                              | mNm                                   | 742      |
| Max. continuous torque                    | mNm                                   | 130      |
| Speed / torque gradient                   | min <sup>-1</sup> / mNm <sup>-1</sup> | 8.3      |
| No load current                           | mA                                    | 97.8     |
| Starting current                          | A                                     | 9.72     |
| Terminal resistance                       | Ohm                                   | 4.94     |
| Max. permissible speed                    | min <sup>-1</sup>                     | 18000    |
| Nominal current (max. continuous current) | A                                     | 743      |
| Max. efficiency                           | %                                     | 80.1     |
| Torque constant                           | mNm / A <sup>-1</sup>                 | 76.4     |
| Speed constant                            | min <sup>-1</sup> / V <sup>-1</sup>   | 125      |
| Mechanical time constant                  | ms                                    | 7.18     |
| Rotor inertia                             | gcm <sup>2</sup>                      | 85       |
| Terminal inductance                       | mH                                    | 1280     |
| Thermal resistance housing-ambient        | KW <sup>-1</sup>                      | 3.2      |
| Thermal resistance winding-housing        | KW <sup>-1</sup>                      | 1.2      |
| Thermal time constant winding             | s                                     | 16.7     |
| Motor length                              | mm                                    | 70       |
| Weight                                    | g                                     | 390      |