Final Report

Karl Brandt
RoboSat

EEL 4665/5666
Intelligent Machines Design Laboratory

Dr. A. Antonio Arroyo
Dr. Eric M. Schwartz

Mike Pridgen
Ryan Stevens
Tim Martin
Thomas Vermeer
Devin Hughes

Table of Contents

I. Abstract

RoboSat: a small satellite equipped with two dexterous manipulator has been designed, prototyped, and tested in an effort to unveil potential on-orbit servicing technology on a relatively effective and inexpensive platform. This first-generation proof-of-concept model has an immediate goal of showcasing its manipulator system that allows docking, "crawling", and manipulation of a target. This demonstration will not only address the first and arguably most crucial step in full on-orbit servicing – docking, but it will also seek to address maneuverability through consumption of renewable energy - electricity. Specifically, the experimental setup will consist of the RoboSat, a hovercraft, and a railing structure. While levitating on the hovercraft, the RoboSat will attempt to dock to the railing structure. Once docked, the RoboSat will begin climbing around on the structure by keeping one hand fixed to the structure and moving the other.

II. Executive Summary

With an increase in more sophisticated and bold space technology, there has been an increasing demand for on-orbit servicing capabilities. Many current projects seek to address this demand such as those involving the Orbital Express, MIT's SPHERES, and the Robonaut. Each of these technologies possesses their very own unique form for providing on-orbit servicing, and they have been an inspiration to yet another platform – the RoboSat.

RoboSat is a 6U Cubesat robotic satellite that seeks to bring on-orbit servicing closer to reality in its own way. Equipped with two dexterous manipulators on a small standardized satellite platform, the RoboSat will possess a unique blend of technologies while remaining relatively inexpensive. After full research and development, the RoboSat will ultimately be able to dock to and manipulate various structures in space. A practical scenario of particular interest is that of the RoboSat docking with another large satellite in geosynchronous orbit. Large satellites at these orbits are very expensive and important, and have consistently been plagued by mechanical failures. Many of these failures (such as an improperly deployed solar panel) could have been remedied relatively easily if on-orbit servicing were available.

There are many technological requirements needed for on-orbit servicing of such nature such as energy resources, high dexterity, cost-effectiveness, and seamless integration. RoboSat attempts to address all of these issues since it is a small satellite equipped with two onboard, robust manipulators, and a variety of sensors. Additionally, the whole package would be launched as a secondary payload, and in the case of it needing to dock to the primary payload, both of them will be relatively close upon ejection in space.

One of the first steps in developing RoboSat was to design a first generation, proof-of-concept model that will showcase some of the immediate functionality desired in this technology. This prototype is equipped with two arms, two force-sensing manipulators, two side-deploying panels, and an ultrasonic sensor. For experimentation, the model will be mounted on top of a hovercraft to provide a near-frictionless plane for two-dimensional testing. The immediate goal will be for RoboSat to dock to and "climb" around on a railing structure using its sensors and manipulators. In developing this prototype, insight will be gained on developed mechanisms and grippers and their advantages and disadvantages. This information will be used to further stimulate interest in the project as well as provide technical insight in its research and development.

III. Introduction

The need for on-orbit servicing for satellites is becoming increasingly evident with the rise in failure costs fueled by the demand for more complex systems. In order to compensate for the paradigm shift demanding the creation of satellites with higher functionality and longer orbital lifetimes, on-orbit servicing (OOS) for satellites becomes a necessity. When considering Geostationary Earth Orbit (GEO) satellites alone, post-launch on-orbit failures accounted for 66% of all failures between 1996 and 2002, a cost which totaled $4.24 billion. Nearly one-quarter of all failures are due to mechanical issues alone. This includes failed solar panel or antenna deployments, and attitude and orbit control systems (AOCS). Therefore, a near seamless and cost-effective solution is needed to help make the transition to on-orbit servicing technology a reality. The current strategy for this project is to design a proof-of-concept small satellite that can dock and manipulate its target with onboard, dexterous robotic manipulators. Unlike current technology, these robotic manipulators will not possess intelligent ports for docking; instead, they will possess fingers in a specialized arrangement that will enable robust grasping and manipulation on structures.

IV. Integrated Systems

The experimental layout of this project will consist of the RoboSat, a hovercraft, and a railing. Together, they will form a fundamental testing ground for evaluating the use of dexterous manipulators as the mechanism of choice on a small satellite for on-orbit servicing.

The RoboSat's geometric dimensions are consistent with those of the standardized 6U Cubesat which is approximately 10 cm x 20 cm x 30. This first generation prototype is equipped with two 3 degree of freedom arms and two dexterous grippers that will allow it to perform dynamic functions such as docking and "climbing" (shown in Figure 1). Each arm consists of two parallel revolute joints that will act as the shoulder and elbow joints, and a wrist joint. Each gripper consists of four fingers – each of which is made up of four links – that are powered through tendon routing and compliant joints. The tendons are thin cables that run through the length of each finger from fingertip to palm. Inside each palm, there are two motors and leadscrew assemblies. The reason benefit of this configuration is that each pair of opposable fingers will be independently actuated.

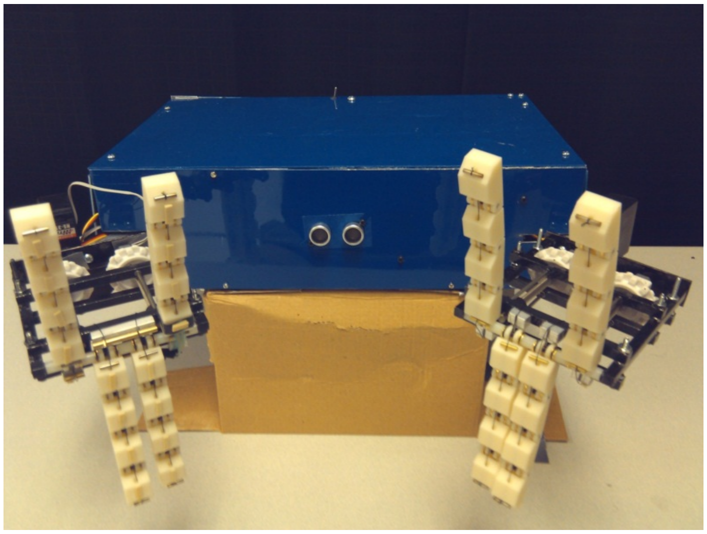


Figure 1. Shows the RoboSat with its grippers deployed and positioned in front of its ultrasonic sensor

The hovercraft consists of a square shaped board, a skirt, and an air pump. Once the air pump is activated, air will flow into the skirt below the board and escape through holes in the middle of the platform between the board and the ground. Essentially, this creates air film bearings that will allow the hovercraft to slide around on a near-frictionless plane.

Finally, the railing structure consists of a single, uniform PVC pipe, and will initially serve as a simple object for the RoboSat to interact with. Theoretically, the RoboSat will dock to and "climb" its way along this structure by holding onto the railing with one hand, and repositioning with the other.

V. Mobile Platform

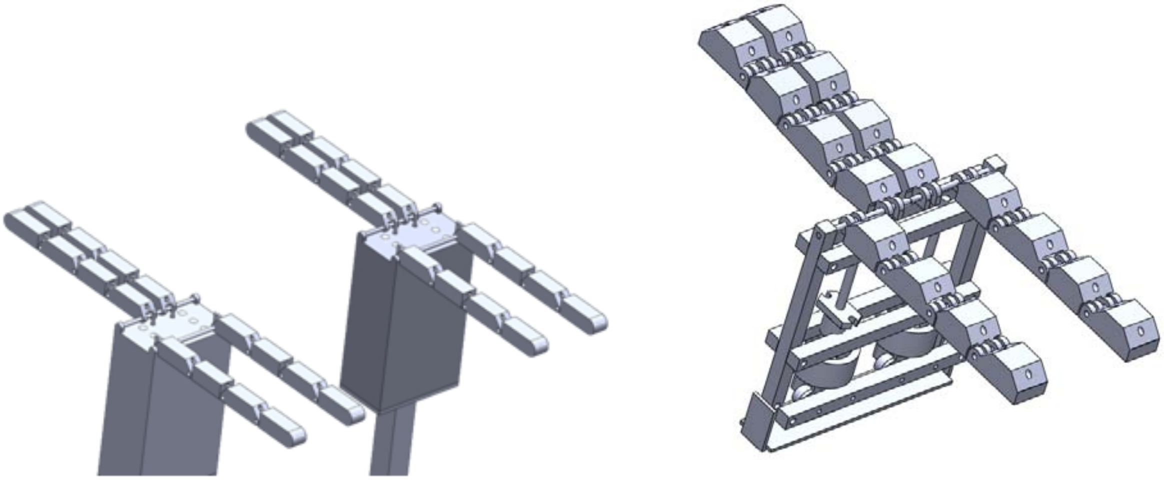
As previously mentioned, the full platform setup includes the RoboSat, a hovercraft, and a railing. The RoboSat's main features include two zygodactyl grippers, two 3 degree of freedom arms, an ultrasonic sensor, two four-bar linkages, and a 6U Cubesat frame.

The gripper design was chosen that is believed to possess the best qualities of simplicity, dexterity, grasping, and robustness. This design consists of a zygodactyl arrangement of four fingers as shown in Figure 2 (a). Biologically inspired (see Figure 2 (b), (c)), this arrangement will potentially allow for grasping on many different sizes and shapes of structures. For example, a parrot possesses these style grippers, and it has the ability to grasp onto larger objects such as branches as well as perform finer, more dexterous tasks such as holding a grape.

(a) (b) (c)

Figure 2. Shows (a) a zygodactyl grasper configuration and how it is used by (b) a parrot for grasping and climbing, and by (c) a chameleon climbing a branch.

The design of the manipulators went through a couple of iterations as shown in Figure 3. The reason for the redesign was largely due to manufacturing constraints, however, there were several ancillary benefits including a larger surface area from the enlarged fingers for grasping, and an open wrist for facilitating the assembling process.



(a) (b)

Figure 3. Shows the (a) first design, and (b) the second and final design of the grippers.

The grippers for the RoboSat consist of rigid links, revolute joints, torsions springs, tendon cables, force sensors, DC motors, and leadscrew assemblies. Figure 3 shows the assembly of a single gripper (note: torsion springs and tendon cables are missing).

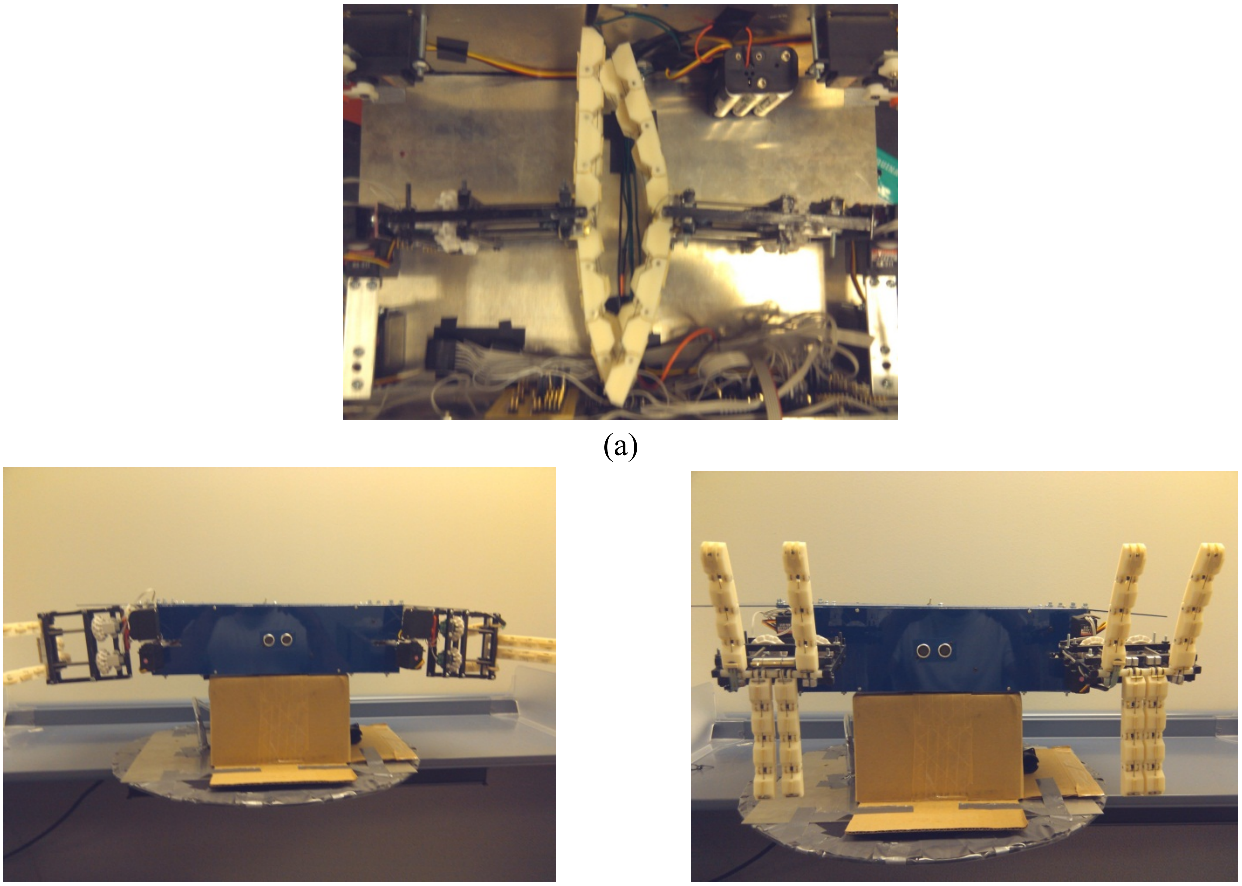
VI. Actuation

There are three main types of actuation for the Robosat: DC motors, servos, and an air pump. The motors and servos will allow for the movement of the RoboSat's panels, arms, and grippers while the air pumps will allow for the levitation of the hovercraft and robot.

The DC motors and servos will be used in two different ways. First, 8 servos will be used to purely rotate one rigid part of the robot with respect to another. For example, servos will rotate the side panels of the satellite as well as power the revolute joints in the arms of the manipulators (see Figure 4 for deployment). The four DC motors will provide torque to a leadscrew assembly which will transform rotary motion into linear motion. The reason for this motion conversion is to provide tension in the tendon cables that run through every finger in the gripper.

Since the force sensors will be mounted on the links of select fingers, the motion of these sensors will be directly associated with the motion of the fingers. The fingers are actuated by the use of DC motors coupled with torsion springs and tendon routing. These DC motors will be powering a leadscrew assembly which will pull on the cable tendons via the moving leadscrew nut (see Figure 5 (a)). Since the joints of the fingers are compliant through the use of torsion springs, the fingers will naturally curl as a human finger while being underactuated. Figure 5 (b) shows the gripper in a curled state with all four fingers being actuated at the same time and with the same driving force.

The air pump will supply a constant air flow and pressure to the hovercraft that will create air film bearings between the ground and the skirt. Since this flow of air will be directed towards the middle, bottom of the hovercraft, the entire bottom of the hovercraft will be levitated. During testing, the hovercraft was determined to be sensitive to ground gradients, and ground surface conditions. That is, steeper gradients induced sliding motions while rougher surfaces increased the overall friction experienced.



(a)

(b)

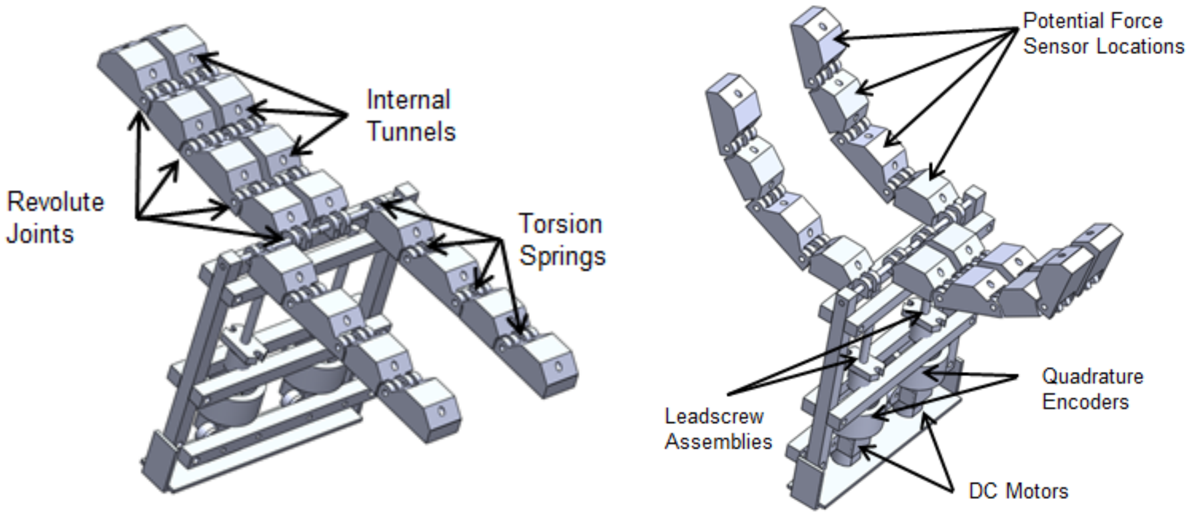
(c)

Figure 4. Shows the RoboSat with its grippers (a) stowed, (b) partially deployed, and (c) fully deployed.

VII. Sensors

The RoboSat will utilize 3 different kinds of sensors: force sensors, an ultrasonic sensor, and a specialized gripper with force sensing. Force sensors will be placed on the surfaces of the fingers in order to provide force feedback for controller purposes. The ultrasonic sensor will provide distance measurements of the object under interaction, are needed to help provide environmental awareness such as railing location.

The special sensor will consist of two specialized manipulators with force sensors strategically placed on each of the graspers (shown in Figure 5). The force sensors will provide direct information such as whether or not contact has been made with an object and how much force is being exerted on the object.



(a) (b)

Figure 5. Shows the gripper's internal tunnels for tendon routing, torsions spring locations, revolute joints, force sensor locations, quadrature encoders, and leadscrew assemblies in its (a) uncurled, and (b) curled states.

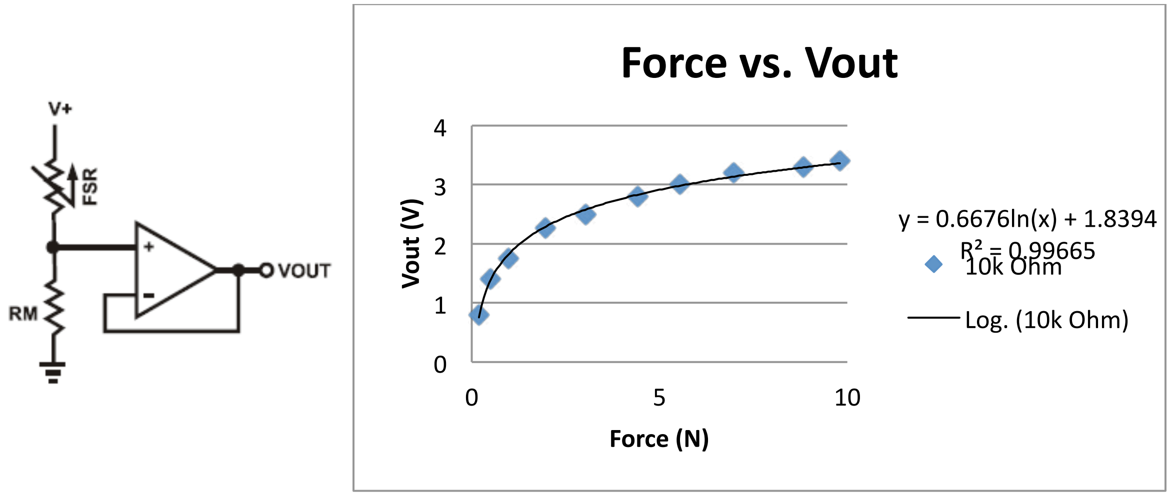
Shown in Figure 6, the output voltage of each force sensor will increase with increasing force applied to the force sensors. The output voltage is related to the input voltage and the RFSR and RM resistances according to the following equation.

$$Vout = \frac{V+}{1 + \frac{RFSR}{RM}}$$

In order to correctly configure the sensors, a force vs. output voltage plot will be generated from experimental data. Next, the data will be fit to a curve and an equation will be generated. In this case, the equation was determined to be the following.

$$y = 0.6676\ln(x) + 1.8394$$

This equation will be used during live sensing to provide the microcontroller with an accurate estimate of the applied force. This information will then be used to generate an appropriate output response based on the control algorithm.



(a) (b)

Figure 6. Shows (a) FSR voltage divider circuit, and (b) plots of Vout vs. Applied Force for varying RM resistance values.

VIII. Behaviors

There are several important behaviors that the RoboSat will have to exhibit in order to properly complete its task. First, the RoboSat's manipulators must be able to encapsulate the railing structure for a successful grasp for docking. Also, once docking has been established and "climbing" has commenced, the RoboSat must be intelligent enough to not relinquish its grasp with one hand unless the other hand has firmly established a grasp as well. The sense of a firm grasp will be established through force feedback in the hand. Finally, there may be gaps presented in the railing, and the RoboSat must be able to use force feedback and its proximity sensors to position and orient itself to bridge these gaps. The full control algorithm used for the RoboSat is shown below in Figure 7, and the code is shown in Appendices A and B.

Once a successful grasp has been made with both hands, the RoboSat must then move along the railing through a grasp-hold-release strategy. The RoboSat will initiate its "climbing" sequence when both hands have made successful grasps. The next step in this cycle will be to relinquish the grasp of one manipulator, and translate its position further along the rail. At this time, the RoboSat will attempt to grab the railing at the new location. Once a successful grab has been made, the other manipulator will release its hold and relocate adjacent to the other grasper's location. Here, it will attempt to make a new grasp on the railing. If successful, this process will continue in a loop for a specified number of steps before shutting down.

There are three main inputs during the climbing sequence – distance, force sensing, and servo hard stops. The RoboSat will determine whether or not it has to climb inward our climb outward based on the distance reading. Once climbing, the distance will be continuously measured until the desired location is met, and a force reading will be measured. If the force does not exist, the RoboSat will try to drive its hand into the direction of the railing. This will continue until a force reading is existent, or a mechanical hard stop is triggered. If zero force is read and a hard stop is triggered, then the RoboSat will try to make a grab anyway because chances are that its gripper is close enough to successfully grab the railing.

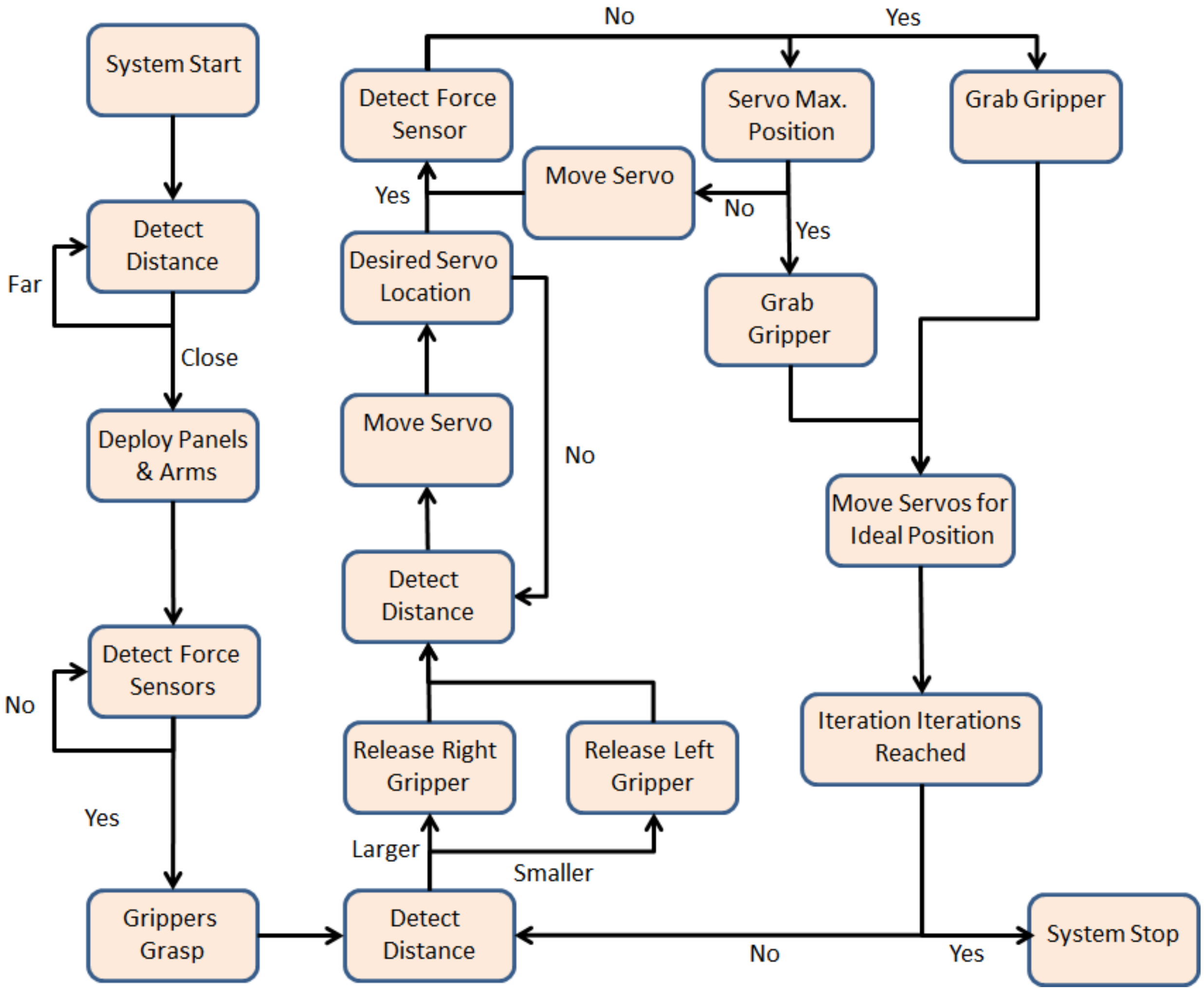


Figure 7. Details the full control algorithm used during the experimental demonstration.

A pitfall of this algorithm design is that if the RoboSat's hovercraft is situated on a floor with a slight gradient, then the whole platform will have the tendency to move downhill. The RoboSat does try to fight these sorts of effects with its algorithm, but due to mechanical hardstops and limited torque from the servos, steeper gradients will prevent a successful demonstration. This exact situation happened during the first run on Media Day, however, this factor was corrected for the second run which was a success.

IX. Experimental Layout and Results

The full experimental sequence consists of five main steps – systems initiation, deployment, docking, “climbing”, and shutdown (shown in Figure 8). First, the RoboSat and hovercraft will be activated during which time the RoboSat will begin levitating and monitoring its distance from the railing. Since no on-board thrusters exist, the RoboSat will be pushed forward towards the railing until the RoboSat detects that it is close enough and commences deployment. During deployment, the RoboSat will deploy two side panels, its arms, and its grippers, and place them in a “ready” position.

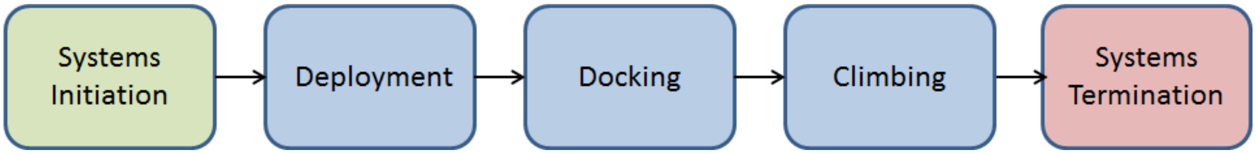


Figure 8. Depicts the theory of operation for the RoboSat during its experimental demonstration.

At this point in time, the RoboSat will be pushed into the railing and it will attempt to dock to the structure by grabbing the PVC pipe with its grippers (shown in Figure 9). Once the RoboSat is docked, it will begin climbing to its right on the structure for a predetermined number of iterations. Once completed, the RoboSat will enter shutdown by relinquishing its grasp from the railing.

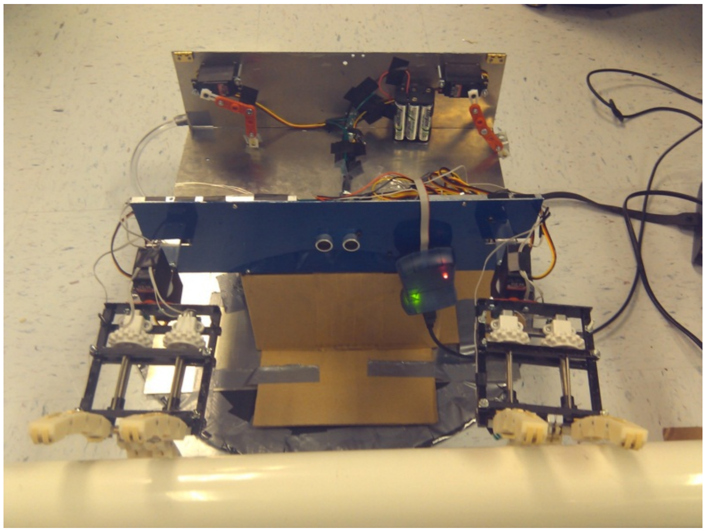


Figure 9. Shows the RoboSat approaching the railing for docking during a test sequence.

X. Conclusion

A pair of highly dexterous manipulators will be designed and mounted on a 6U Cubesat form factor. This robot will then be placed on a hovercraft such that it experiences minimal friction in a plane. While levitated, the RoboSat will attempt to dock to and "climb" around on the railing that is placed nearby. This experiment will deliver insight into the design and control of a small satellite capable of docking and performing on-orbit servicing in space.

XI. Documentation

[1] A. Dollar, A Robust Compliant Grasper via Shape Deposition Manufacturing, IEEE/ASME Mechatronics, Vol. 11, NO. 2, April 2006.

[2] M. Laffranchi. Antagonistic and Series Elastic Actuators: a Comparative Analysis of Energy Consumption, IEEE October 11-15, 2009.

[3] N.I. Glossas, Fuzzy Logic Grasp Control using Tactile Sensors, Mechatronics 11, 2001.

[4] A. Dollar, Towards Grasping in Unstructured Environments: Grasper Compliance and Configuration Optimization, Advanced Robotics, 2005.

[5] M. Diftler, Evolution of NASA/DARPA Robonaut Control System.

[6] S. Dalley, Design of Multifunctional Anthropomorphic Prosthetic Hand with Extrinsic Actuation, IEEE Mechatronics.

[7] A. Ellery, The Case for Robotic On-Orbit Servicing of Spacecraft: Spacecraft Reliability is a Myth, Acta Astronautica 63 (2008) 632 – 648.

[8] C. Cougnet, On-Orbit Servicing System of a GEO Satellite Fleet, ASTRA 2006 ESTEC, Noordwijk, The Netherlands, November 28-30, 2006.

[9] L. L. HOWELL, *Compliant Mechanisms*, 2001. John Wiley & Sons, Inc., New York ISBN 0-471-38478-X

XIII. Appendices

Appendix A

```
void SonarInit(void)
{
	PORTF_DIR = 0x00; //set PORTF7:0 to input
	PORTF_PIN0CTRL=0x00; //set PORTF PIN0 both edges trigger
	EVSYS_CH2MUX=0x78; //set PORTF0 as multiplexer of event channel 2
	TCF0_CTRLD = 0xCA; //set TCF0 to pw
	//event channel 2
	TCF0_CTRLB = 0x10; //Enable TCF0 OC A. Set to Normal
	TCF0_CTRLA = 0x05; //set TCF0_CLK to CLK/64
}

float distance(void)
{
	float value;
	PORTF_DIR = 0x01; //Set Port J, pin 0 to output
	PORTF_OUT = 1; //Set Port J, pin 0 high
	delay_us(5); //Delay 12? microseconds
	PORTF_OUT = 0; //Set Port J, pin 0 low

	//PORTF_DIR = 0x00; //Set Port J to input

	PORTF_DIR = 0x00; //set PORTF7:0 to input
	delay_us(750); //time to emit sound
	PORTF_PIN0CTRL=0x00; //set PORTF PIN0 both edges trigger
	EVSYS_CH2MUX=0x78; //set PORTF0 as multiplexer of event channel 2
	TCF0_CTRLD = 0xCA; //set TCF0 to pw
	//event channel 2
	TCF0_CTRLB = 0x10; //Enable TCF0 OC A. Set to Normal
	TCF0_CTRLA = 0x05; //set TCF0_CLK to CLK/64

	value=((TCF0_CCA)/147)*2;
	return value;
}
```

Appendix B

```
#include <avr/io.h>
#include "PVR.h"

void main(void)
{
xmegaInit(); //setup XMega
delayInit(); //setup delay functions
ServoCInit(); //setup PORTC Servos
ServoDInit(); //setup PORTD Servos
ADCAInit(); //setup PORTA analong readings

int i = 0, j = 0, D0 = 0, D1 = 0, D3 = 0, D4 = 0, sensor1, sensor2, cycle = 0;

// Deployment & Ready Position Sequence /////////////////

distance();
delay_ms(100);
while (cycle <5)
{
distance();
delay_ms(100);
cycle = cycle+1;
}

while (TCF0_CCA > 800) //Wait for distance to be small enough
{
distance();
delay_ms(100);
}

while(i < 130) //Deployment of solar panels
{
ServoC4(-25+i);
ServoC5(5-i);
delay_ms(100);
i = i + 1;
}
```

```
ServoC1(50); //Provide power to these channels
ServoC0(50);
ServoC2(50);

i=0;

ServoD0(112); //Initial arm joint angles
ServoD1(55);
ServoD2(98);
ServoD3(-110);
ServoD4(-55);
ServoD5(95);
delay_ms(1000);

while(i <= 55) //Begin arm deployment sequence
{

ServoD0(112-i);
ServoD1(55+i);
ServoD3(-110+i);
ServoD4(-55-i);
delay_ms(100);
i = i + 1;
}

while (i <= 150)
{
ServoD0(112-i);
ServoD3(-110+i);
delay_ms(100);
i=i+1;
}

i = 0;

while (i <= 150)
{
ServoD1(107-i);
ServoD4(-105+i);
```

```
delay_ms(100);
i=i+1;
}

i = 0;

while (i <= 50)
{
ServoD0(-38-i);
ServoD2(98-i);
ServoD3(40+i);
ServoD5(95-i);
delay_ms(100);
i=i+1;
}

while (i <= 100)
{
ServoD2(98-i);
ServoD5(95-i);
delay_ms(100);
i=i+1;
}

// Collision Sensing & Docking Sequence /////////////////

sensor1 = ADCA0();
sensor2 = ADCA1();
while((sensor1 < 1000) && (sensor2 < 1500)) //Wait for force sensors to trigger
{
sensor1 = ADCA0();
sensor2 = ADCA1();
}

PORTJ_DIR = 0xFF; //Grasp with both hands
PORTH_DIR = 0xFF;
PORTJ_OUT = 0x0E;
PORTH_OUT = 0x70;

ServoC0(50);
```

```
ServoC1(50);
ServoC2(50);
ServoC3(50);
delay_ms(1600);
ServoC2(0);
ServoC3(0);
delay_ms(100);
ServoC0(0);
ServoC1(0);
delay_ms(2000);

// Climbing Sequence ////////////////////////////////////

while( j < 4)
{
PORTJ_OUT = 0x15; //Release Left Grasp
ServoC0(50);
ServoC1(50);
delay_ms(1900);
ServoC0(0);
ServoC1(0);

i = 0;
D0 = -88;
D1 = -43;
D3 = 90;
D4 = 45;
distance();
delay_ms(100);
while ((TCF0_CCA > 300) && (i <45)) //Climb Outwards: wait for distance and position to be right
{
ServoD0(D0+i);
ServoD1(D1-i);
ServoD3(D3-i);
ServoD4(D4+i);
distance();
delay_ms(100);
i = i+1;
}
```

```
D0 = D0+i;
D1 = D1-i;
D3 = D3-i;
D4 = D4+i;

i = 0;
sensor1 = ADCA0();
while ((sensor1 < 1000) && (i<15)) //Move left hand closer if not touching already
{
sensor1 = ADCA0();
ServoD3(D3+i);
delay_ms(100);
i = i+1;
}
D3 = D3+i;

PORTJ_OUT = 0x0E; //Grasp left hand
ServoC0(50);
ServoC1(50);
delay_ms(1900);
ServoC0(0);
ServoC1(0);

i = 0;
while (D3 > 45) //Readjust to desired orientation
{
ServoD3(D3-i);
delay_ms(100);
i = i+1;
D3 = D3-i;
}

PORTH_OUT = 0xA8; //Release right hand
ServoC2(50);
ServoC3(50);
delay_ms(1600);
ServoC2(0);
ServoC3(0);

i = 0;
```

```
distance();
delay_ms(100);
while ((TCF0_CCA < 1000) && (i<45)) //Climb inwards: wait for distance and position to be right
{
ServoD0(D0-i);
ServoD1(D1+i);
ServoD3(D3+i);
ServoD4(D4-i);
distance();
delay_ms(100);
i = i+1;
}
D0 = D0-i;
D1 = D1+i;
D3 = D3+i;
D4 = D4-i;

i = 0;
sensor2 = ADCA1();
while ((sensor2 < 1500) && (i<25)) //Move right hand closer if not touching already
{
sensor2 = ADCA1();
ServoD1(D1-i);
delay_ms(100);
i = i+1;
}
D1 = D1-i;

PORTH_OUT = 0x70; //Grasp right hand
ServoC2(50);
ServoC3(50);
delay_ms(1600);
ServoC2(0);
ServoC3(0);

i = 0;
while (D1 < -43) //Readjust to desired orientation
{
ServoD1(D1+i);
```

```
delay_ms(100);
i = i+1;
D1 = D1+i;
}

j = j+1;

}

// Release Grasp ////////////////////////////////////////

PORTJ_OUT = 0x15;
PORTH_OUT = 0xA8;
ServoC0(50);
ServoC1(50);
ServoC2(50);
ServoC3(50);
delay_ms(1600);
ServoC2(0);
ServoC3(0);
delay_ms(300);
ServoC0(0);
ServoC1(0);
}
```