CPS342: Embedded Linux

Report 1

Servo Motor with web access

Group Name:

Group Members	Department	Major Contribution
Mitchell Wagner	CE	3D Design/Servo configuration/Building of Arm
Paul Boston	CE	Servo configuration/Building of Arm
Jose, Da Costa Filho	CS	Website development and camera control
Charles Palsa	CS	Website development, camera control and GUI
Shamiul Islam	CS	? website dev?

Course Professor: Easwaran Chirakkal

04/06/2015

Table of Content

1. Introduction:

The assignment for this project was to create a web page that has a live feed from a camera that can be controlled to move in any direction. Our group took this to mean that a servo motor would be connected to the pi and a person could control this motor and see where it is pointed via the web page. Our project is a functioning camera arm that has 3 servo motors to control the camera direction, this enables the user to see in every direction possible and turn on any axis. These servos then communicate with the pi and are able to be controlled from the web page with a built in GUI to control the location of the servos and see the live feed of the camera.

1.1 Project Description:

This project is to design a system that will be able to be accessed through the web and the user will get live feed and be able to control the camera 360 degrees in every direction. Our system that we created is a camera arm with 3 different servo motors, these motors then connect to the pi. The pi is connected to the internet and is hosting our website that displays our GUI that will control everything.

1.2 Project Goals:

Our project goals are to have a working camera arm that will operate from the internet using a GUI to communicate with the servo motors and the camera. The servo goals are to design a rig to hold 3 motors to complete the arm. Second create a way to connect all the servos to the pi. Third, have the servos accurately move to the degrees desired and not get the wires tangled. Our goals for the camera is to hopefully display the camera feed in a high FPS (Frames Per Second) so that viewing the feed looks normal and would be able to take snapshots of a live feed and store them. We want to also make sure there would be a low delay time when accessing anything on the website. The website goals are that it will be mobile friendly and that the camera feed will display natively as well as all the controls for the servos will relay back to the servos and actually work.

2. Implementation:

In order to implement our plans to complete this project we decided to break it down into several different parts: website, camera, servo's and stand.

2.1 WebSite:

The web site was going to be written in different programing languages to be able to be able to be used on multiple portable devices.

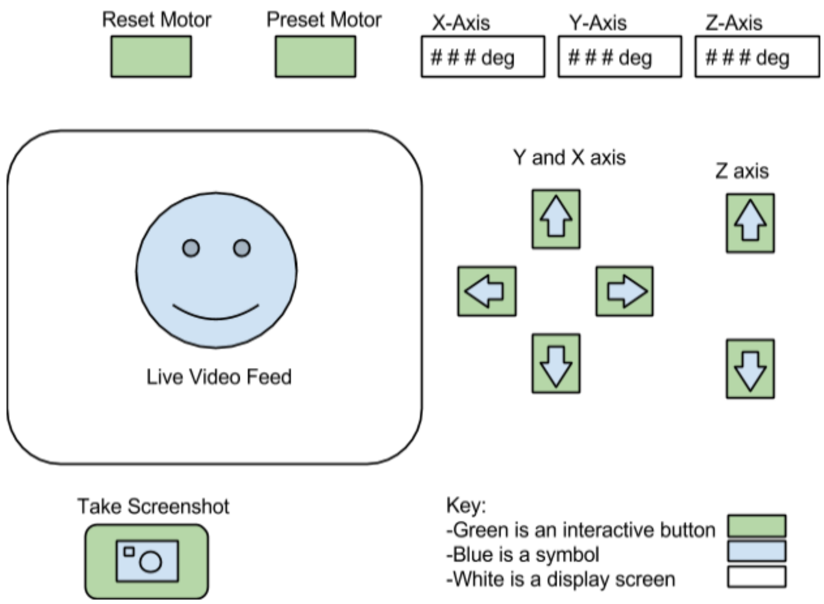

Figure 1: Preliminary Website Design

Figure 1 helps to show how the design for the end result for the website will look. This has the ability to use buttons to control the motors as well as type the location desired and display where you are. There is also a screenshot button and reset the motors to the origin location (0, 0, 0).

2.2 Camera:

The camera was going to be the raspberry pi camera so that the code should be able to since up much more smoothly. We would also need a raspberry pi camera extension cable to actually make the camera reach to raspberry pi from the camera rig. This camera should sit at the end of the rig on a platform.

2.3 Servo:

The servo motors were determine to be a 5V high torque Continuous motion 360deg motors and that it was decided that 3 motors would be needed to get full rotation of the camera. This would allow full rotation in any direction across all three axis. These motors have a signal pin that would need to be controlled by a pwm port. We figured out that since the raspberry pi only has 1 pwm port we would need to get a pwm servo controller board to interface with the pi to control all three. Below in Figure 2 our design for the camera movement and control.

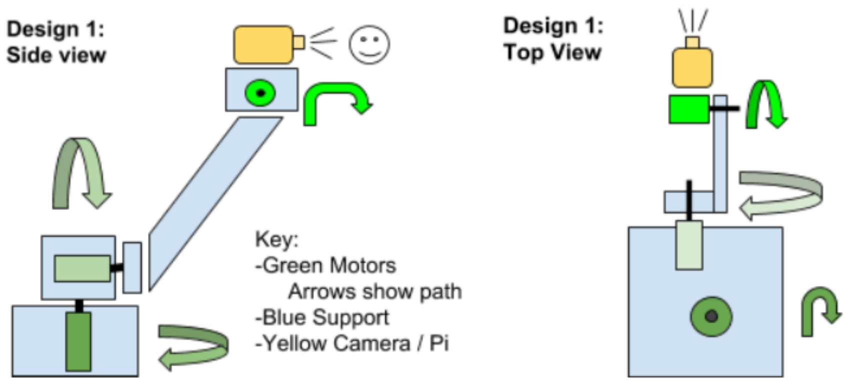

Figure 2: Motor rotation

Figure 2 helps to show how each motor will be operating the cameras movement. This initial diagram was created to help realize the amount of motors needed to get a full 360 degree rotation. As well as where they could be placed.

2.4 Stand:

The stand was decided to be printing form the 3D modeling place. The material used is a hard plastic which is durable and strong enough to support all the devices that it will interact with. We figured that we needed to create brackets that each motor can mount to and can also hold another motor. We also figured that we would need to have a base for all the components to sit on. In figures 3 and 4 are the preliminary drawings of our 3D designed pieces.

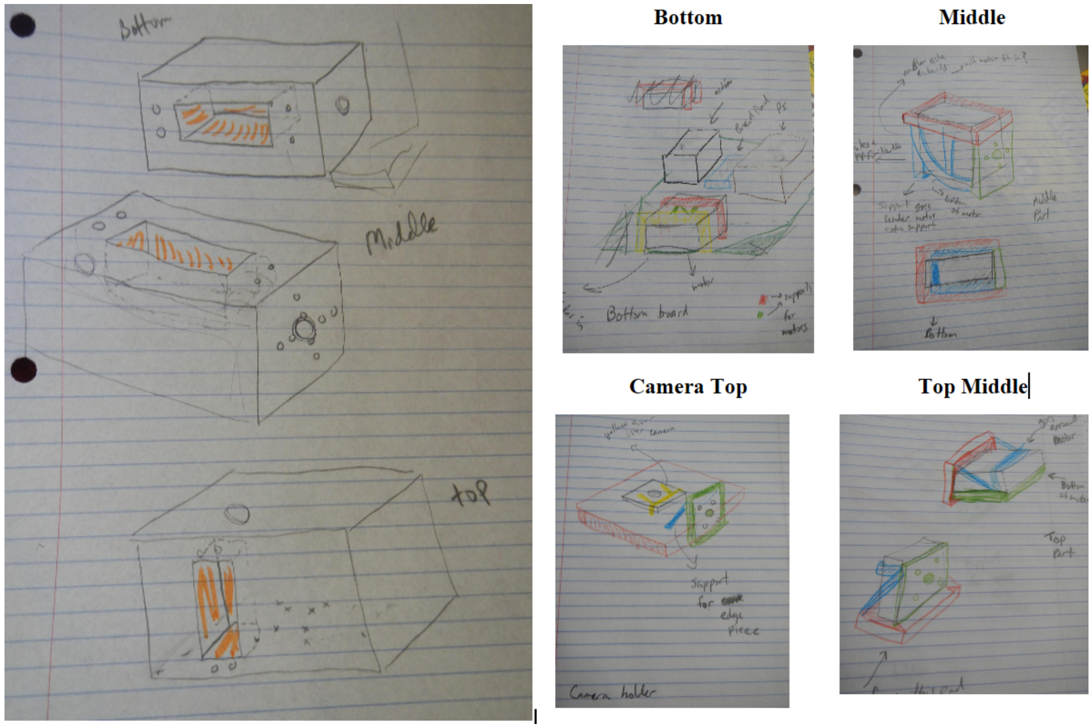

Figure 3&4 Rough drawings for a prototype of the stand

Figure 3 was a design that incorporated just printing out box holder for each of the motors although this idea should work, we were advised to think outside the box. This way to make a product that one wouldn't take up as much material as well as look more appealing. Figure 4 shows a design for a bracket look. This was broken up into all the components to help be more appealing and use less materials, but to check the functionality the design went through a prototype phase before actual printing.

3. Components:

This area of the project consisted of what areas were broken up into their different parts and who did these parts or more or less focused in these parts. This would go into how each area of the project would be established and what has been done in these areas in order to get closer to the final end design.

3.1 Website:

The website was preliminary done in bootstrap, CSS and HTML a diagram can be seen at Figure 1. The website GUI and front end will be designed using the Bootstrap framework, and motor/camera controls will be interfaced using python scripts. The screenshots taken by the client will be saved in the server side. The information of the pictures will be save on a sqlite3 database. Php scripts is used to save the images and manipulate them on the server. The images will be loaded to the gallery section in the website using Ajax.

3.2 Camera:

When the implementation for the camera was started it was found to give live stream but very slow rate. This was then where it was decided to start with just having a single image show up and as we get the understanding of increasing the speed we could be able to make the camera run at a faster time this way the amount of frames shown would be more accurate. The video feed will be streamed over http via a motion-jpeg server. This works natively in Chrome, Safari, and Firefox. Internet Explorer does not natively support this feature.

3.3 Servo:

There are several key components for the servo motors. First was the assembling of the motors with the stand. This required the stand to be constructed and measured properly. Then there was soldering of the PWM controller board, this board was important because the raspberry pi only has one PWM port and we needed at minimum 3 ports to control each motor. Then the main component was all the things that required the coding for the motors to work in ways that the user would desire.

The motors were started by assembling the entire camera arm but focusing on one motor at a time. This was difficult at first to understand how to get the motor to move just a degree motion and not freely. The problem that would occur would be that the motors continued to move freely and would be the restricted by the wires getting all tangled up and resist future movement, which was discovered when first trying the motors out. In order for the motor to move by degree this required the understanding that the amount of time that was giving for a command to be on would determine the amount of degree that it would move. There was other ways that were tried by changing the pulse length, the best that this method got was to get only 15 degrees of movement at a time. This was discovered when the entire motor would travel 360 degrees in 24 different movements. Then we discovered that by removing the time implementing and using a for loop to represent the amount of times it would cycle through gave a much more accurate degrees of rotation and would then give the user a better control over the entire camera arm.

test.py

```
if(input<0):
 y=1
 input=abs(input)
else:
 y=0
if(input<=90):
 input= int (round(input*.70))
if((input>90) and (input<180)):
 input= int (round(input*.80))
if((input>=180) and (input<=200)):
 input= int (round(input*.88))
if((input>200) and (input<=275)):
 input= int (round(input*.97))
else:
 input= int (round(input*1.02))

pwm.setPWMFreq(60)
x=0

if(y==0):
 for x in range(0,input):
  pwm.setPWM(0, 0, servoMin)
  pwm.setPWM(0,0,0)

if(y==1):
 x=0
 for x in range(0, input):
  pwm.setPWM(0,servoMax,0)
  pwm.setPWM(0,0,0)
```

Figure 5: Motor Code

Once the first motor was understood then each motor was added one at a time checking that it would also move the correct distances.

Another construction was the ability to save the current location of the motors to a file. This was important to be able to have the motor know the location it was sent to in order to know how much farther it would be able to go before tangling the wires.

Areas that still need to be worked on is getting the more accurate way of each degree. Checking that the motors will stay between -360 and 360 degrees. Also having several numbers fixed for a certain purpose like go back to 0, 0, 0.

3.4 Stand:

We needed to create a stand that would not only stand on its own but be able to move in every direction via the motors. This was a difficult challenge, originally we had boxes that the servos would fit into to use in the arm. We soon figured out that boxes would not work because we would not be able to fit all the pieces together if we went that route. We decided to use a different approach by using brackets instead, below in figure 6 is a prototype of our brackets before we designed them.

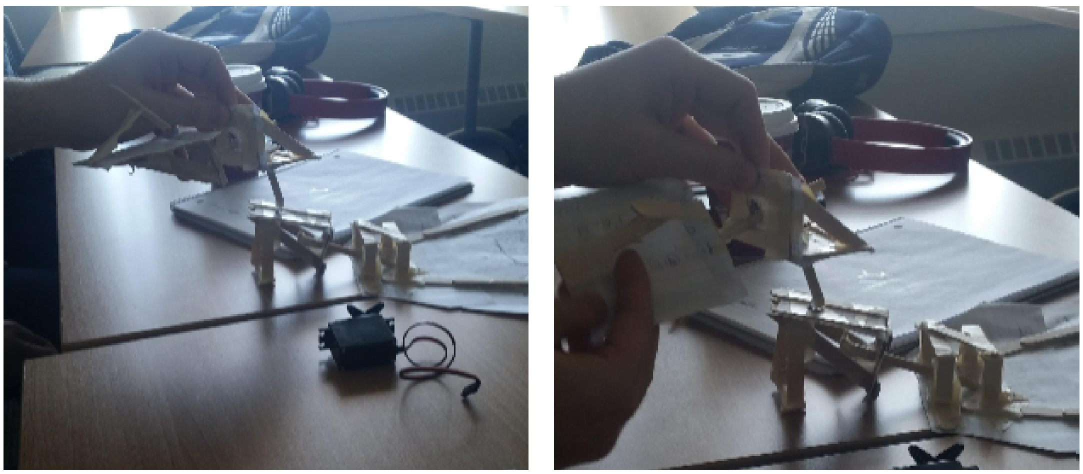

Figure 6: Prototype of Stand

Then we decided to model the parts in AutoCad Inventor since we liked the design. First we made the brackets to fit to the motors, these were the junctions between motors 1&2 and 2&3. Then we designed the mount for motor 1 which would eventually be mounted to the bottom plate which wasn't even made yet. The design for the mount for motor 1 was a simple box holder so that the motor could sit upright with a hole for the wire out the side. Motor 3 has a mount on the side which is a large flat square that our camera will be mounted to, this just ensures that the area that the camera attaches to has a wide amount of room. Finally the base was designed to be a large flat slab that has indents for all the other pieces to fit in properly. These pieces were successfully printed on the first try and installed without a problem. Below in Figure 7 is our designs for these pieces in Inventor.

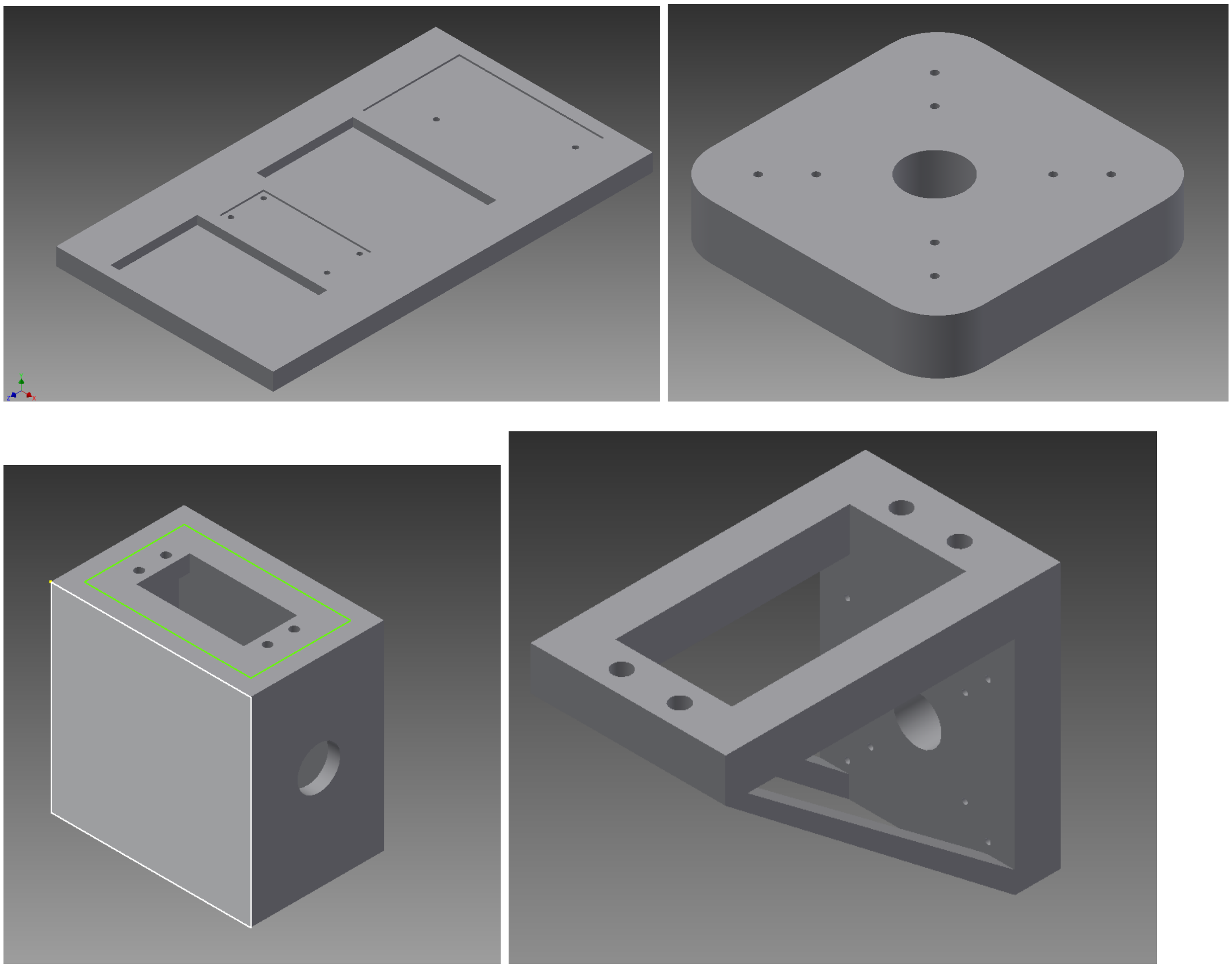

Figure 7 3D printed parts from inventor

Each of these parts were just printed once except for the bottom right was printed twice. There could've been another design but it was realized that the motions were still based on the very similar design.

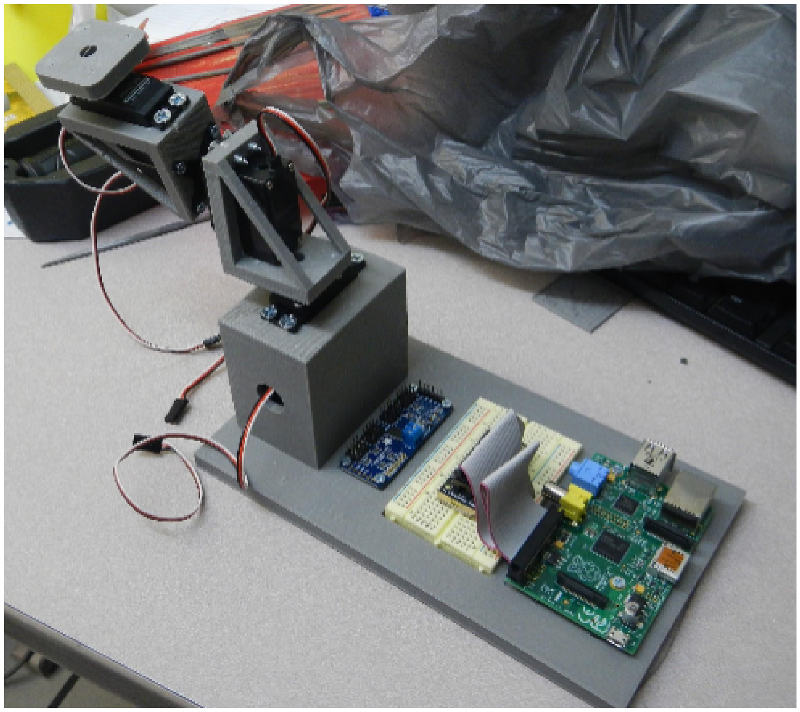

Figure 8: Complete Stand with Pi, motor and Servo attached

The assembling of the 3D modeling was constructed and it can be observed in Figure 8. This shows all the parts form Figure 7 printed and assembled.

4. Aspects:

This area of the report helps to dictate what areas held difficulty and what areas were able to be done with relative ease. We had relative ease building the arm but figuring out what type of supports to design was tough as well as getting the servos to work correctly. Making the servos calculate distance in degrees was a hard challenge and we are still trying to perfect it. The website was a good problem to work out, the camera freed on the website was a challenge because we were having a java applet problem with the website, our frame rate is not as fast as it could be as well.

5. Future Goals:

The future goals for this project would be to start tying in all the components and get the GUI all set up. Also the motors still need to be more accurate if going to the correct degrees inputted as well as returning accurately returning to the origin.

A future goal for the website would be if the project is that far ahead we would add instead of sliders incrementing the degrees of the motors a circular button would be used.

Additionally, the motion jpeg server supports motion-tracking capability. A future goal would allow the camera head to track movement.

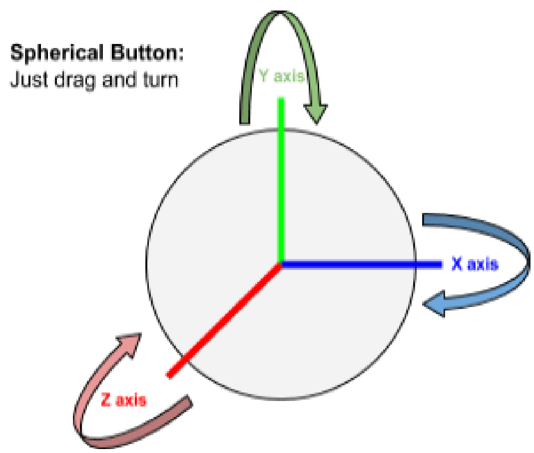

Figure 8: Future button for website

6. Conclusion

When projects are being processed it is often satisfying when they are coming smoothly along. There is often many places of difficult frustration but in the end when things work the result is satisfying. This doesn't mean that the creation for this project is over but being able to move along with results that show progression is always good.

7. Appendix: