# Introduction

Ada is an open source project initiated by the University of New Haven Cyber Forensics Research and Education Group (UNHcFREG). Ada was designed to provide users with valuable cyber security tips and information in a way that is both interactive and entertaining. Ada is capable of reading tweets, checking if email accounts have been hacked, reading RSS feeds, providing tips, prompting quizzes, telling jokes, and showing articles – all related to cyber security. It is our hope that people will take advantage of the open source nature of this project to build Ada for use in their own schools and universities and contribute to the project to make Ada better.

Ada has three major components – Mechanical, Electrical, and Software. The mechanical components were designed using AutoDesk Inventor 2014, several of which were 3D printed using a MakerBot Replicator 5th Generation and a Stratasys Dimension 3D printer. The electrical design was done by hand with good ole’ pencil and paper. The software components were made using a few different programs. Two separate Android applications were designed using Android Studio, these are SecuroBotMaster and SecuroBotSlave. The master app acts as Ada’s software eyes. The slave app is responsible for fetching and running all of the content as well as interfacing with the hardware to move the head and handle the communication and interaction with the user. The on board Arduino is running a program that was written using the Arduino IDE. This program is used to interface with the servo which controls the head movement.

The following provides a basic instructional guide for how you can build your very own Ada. All software, CAD, bill of materials, schematics, and other relevant documentation is available for download at the top/bottom of this page and on GitHub. We hope you decide to build your own Ada and look forward to Ada’s future!

# Procedure

The following outlines the procedure for constructing Ada. It is separated into three main sections – Mechanical, Electrical, and Software. Ada is capable of running on as little hardware as just a tablet. Therefore, it is up to you as the builder to decide what other features you would like to include. Our version of Ada is stationary but animatronic. Her head moves but the body stays still. She also has an onboard speaker system with a light that blinks to the sound of her voice. We have integrated a 5000mAh battery into our model which is enough to keep the system running as well as charge the tablets for several hours. Feel free to add or subtract components as you wish! The following procedure assumes you have obtained all of the items outlined in the bill of materials and have access to a basic tool set as well as a few more advanced tools such as a soldering iron, Bridgeport machining center, horizontal ban saw, and a medium to large sized 3D printer.

## Mechanical

As mentioned earlier, all of the CAD was designed in AutoDesk Inventor 2014 and is included in the download. We have included all of the .ipt files as well as the .stl files for those parts that are 3D printed. The fully assembled Ada can be seen in Fig. 1 below.

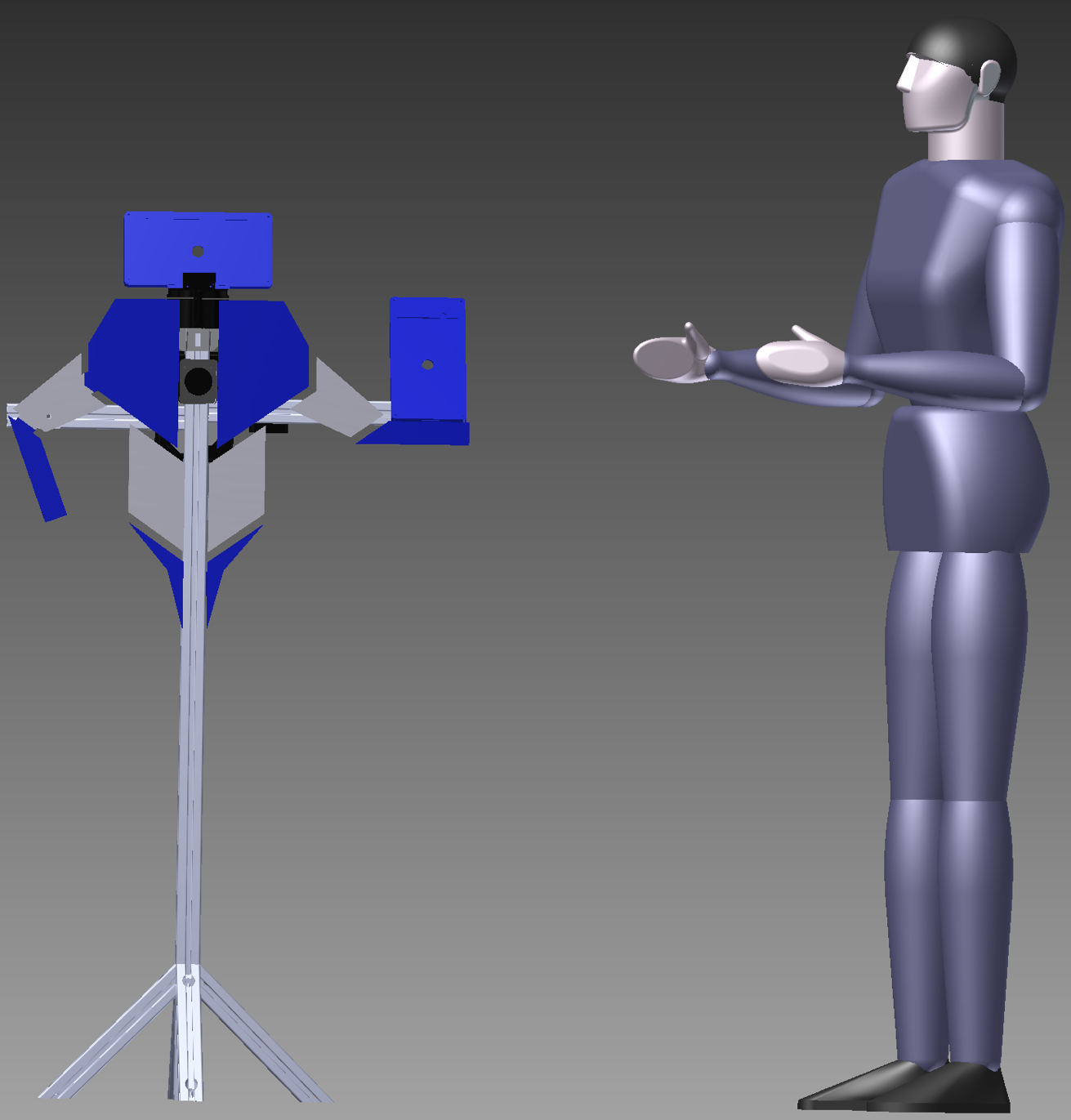
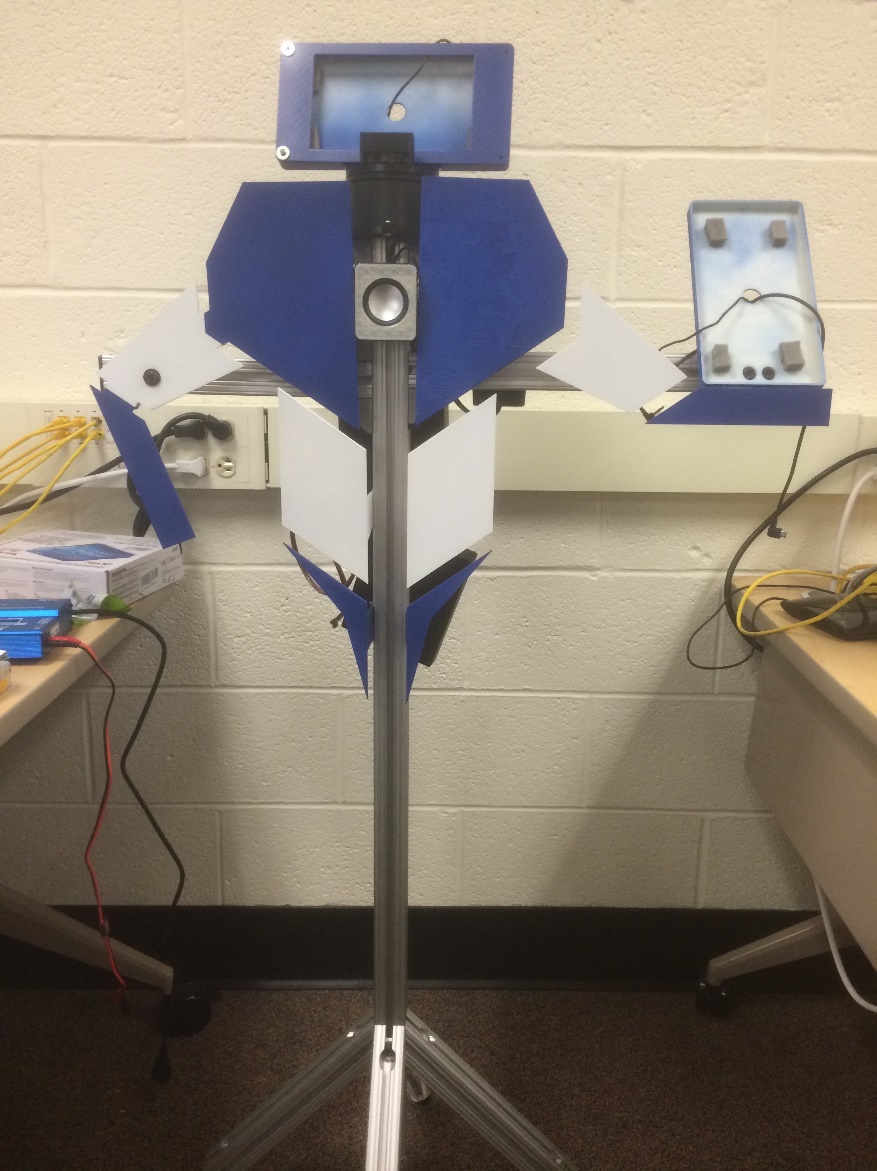
 

Figure 1 – Ada fully assembled.

The first step in building Ada is to construct the frame. The frame is the rigid aluminum structure that holds all of the other components. The frame components are located in a folder called “Milled Extrusion Parts”. This frame was constructed out of 1.5in. 80/20 aluminum extrusion. The extrusion was cut to rough length with a horizontal ban saw and was then milled to exact length using a Bridgeport machining center. It is possible to skip this last step but it will leave the edges in rough shape and the parts may not line up perfectly square. Once all the pieces are cut, they are assembled using the 80/20 T-slot fasteners and the angle braces. The fully assembled frame can be seen below in Fig. 2.

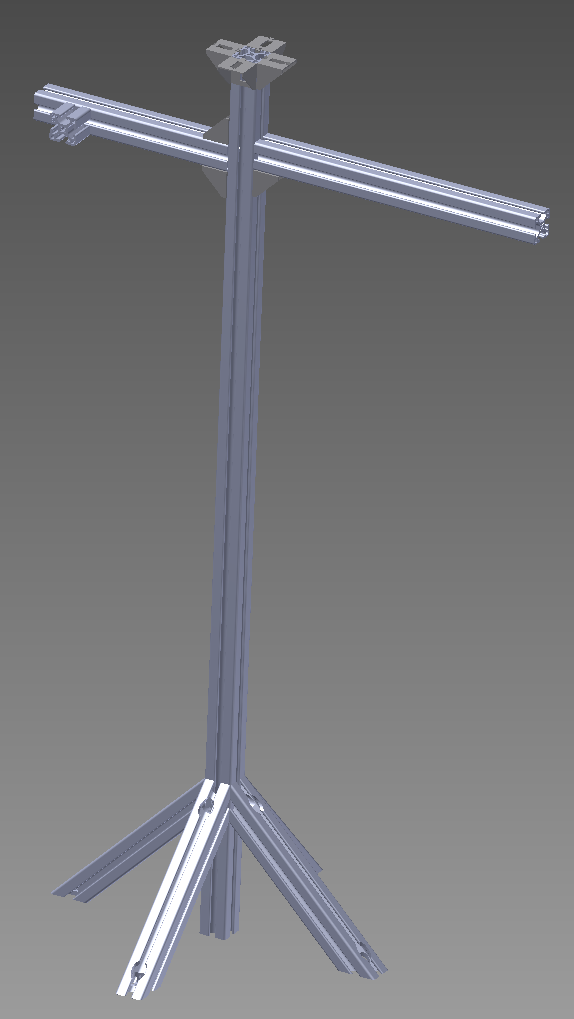


Figure 2 - Ada’s frame constructed of 80/20 extrusion.

The next step was to print all of the pieces in the “3D Printed Parts” folder. Two different 3D printers were used to print these parts – A MakerBot Replicator 5th Generation and a Stratasys Dimension 3D printer. The Dimension was the preferred printer for most of these parts, however both have their advantages and disadvantages in printing these parts. The Dimension was able to print the bigger parts much better. This is because of the climate controlled environment, not available in the MakerBot. The Dimension was also much better equipped to print more complex parts due to the nature of its build process. The Dimension actually builds a dissolvable support layer which holds parts together that would otherwise not be able to support themselves on their own during the build process. These supports do however come at a cost. The cost being time and money. It is much more expensive to run this printer and cost more because of the material. The MakerBot was used to print small parts much faster and cheaper than the Dimension. Parts such as the speaker cover, stand offs, rotating base mount, and other small parts were all printed using the MakerBot. We found that the MakerBot could not print the larger pieces such as the chest plates, because the material would peel off the build surface while being printed. Parts that actually managed to stick to the surface would come out warped. This is most likely due to the lack of a climate controlled build environment. For our version of Ada, we decided to do most of the printing with a single color of filament (white) however a few parts were printed in other colors. The IR cover, Rotating Base Mount, and Master Tablet and IR Mount were all printed in black to match the rotating base. The Speaker Cover was printed using a translucent filament which allowed the light from the LEDs to be seen. The body parts were then spray painted in alternating blue and white, representing our school’s colors. It is recommended that you actually print the parts using the correct color filament, as it looks nicer in the end and you will save time by skipping the spray paint step. At this point, most of the body parts can be fastened to the frame, again using the T-slot fasteners. Ada’s Right chest, bicep, and forearm are fastened using #10-32 screws and nuts. At this point, Ada should start taking a recognizable shape as seen in Fig. 3.

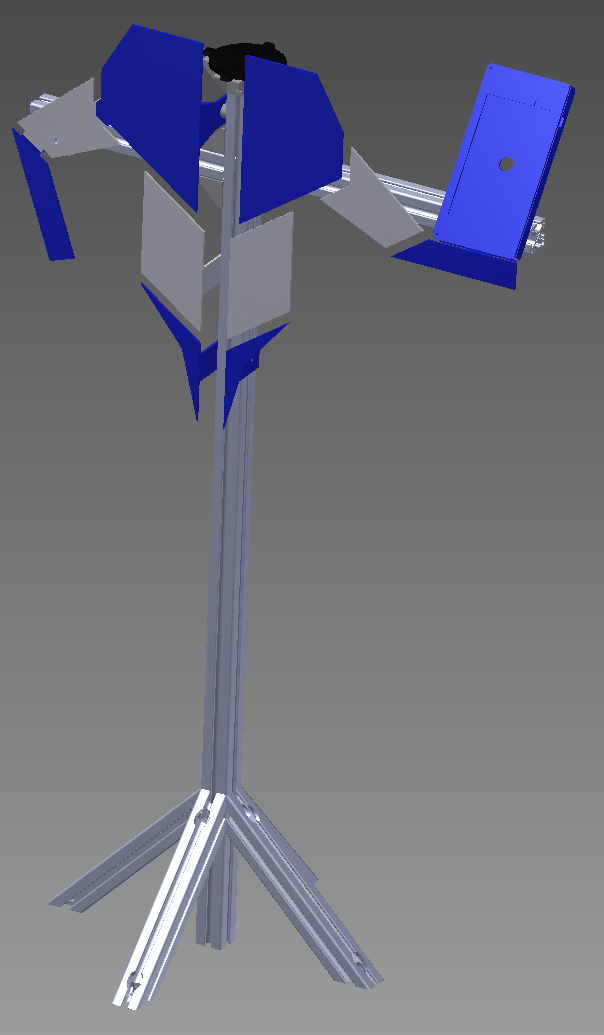


Figure 3 – Ada with most of her body parts mounted using T-slot fasteners.

The wheels are fastened to the legs using 3/8-16 bolts and 3/8 lock washers as seen in Fig. 4 below.

Figure 4 – Swivel wheels are mounted to Ada’s legs for easy transportation.

Next, the servo is fastened to the rotating base using the included hardware. Because of the way we have built our Master tablet mount, it is necessary to put washers under the servo before fastening it to the underside of the rotating base, seen below in Fig. 5.



Figure 5 – Washers are placed under the servo before mounting.

With the rotating base assembled, the IR sensor can be screwed into place on the Master Tablet and IR Mount, along with the IR cover. The mounted IR sensor assembly is show in Fig. 6.



Figure 6 – The IR sensor is fastened to the mount.

After this, the speakers are mounted. 5/16 in. holes are drilled in each of the speaker heads in order to mount them to the frame using the T-slot fasteners (Fig. 7). Small pieces of foam can be hot glued to the tablet enclosures to keep the tablets firmly pressed against the cover and not shaking around (Fig. 8).

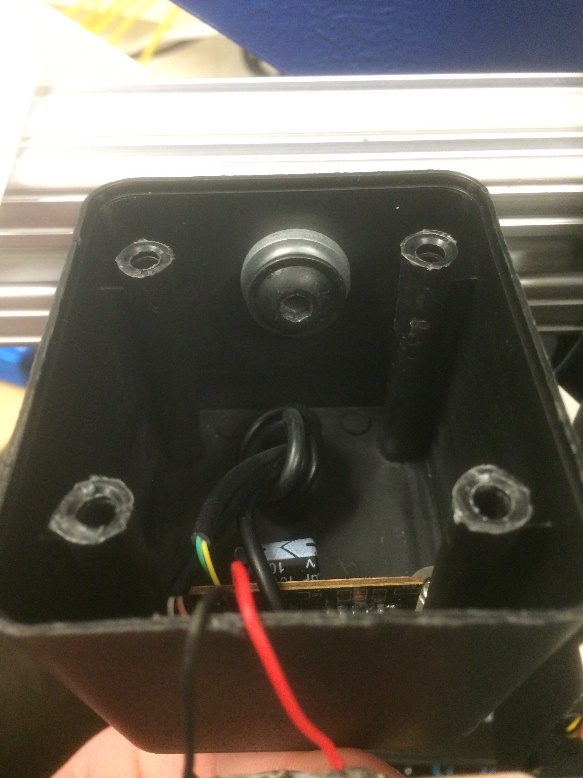
 

Figure 7 – Speakers mounted to frame. Figure 8 – Foam is added to tablet enclosures.

At this point, the master tablet enclosure can be fastened to the Master Tablet and IR Mount using #6 screws. The assembly now looks like Fig. 9.

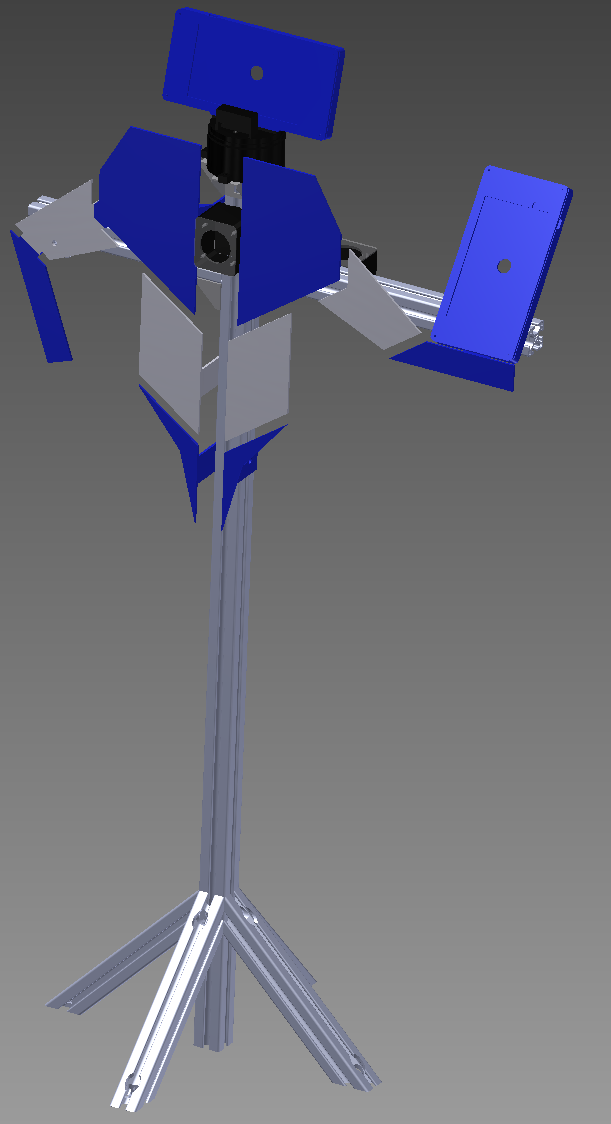


Figure 9 – All body parts attached, just missing the electrical enclosure.

The final stage of the mechanical assembly is to mount the electrical enclosure. Before this can be mounted however, a few holes need to be drilled. First, four holes are drilled in the top, right, and bottom sides of the enclosure. These holes are for 3 wire strain reliefs and the main power toggle switch. The next set of holes to be drilled are for the circuit board. Holes for the main board can be drilled through the back of the enclosure and holes for mounting the Arduino board can be drilled in the cover. It is our experience that this enclosure is pretty tight with everything in it, so you may want to invest in a slightly larger enclosure, especially if you plan on expanding more on the electronics. Finally, two 5/16 in. holes can be drilled in the back of the enclosure on vertical center for mounting to the frame. With all of this done, the enclosure can now be mounted to the frame using T-slot fasteners. Ada should now look like Fig. 10 from behind.

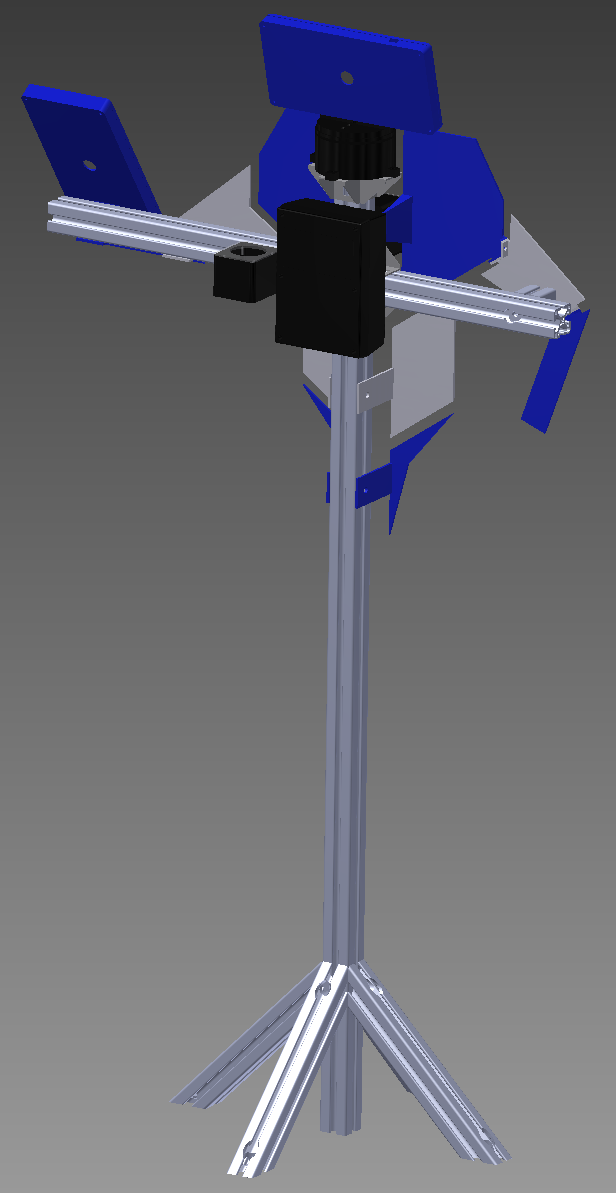


Figure 10 – Ada rear view. All components mounted.

## Electrical

With the mechanical components all in place, it is now time to being the electronic assembly. The schematic in Fig. 11 was created to assist in the wiring process.

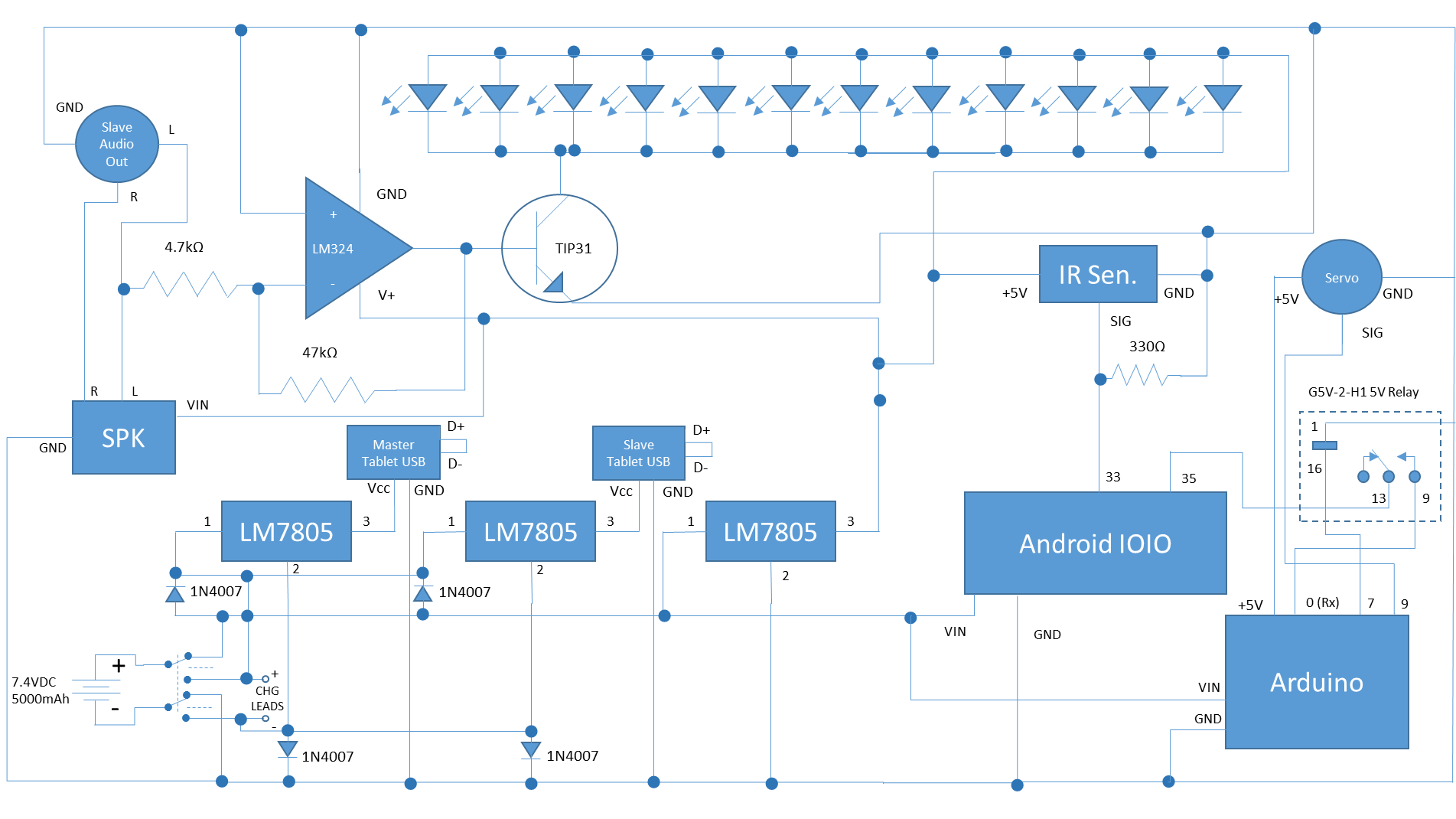


Figure 11 – Electrical schematic. Includes components internal and external to enclosure.

Most of these components reside on the main board, however there are many which reside elsewhere in the system. Components such as the servo, IR sensor, and speaker are mounted externally. Other items such as the relay, Arduino, and toggle switch are contained within the enclosure, just not on the main board. The mounting holes for these components were discussed in the previous section. With the above circuit in mind, the components were arranged on the proto-board for soldering. One way of doing this can be seen below in Fig. 12.

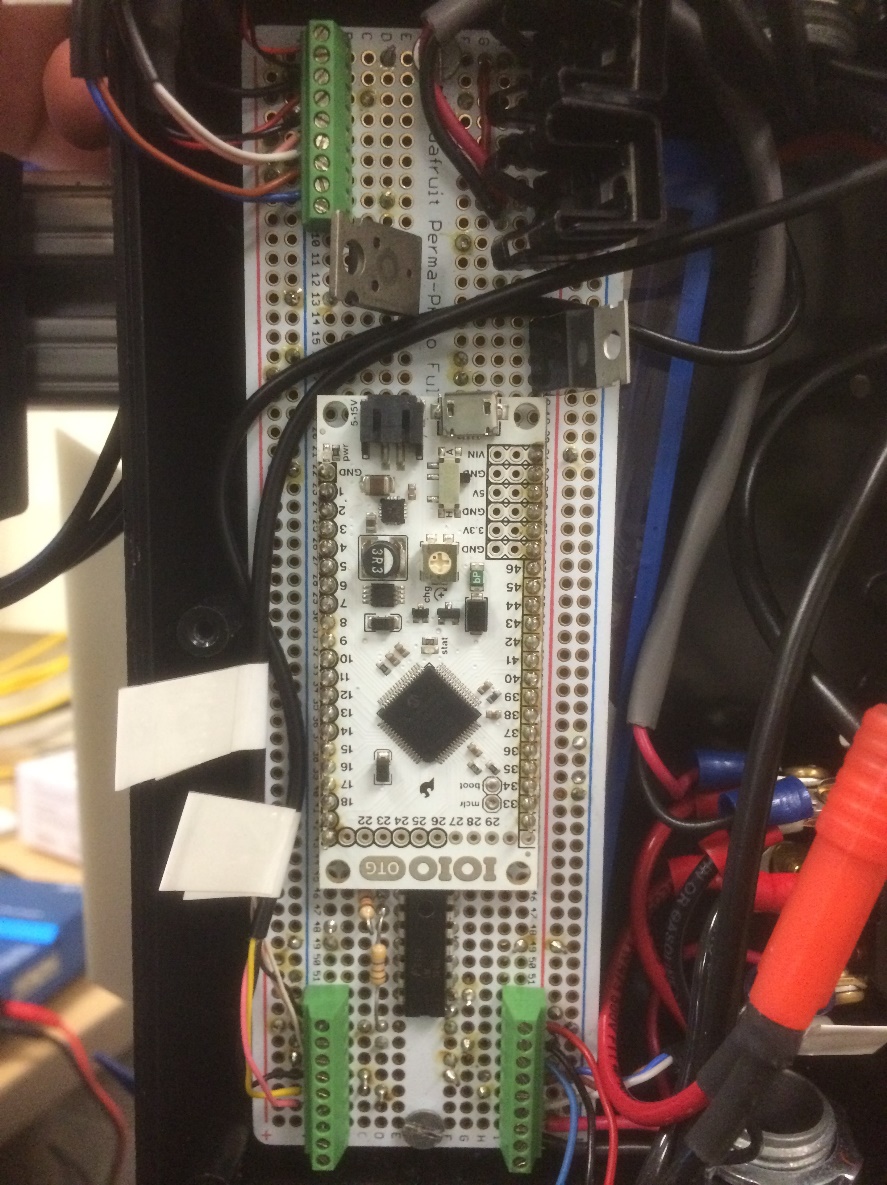
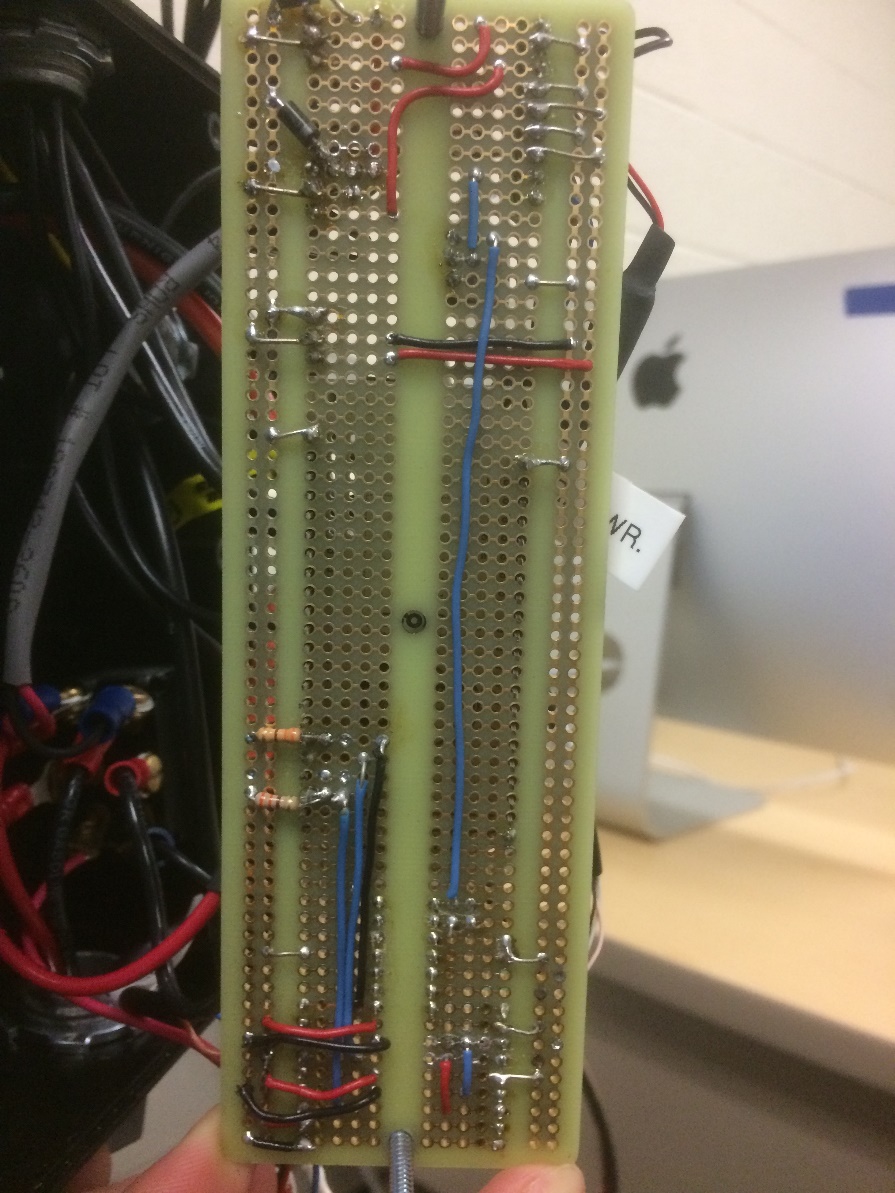
 

Figure 12 – top and bottom views of the main circuit board.

Above is the configuration we have chosen to use, where the picture on the left shows the top of the board and the right shows the bottom. The picture on the right is a 180 deg. rotated view of the picture on the left. The green 9-pin screw terminals are used to connect external electronics to the board. Heat sinks are used on two out of the three LM7805 voltage regulators. These regulators are responsible for charging the tablets which can draw up to 0.75 Amps while charging, thus requiring heat dissipation via the heat sinks. Once this board has been assembled, it can be fastened to the inside of the enclosure. The battery should be first slipped in, and then the main PCB with 1.25 in. standoffs can be screwed into place. Next, the toggle switch, Arduino, and relay can be installed and wired. It should be noted that anything components that are in direct contact with the battery are wired using 14AWG wire, where as anything that is 5VDC and less is wired using 26AWG wire. The toggle switch, relay, and Arduino wiring can be seen below as seen below in Fig. 13 and 14.

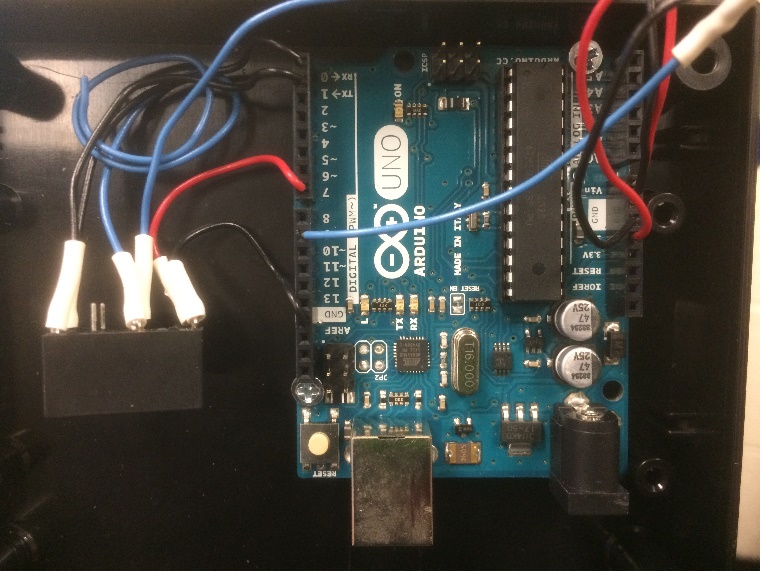
 

Figure 13 – Toggle switch mounted & wired. Figure 14 – Arduino and relay mounted and wired.

Wires can now be run from external components to the enclosure. All of the wires should be run neatly trough the channels of the frame into the enclosure via the strain reliefs. In our case, we found the best way to run the wires was to run all wires pertaining to the head through the right side strain relief, the charge leads run through the bottom strain relief, and everything else runs through the top. The charge leads include both the main charger connections as well as the balance port for the two cells. 3 pin JST-XH male-female balancer cables are used to extend the balance leads outside of the enclosure so that they can be plugged in without taking the enclosure cover off. A female T-style connector is used for the main charge leads for easy and safe connection to the charger. When running the cables from the enclosure to the slave tablet, it is best to run these cables in two separate channels in the frame. This is because one cable is used for charging (high current) and the other cable is used for audio signal (low current). When these two cables are close to each other, the high current cable induces a voltage (electromagnetic noise) on the low current cable, ultimately resulting in unwanted audible noise heard from the speakers. It is for this reason that these two wires should be kept away from each other as much as possible. The wires can be hidden in the channels using the T-slot cover, seen below in Fig. 15.



Figure 15 – T-slot cover used to hide wires.

## Software

With the mechanical and electrical components taken care of, it is finally time to prepare the software to be loaded onto the tablets. The master tablet can be loaded with the SecuroBotMaster application. This app does not require any intervention, but if you would like, there are several different eye options available included in the drawable folder. You can change the code to use these files instead if you wish. The next step is to set up the SecuroBotSlave app. Before this can be loaded, you must first take care of a few preliminary things. You must create an account with mashape.com and purchase a subscription to the SIRI API by pannous, which can be done here:

<https://market.mashape.com/pannous/siri/pricing>

Next you will have to create a twitter account for your Ada, and then sign in to Twitter’s app management system with your username and password by going here:

<https://apps.twitter.com/>

Here, you will start a new app and fill out all the required fields. You should see four different keys – Consumer key, Consumer secret, Access token, and Access token secret. You will then take your mashape SIRI key, your four twitter keys, and your twitter username and enter them as the corresponding strings in the Constants.java within the SecuoBotSlave project in Android Studio. Once this is done, you can plug your slave tablet into the computer and press the green play button in Android Studio to load the app onto the tablet.

Now that you have the apps loaded on both tablets, you can install the tablets in their respective places on Ada (master on the head, slave on the arm). If you plan on having Ada reside in a public place with low supervision, it is recommended that you take some security precautions. It is best to set a screen lock, preferably one that is not easy to guess. The other thing you may want to do is to pin the app. This is a feature that was recently built into android which allows for you to lock the screen on a certain app. That way, users can’t purposely or accidentally navigate away from the Ada app. To pin an app, open the app as normal, then click the android menu button and press the green pin icon in the bottom right corner of the app view. To exit the pinned view, press and hold the back and menu buttons and then enter your password. You may also find that the screen dims after a few minutes. One way to keep the brightness on all the time is to download an app called Stay Alive. This app can be set to stay on all the time or turn off after a certain amount of time.

We hope that you enjoyed reading this article and have been inspired to create your own Ada. We look forward to seeing how the world reacts to Ada and be sure to stay connected with UNHcFREG for updates on our research. Please feel free to comment with questions.