1. Had to power light organ from the receptacle next to the downstairs work bench to avoid noise getting on the audio speakers (used the long orange extension cord). Also had to make sure the audio line going to the light organ (the one without filters on the ends) was not crossing over any servo wires or the extension cords from the light organ to the flood lights. Get another filtered audio line to use from the PC to the light organ to reduce or eliminate this problem or use some type of isolation scheme (optical?). Web searches indicate that removing the ground fault plug using a 3 to 2 prong adaptor helps although not sure why this would? Also the jack on the light organ was a little flakey and required the plug to be slight pulled out to work so need to work on that too (just need to verify continuity of the jack wires inside the light organ to the other end of the audio plug when it is plugged in). Using a pillow to lay the audio lines on helped separate them from the other lines.  
   Ensure that the volume level control of the PC is set appropriately otherwise if it is not high enough the light organ won’t work. The organ has gain sliders on it as well so also check those although they are typically left in the same positions used on Halloween. Used the PC 65% volume level to get the organ to work at its current settings. Used the LL and MM outputs of the organ and that seemed to work ok.
2. Get a plywood board that can put both the terminal strip wood board, breadboard, and Arduino on as one unit to make it easier to move it, and a way to attach them to that plywood board.
3. Eventually to eliminate position recordings size limits, see if can get an SD card attachment for the Arduino so can store them no there. Not a big deal at the moment as can store all the current recordings (50msec interval) and total program memory taken up is 97K out of 256K.
4. Didn’t use the air cylinder this year so not sure if that’s working ok still.
5. Get the mouth servo wire on a quick connect/disconnect plug like all the other connections at the prop
6. Look at replacing the bottom half of the eye LEDs connection wire from the head with the 3 line servo wire to make it more flexible.
7. Check to see if can use the aluminum servo hub in place of the plastic one on the mouth servo although the plastic one seems to work ok.
8. See if can get some swing point on the left side of the mouth connected to the head to provide more support as didn’t use the backup/down since adding the mouth servo. Not sure if back up/down will cause problems with this so need to try out.
9. Ensure Foscam is working ok with laptop a few days before.
10. Got IMUs now so that can attach them to my head and mouth so that prop mimics my movements. Head IMU can be attached with its rubber band and duct tape to the top of a baseball cap (see orientation markings on the IMU). Mouth IMU is permanently attached to a chin guard with an elastic strap to hold it onto my head. Look at adjusting the software scaling for the mouth movements to optimize it so mouth opens wider than normal mouth when normal mouth fully open.
11. Need a way to feed mic audio to the prop so that can do live interactive prop operation (use the web cam as the eyes and ears to see and hear what’s going on, and the mic for talking and the head gear for head and mouth movement). Figure out how to get a mic into the laptop and then out to the prop and yet have headphone out of the laptop be the audio from the webcam without much audio delay. May need separate device for Mic for this other than the laptop? The Mic audio really needs to be no delay to be in sync with mouth movements.
12. See if there is some way to make a skin cover for the bottom of the mouth to the neck to cover up the servos and make it look better.
13. See if there is a way to limit the head from completely bending over where the chin hits the neck pvc pipe, when the servos are not powered. Possibly use a spring that has a little give and then attach it along with some slack wire to the center of the vertical servo metal connection so that when that servo is holding the head at 160 degrees or 20 degrees the slack/cable is completely taught and starts to pull on the spring. If the spring has a little give it can cushion the head when it drops from 90 degrees to those positions.
14. Have the servo device class have a method that limits the servo setting to provide a better way to limit this in one place (possibly in the base device class so every one inherits it)
15. See if can also have the prop audio go to the loud speakers (although the prop speaker did seem to be loud enough)
16. Have light shining on the vampire prop itself to make it more visible as people can stand in the way of the porch light that is the only light on it now.
17. Last time used base had a problem trying to fit the prop into it and ended up not using it? Will need this working if want to use back up/down.

Enhancements -

1. Making the prop skeleton arm move with servos : one for the hand, up/down two for the lower part of the arm (up/down and twist) and two for the upper part (up/down and horizontal rotation side to side) where the upper part would need to be geared to get the force needed to lift the arm or a really strong servo. Want the arm to move at a decent natural speed. Would need another 2 IMUs for choreographing for this (already have one but having a spare is always a good idea). It appears that should be able to have another 9 servos that the Arudino Mega can control by using 16 bit Timers 3,4,5 (3 servos for each timer and timer 1 already used for the 3 existing servos)) which currently no other arduino mega packages I use actually use, or plan on using use, as far as I can tell by searches. See the Atmega2560 datasheet for any difference in using those from timer 1 (17.2 Overview and on in that datasheet). Would need to modify the PWMServo package to use these additional timers. Optionally there is a adafruit shield that uses I2C that can control a massive number of servos ( it is 14 bit instead of 16 bit but that should be good enough for 1 degree changes). I2C should be fast enough even to command 9 servos every 50 milliseconds.
2. See if using some type of motor instead of the air cylinder would work for back up/down as it would be easier to control the stopping speed than the elastic cord mechanism we have now with the air cylinder. The only issue would be the speed it could achieve and the toque it can produce (has to be at least as quick as the pneumatic cylinder). Or if there is a way to more accurately slow down the pneumatic cylinder at the end points. Or could use proportional valve controllers or possibly servos hooked to valve controller and Arduino PIDs (proportional integration derivative) control using feedback of the position the cylinder is at (haunt forum has posts on this in the pneumatics forum)

2019 – 39 degrees high 22 low, snow on ground but not sidewalks - Count 60 Trick or Treaters