1. Had to use 3 to 2 prong adaptor (to disconnect earth/gnd wire) on light organ power plug to avoid servo noise getting on the audio speakers as well as falsely triggering the light organ lights. Need to check inside light organ to see what that gnd is connected to as it appears noise is getting on this gnd/earth wire which gets back thru the circuity to the audio line in and then back into the audio speakers. 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).   
   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. Eventually to eliminate position recordings size limits, see if can get an SD card attachment for the Arduino so can store them on 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.
3. Didn’t use the air cylinder again this year so not sure if that’s working ok still.
4. 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.
5. 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.
6. Ensure Foscam is working ok with laptop a few days before (use Internet explorer tab in chrome to access all functionality)
7. The up/down side of the IMUs is important in that if an IMU is placed upside down a surface (the chip side of the board is the up side , i.e pointing up, the solder side is the down side ) it can cause the yaw (heading) readings to go haywire when the pitch of the IMU is outside a certain range. If upside down, then when keeping the heading direction the IMU is pointing constant and moving the IMU so the pitch starts to change beyond a certain range (more than 30 degrees from horizontal pointing down) the IMU heading reading suddenly jumps to large values like 255 or greater. May be able to use the IMU config reg remap axis to fix this but simply having the IMU upside pointing up fixed the problem.
8. For the Mic to prop speaker the use of the MAX9814 with AGC disabled (TH pin connected to MICBIAS pin) eliminates the mic from picking up sounds far away from the mic.
9. **Remember to connect mic audio**. In 2020 had forgotten to do this for the first few TOTers and they couldn’t hear me talk.
10. 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.
11. 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. When servo class invoked the limits for the servo can be passed. Currently I set the servo to around 140 degrees before removing power and use a pvc pipe insulator to cushion the mouth hitting the pvc pipe when power is removed. That works as long as I remove mouth power before removing any other servo power. May be able to have a limit tab right at the servo hub and servo block to do this instead so don’t have to worry about power removal sequence as much.
12. Have light shining on the vampire prop itself to make it more visible as taller people can stand in the way of the porch light which is the only light on it now.
13. 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.
14. Currently have quick connect disconnect at the prop itself so end up rolling up the wire/cable from prop to controller board, putting it on the board and transporting it with the controller board to the dining room and then unrolling the wire and cable and reconnecting it. That seemed to work ok.
15. Did add blocking capacitor to receive audio from prop before it goes to the Headset headphone which helped overdrive of speaker when headset volume control turned way up
16. Use of quaternions instead of Euler angles and then converting to Euler angles didn’t seem to help much in the arm IMUs where pitch can go wonky when Roll is above 85 deg but need to investigate this more. In the Adafruit\_bno055 class there is code that has special case when roll is 90 degrees so that pitch is always set as 0 and yaw works as is, but that did not seem to help. The basic problem is at certain points of the arm position (roll at or close to 90 deg), moving just a little off the position can cause the pitch to change drastically to move to this new slightly different position because yaw and pitch do the same thing at 90 deg roll. We do limit the use of IMU roll angles over 85 degrees (prop stays where it is until go back into range) but we still could get this small change causing large change issue in the IMU when we go back into range.
17. Added servos to arm and elbow and that worked out well. GoBilda servos had enough torque and speed to do the job and are dead quiet when holding a position even when under load.
18. Did not find a good way to attach the IMUs to the arm/forearm without the flexing muscles causing unwanted movement of the IMUs. Maybe lifting the 1x2s off the arm so they only contact the arm at the ends would help. Possibly replacing the elastic Velcro strips with something thinner would also help. For now used two 2x2s connected with a hinge and put IMU on each to simulate the arm and forearm then manipulated this by hand to move the arm/elbow servos.

Enhancements -

1. 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)
2. Update code so that player registered devices override interactive prop device movement when player is playing. This allows playing short mouth and audio sequences during interactive prop movements (like a scream from the mouth).
3. See about having voice activated mouth control option. Purchased an audio level board, see if can use amp output of it in some manner instead of connecting off LEDs on that audio level board for audio level. The audio needs to have enough release time to hold the level so that the Arduino is not taxed sampling that audio level.
4. Make LED eye brightness controllable. Either use DtoA output or pulsed width modulation although use of PWM reduces max number of servos can have connected.
5. See about using a GoBilda servo for the head nod instead of the current Hitec servo as the GoBildas are dead quiet when holding position even when under load and they have as much torque and speed as the Hitec servos when run at 7.4 volts.
6. See if can also have the prop audio go to the loud speakers (although the prop speaker did seem to be loud enough)
7. See if there is a way to cover/hide the shoulder servos.
8. In the software where the servos are attached and the pulse width and degree limits are passed see about having a function to calculate the passed pulse width limits based on the passed degree limit and constants for that servos degrees and min pulse width. Basically so it does what the calculations in the ArmServoAngularAccelCalcs for getting the limited pulse width from the limited servo angles do. That way if you change the degree limits the passed pulse widths are automatically correct.

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

2020 – 58 degrees on Halloween night, clear and windy. Short of wind it was very nice night. Count 40 TOTers. Had candy in bags on table and used vampire prop to interact with TOTers only.

2021 – 50 deg on Halloween night, clear and windy. Turned out to be nice night for TOTers. Count 31. So had a lot of candy left as was expecting a lot more TOTers after the 2020 pandemic year.