

1) Problem Statement/Research Question and Background

Physical therapy is hard, especially for kids. It would be wonderful to be able to cheer children up at their physical therapy appointments - obviously because it's nice for kids to be happy, and also because happy kids are more likely to put effort into their therapy. The clinicians at the Child Development and Rehabilitation Center (CDRC) have continuously tried to figure out ways to make therapy appointments more appealing to children. A few years ago, they had the idea of raising a dog and bringing her to help entertain kids at their physical therapy sessions. The dog's name was Shelby, and she was adorable and sweet, but her personality was much too spunky for the job. She had a tendency to jump up on people. Her main contribution to the office was chewing through a couple of phone lines; she never made it into sessions with the kids. Shelby is now the happy pet at the home of one of the physical therapists. We proposed to the physical therapists that we might build a robot to try to make appointments more fun for the kids. Our robot, Shelbytron, was named in Shelby's honor. So far, Shelbytron has not chewed through any phone cords.

It was the physical therapist's request that the robot would be used to encourage children to practice walking down the halls of the clinic. The therapists asked for the robot to be able to autonomously lead a child down the hall, stop to wait, and entertain the child as it went. With clinician input, we decided it would be best if Shelbytron's designated pathway would be the north and west hallways of the CDRC, so that it wouldn't disturb or distract people in offices in other hallways. Our discussions with a physical therapist also led us to decide that the robot dog should sit in a little wheelchair - the therapist suggested that this could help to normalize wheelchair use, and of course using wheels as opposed to trying to make a robot that could walk was much easier from a technical standpoint. The physical therapist requested that the robot have handles and be large enough that a child could have the option to use it somewhat like a walker and pretend to take Shelbytron for a walk.

2) Methods/Approach/Solutions Considered

We carefully discussed as a team which parts we would get for the robot, after considering their pros and cons. For example, we chose a nickel metal hydride battery, instead of a lighter lithium battery, because we wanted it to be safer. We decided to incorporate a battery charger into the robot's body so that the charger couldn't get lost. While we considered using a single board computer, we chose an Arduino compatible microcontroller (specifically a Teensy 3.6) so that the robot could boot up faster.

A major decision about Shelbytron was what sorts of sensors and system would be best to use to help the robot navigate down a hallway. We decided against using sonar sensors, because we didn't want any ultrasonic sounds to bother any (real) therapy dogs that might enter the

clinic. We chose instead to use infrared sensors. This meant that Shelbytron would strictly be an indoor robot dog, since the infrared sensors would be overwhelmed outside in sunlight. The therapists were planning to only use Shelbytron inside the clinic anyway.

We considered equipping the dog with touch sensors so that it could tell that a child was close when the robot got petted, but we decided against that because we were afraid the dog would be too fragile to be touched repeatedly. Shelbytron actually ended up quite sturdy, but our decision turned out well given that Covid came along.

Yet another question was whether to use an off the shelf remote control or make our own. If we had made our own, we could have had more buttons available for the different features, we could have added a joystick to control the robot, and we could have had better range. However, we chose an off the shelf TV remote because it had the advantages of being light and small, and of not needing to be charged often, and of being sturdier.

One approach we took was to err on the side of adding extra sensors rather than too few. A TA in a class we took at Brown advised that it's better to avoid wishing later that you had a particular sensor. It was a bit difficult to know ahead of time exactly which sensors we needed because it wasn't until we tested the robot with children that we were able to know which features were actually most wanted and needed. We also didn't know exactly how we would program the robot. What this means is that the robot has a pixy camera as well as an orientation sensor which are not currently being used. We thought the orientation sensor might be helpful for knowing the rotation of the robot, but as it turned out, the wheel encoders sufficed. We had thought about using the Pixy to read "street sign" tags on walls so that Shelbytron could drive in any building with the proper signs up, but testing with kids and talking with the physical therapists taught us that we should stay in the widest halls and that parents were happy to drive the robot manually if needed. There are advantages to having extra sensors - they allow for additional features to be added in the future. In the future, Shelbytron could be programmed to navigate halls other than those at the CDRC.

3) Description of Final Approach and Design

Benjamin was fortunate to be able to create a CAD model of the robot in a class at Brown taught by Professor David Bamford, in the engineering department. Professor Bamford was particularly helpful in advising how to model the complex shape of the dog with surface features, and how to make the 3d printed parts of the dog bolt together.

Since the dog was larger than any 3D printer we had access to, we printed it in 5 parts, which Benjamin then pieced together. The eyes of the dog are translucent and have RGB LEDs behind them. The tail of the dog is attached to a servo so it can wag. The neck of the dog was constructed so that its head can pan and tilt on two axes.

The frame of the wheelchair is made from aluminum, which Benjamin brazed and epoxied together. He 3D printed slippery plastic inserts and bolted them to the handle tubes to allow them to telescope smoothly. The seat for the dog is also the electronics box which Benjamin laser cut from wood.

The large back wheels were constructed with discs covered with individually addressable “dotstar” LEDs. These were covered with frosted acrylic sheets, wheel rims we 3D printed, and rubber tread. The wires to the wheel lights go through slip rings which let power get to the turning wheels. The gearmotors on the back wheels have encoders that give feedback for precise control. The front wheels are casters; Benjamin machined adapters to attach them to the frame in front.

There is a short range IR distance sensor on all four sides of the robot. There are longer range sensors on both the left and right sides of Shelbytron; these sensors sit in 3-D printed turrets that can be turned with a servo in order to scan the surroundings and detect whether a child has come close to Shelbytron while it is waiting for the child to catch up.

For audio, Shelbytron’s Teensy controls a “WAV Trigger” board to play sound files from an SD card. The attached speakers are loud, but there is a volume control.

We installed a button on the back of the robot as a backup to the start and stop buttons on the remote. A RGB status light on the button also shows whether the robot is running or stopped, and whether the battery is low. Next to the button, there is a touchscreen on the back of the robot which is integrated with the many settings variables of the robot. The screen has menus for tuning values without needing to reprogram the robot, and it also gives feedback when the remote is used.

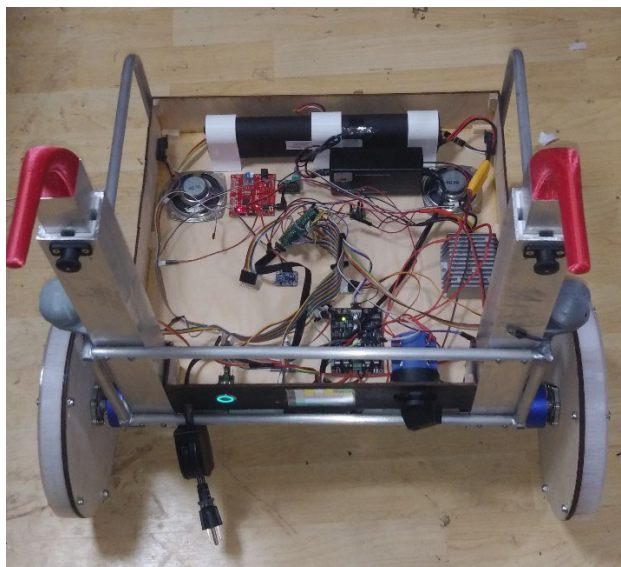


Figure 1: A view of Shelbytron with the dog and electronics box top disconnected. Most of the space of the electronics box is taken up with a large ribbon cable and the wires that branch off from it. The battery and speakers are towards the front, the Teensy is towards the center, and the power switch and motor controller board are at the back.

While Benjamin built most of the mechanical parts of the robot, Joshua was in charge of programming it. Shelbytron's software is written in C++ with Arduino libraries. Joshua wrote the "JMotor" library he used for motor control.

One of the biggest challenges was programming Shelbytron to move down a hallway autonomously without bumping into the walls. This was particularly difficult at the CDRC, because the halls there aren't straight, and there are doorways along the way. We wanted the robot to go down the hall roughly in the center, but if we had simply programmed it to keep an equal distance on its left and right sides, it would have zigzagged too much.

The algorithm we ended up with had two parts to it. One piece of code was designed to find an overall direction of the hallway. After brainstorming and testing a few different methods, we realized that the robot could take measurements as it moved, and that these measurements could help to refine the direction that the robot was heading. Every few centimeters, the robot measures how much closer or farther it has moved from the walls on both sides of it. The robot then uses trigonometry to calculate how it is rotated compared to the walls of the hallway. These are averaged over time to find the overall direction of the hallway. If at any time there are dramatic changes in the distance from the robot to either wall, then these measurements are discarded so that the robot isn't thrown off by obstacles or irregularities in the walls.

A second part of the code helps the robot to stay centered in the hallway. To do this, the robot uses a proportional control loop to try to keep the same distance from the two walls on either side of it. The output of the loop is how much the robot should turn right or left from the heading calculated in the first part of the code. To keep the movement of the robot relatively smooth and straight, the robot can only rotate a limited amount and at a limited rate. Joshua spent a lot of time using trial and error to find a balance between having the robot be responsive to its environment and having it avoid sudden movements.

The final robot has four modes. The first is autonomous mode, in which Shelbytron automatically drives in a loop through the north and west halls of the CDRC, turning and waiting for the child to catch up every few meters. While Shelbytron is waiting for the child, it turns its wheelchair 90 degrees and then turns its head another 90 degrees to "look" at the child. It also wags its tail and says encouraging phrases. When the child gets close to Shelbytron, one of the wall-following distance sensors detects this, and the robot turns forward again and resumes its way down the hall. When Shelbytron gets to the end of a hall along its designated loop and is ready to turn around to drive back the other way, it tells people this in a friendly manner. A second mode is available for people wanting to drive the robot with the remote control. Within this mode, the person operating the remote has the option to have some hall following assistance or take more manual control. In the third mode, the wheels on the robot don't move, but Shelbytron can move its head and its tail, play music, tell jokes, and say some other phrases as directed by the remote. The final mode is basically reserved for us to use for adjusting and testing the code.

The robot has various features designed to entertain the children. Rainbow lights in the wheels flash attractive patterns. Shelbytron can tell jokes, and when it does this, the lights in its eyes wink. It can greet a child, say goodbye, and say a number of encouraging phrases. Shelbytron is programmed with a repertoire of 100 songs (and more could easily be added). The songs are divided into four different playlists: one for the youngest children, one calmer, one designed to be more exciting, and one geared towards the oldest users. The music was chosen to represent a diverse range of music styles and cultures. The volume can be adjusted. All of the above features can be operated with the remote control.

Additional aspects of the robot's operation can be adjusted using the touch screen on the back of the robot. For example, it is possible to dim or turn off the lights on the wheels, turn off the speaking or music on the robot, and adjust the driving speed of the robot, in order to accommodate sensitivities or preferences that the child may have.

For safety purposes, Shelbytron uses its proximity sensors in the front and back so that it can detect obstacles and stop before hitting them when driven manually and when following the hall. The stop button on the robot and the remote make the robot turn the motors off very quickly instead of its normally smooth motion. While it can be startling that sometimes Shelbytron will literally skid to a stop, it was important to make the software for disabling movement act as reliably and quickly as possible.

The robot can detect when its battery is running low. In these cases, Shelbytron will complain twice about how tired it is, before it automatically stops driving. There is also a battery indicator on the screen to inform the therapist how much charge is left.

4) Outcome (Results of any outcomes testing and/or user feedback)

We were fortunate to be able to test the robot with four different children at their therapy appointments. We are indebted to Alieh Zamany, a physical therapist who invited Shelbytron to meet some of her clients at this early stage. We made several adjustments in response to our observations and the feedback we got.

We adjusted the range of speeds because some of the children walked more quickly than we had anticipated. We added some greetings so that the robot could be a bit more interactive with the children. We realized that some kids would want much more complex interaction than we were able to program, so we expanded the aspects of the robot that could be controlled remotely. We made changes to what the buttons on the remote control do to make it more user friendly. We learned that it often takes a while for the therapists to get children situated with their walkers, so at the therapists' request, we added jokes (which the therapists probably regret asking for) as well as other features to entertain the kids during this time. We enjoyed being able to ask children what their favorite song was and surprising them with it the next time they showed up to therapy.

The children we observed used their own walkers rather than push Shelbytron along, but we're still glad that we made Shelbytron large enough that it can be petted easily.

5) Cost

The total cost for the materials in Shelbytron was around \$2000. A more detailed list of parts can be found in the Appendix.

6) Significance

It was fun to see kids light up when they saw Shelbytron. It was clear from their smiles and attention to Shelbytron that they enjoyed the robot. Moreover, Shelbytron was motivating. Caretakers of children using Shelbytron commented to us that they had never seen their child walk for so long during therapy appointments. Their physical therapists agreed. It would be interesting to research this more objectively in the future.

We noticed that all of the families using Shelbytron pulled out their phones to take pictures. Caretakers were able to learn to operate the remote, and they seemed to enjoy being involved in their child's physical therapy in this way.

Families who were in the building at the same time often stopped to watch the robot for a while. Shelbytron provided good entertainment for children waiting for their appointments. It was clear from everyone's comments that they appreciated the efforts to make the CDRC a more fun place to come to.

There aren't any commercially available robots like Shelbytron, designed to encourage children with mobility impairments to practice their walking. Shelbytron is a proof of concept that robots can be useful in physical therapy to motivate children and put smiles on their faces.

Acknowledgements

Special thanks to the families at the CDRC who helped us to test Shelbytron (due to privacy considerations, we cannot name them). Thanks to these CDRC clinicians who also gave us feedback: Alieh Zamany, PT, DPT, PCS; Dianne Hrubec, MS, PT; and Randall Phelps, MD, PhD. We appreciate the guidance in Computer Aided Design from Professor David Bamford at Brown University. Thanks to Liz Phelps for being the voice of Shelbytron. Thanks also to composer Paul Safar for allowing us to use his original music for our project.

Appendix

Here is a link to some progress pictures from when we were building Shelbytron:

<https://shelbytron.tumblr.com/>

Here is a link to the code: <https://github.com/joshua-8/shelby>

Here is a link to our parts list: [Shelbytron parts](#)

Here is a link to a basic manual for using Shelbytron: [Shelbytron Manual](#)