

## ***Project 1 Summary***

Our device successfully controls the cursor on a computer screen. This prototype should be used as an alternative mouse for those without the dexterity or fine motor skills to control a standard mouse or make typical hand gestures to interface with a trackpad. However, we take advantage of a universal design approach such that any computer user with full range of motion in their fingers and hands can still comfortably operate the accessory.

The materials for the project include an Arduino, an Adafruit VL6180X Time of Flight MicroLIDAR Distance Sensor, an HC-SR04 Ultrasonic Distance Sensor, and requires Python to be downloaded on the computer to be controlled. The Time of Flight Sensor is used to control the position of the cursor along the x-axis, while the Ultrasonic Sensor is used to control the position of the cursor along the y-axis. Using the Arduino map function, the minimum and maximum readings of both sensors are mapped to the size of the screen. This number is hardcoded into the program (in our case we used a 2018 Macbook Pro which has a resolution of 1440x900 px). If a hand (or other object) is not detected, a “No Convergence Error” is printed.

We were able to determine the size of the screen and then move the cursor by importing PyAutoGUI in Python using pip. Once the x and y values from the sensor are adjusted to match the screen size, the cursor can be moved by the function in pyautogui called `moveTo(x,y)`. For the Python code to communicate with the Arduino code, serial must be installed through pip.

While the cursor can be controlled by this prototype, the sensors are extremely sensitive to hand movement. It is difficult to control as the user's hand can quickly move out of range. A future design would incorporate a proper casing or enclosure that hides the wiring and mounts the sensors in their proper position respective to the x and y-axes. This would also provide a better visual representation of how far a user should move their body before there is no convergence. Note, there is no feature using these two sensors to "click" the mouse, but there is a `click` function in the PyAutoGUI documentation to do so. We suggest using a knock sensor to incorporate this feature.

*Due to limited access to the WakerSpace, we did not always have all our materials while working on the project, but the code successfully ran when we switched to connecting to an Arduino Nano owned by Henry. Moreover, for the testing we used a second breadboard so that the wiring would not interfere with the sensor readings. A picture of this version of our prototype is below. A second circuit diagram using an Arduino Nano is uploaded to our GitHub Repository.*

Ethan Harrison and Henry Lynn  
Dr Pauca  
CSC 332: Mobile and Pervasive Computing  
February 20, 2022

