Mason A. Channer | 785696 | 1/8/2018

**Computer Science Project**

One of the projects I worked on last year, that was unrelated to school projects or coursework, was a second-generation people counter. A second-generation people counter uses thermal sensors to detect heat sources and count the number of people navigating a certain passage. The sensor I used, HC-SR501, gives you the best compromise between accuracy and cost. However, thermal sensors can only cover a short distance, accurately count multiple people entering concurrently, and requires thermal conditions to remain the same on average.

I began building the [People Counter](https://i.imgur.com/l9eA2zU.png) using an Arduino, three [HC-SR501 PIR Motion Detectors](https://www.mpja.com/download/31227sc.pdf), and a [Raspberry Pi 3 Model B](https://blogs.windows.com/buildingapps/2016/02/29/windows-10-iot-core-support-for-raspberry-pi-3/) running Windows 10 ARM. I took the plastic cover off the HC-SR501 sensor and used a piece of [PVC pipe](http://www.rehomepipe.com/blog/wp-content/uploads/2015/12/045.jpg) to focus the sensor into a beam. I spaced the three HC-SR501 sensors apart and placed them above my door pointing down. The spacing I used was my width plus two inches on either side. I have completed the barebone application and wrote the database connection code for the application. The Pi stores the date and time for each trigger into a database and syncs the database with an online database every night at 12 a.m.

 I learned how to apply my knowledge from my previous classes, such as Digital Logic Design and Computer Architecture, on how to read specification sheets. This allowed me to understand the circuitry of the HC-SR501. On the HC-SR501, there are two dials that control the output timing and the sensitivity. I wanted to reduce the time the sensor stayed high when it was triggered, so that I could control it programmatically. To solve this, I analyzed the circuitry on the specification sheet and I applied some solder onto the R13 and R33 transistor. This would by pass the transistors function of increasing the time the signal stayed high. I also learned how to work with the C# language, HTML, and binary on a low-end device.