

The Sensibility of Sensors

1 Introduction

The sensibility of sensors is not realized, appreciated, or used in most instances. While sensors aren't practical in all situations, they can be very useful. In Botball, especially as a new team or team member, sensors can seem daunting and pointless. Why program sensors for redundancy when there's pipes to back up against? What's the point when most game tasks require only simple dead-reckoning and easy programming? Why complicate your program and add loops and variables for sensors? The contradictory answer to this set of questions is that sensors simply make your life easier. While there are several good arguments that one can bring up against using sensors, the benefits from learning how to use them are bountiful.

2 Sensors: What Are They?

Sensors themselves are incredibly simple electronics. There are several sensors provided in the Botball kit: touch sensors, light sensors, IR sensors, and distance (or ET) sensors. An addition this year to the plethora of sensors was the Xtion Asus depth sensor, but that's a completely separate topic. All of these sensors have one or two specific purposes. Most of the sensors are pretty self-explanatory. However, IR sensors have a dual purpose. They can be used to track lines or sense incredibly small distances. This makes them very useful, but also somewhat problematic when tracking a line or sensing small distances.

3 Cons of Not Using Sensors

Last year, I didn't use any sensors, save for the light sensor, for obvious reasons. When my team went to the tournaments, we didn't do well. This result came from a combination of many reasons. This year was my first year in high school, and I decided to finally use sensors. I delved into them at workshop and quickly discovered how handy they could be. However, upon returning to our normal practices, it looked like there wasn't going to be too many opportunities to use my new knowledge.

4 Experience in Using Sensors

Part of using sensors is figuring out how to use them creatively. For example, look at this year's game board. There's only one line to track on the board, and it seems pretty pointless when you first look at it. However, if you track one side of the line, it can lead you to the lower storage area, and if you track down the middle of the line, you can go straight to the center of the board, which is helpful for orienting yourself.

5 Programming Sensors

When you first look at the programming for sensors, it looks pretty daunting, especially for a student in middle school, or one that's never coded before. The addition of while loops, if-else statements, and variables can turn a very simple motor-oriented program into a minefield of compiler errors. The extreme quantity of compiler errors one gets when writing their first program with sensors is enough to make anyone quit programming for a while. That's one of the reasons sensors aren't that popular. However, once over the first minefield, things get much smoother and simpler. This then leads to a second minefield - encountered once one has mastered sensors and left robotics alone over the off-season.

6 Benefits of Sensors

Once over these two minefields, the benefits of sensors can be continually reaped over and over again. Every year, there is a line somewhere on the board in perfect position to track. Touch sensors are very helpful for determining when to stop tracking the line, and an ET sensor could also be used for that. Need to measure really precise, small distances? Use an IR sensor. What about color? There's even a camera provided in the kit for that. Lastly, the robots can't start without a light sensor, so there better be one on the robot.

7 Sensors' First Impressions

However, in middle school, sensors look incredibly challenging. The concept of math itself frightens enough people. To use sensors, while loops and if-else statements are necessary, and variables are recommended. If all that's known is simple drive commands, the syntax of these commands are confusing and daunting. Also, as mentioned before, a simple flat back can be put on a robot and it can be backed up against pipes to straighten out and orient itself on. It really seems like there's no point.

8 Sensors are Handy

However, sensors are an incredibly handy tool to have. They help the robot orient itself on the board and even helps the robot control its effectors. With the several new additions of the slide sensor, and rotation sensor, sensors can do even more tasks that were previously pretty difficult. For example, if there was an actuator going up and down, the rotation sensor could measure the rotation in degrees to accurately stop the actuator at a precise angle.

9 What to Use Sensors For

With all this new-found knowledge about sensors, why not use them for everything on the board? There is absolutely no viable response to "But sensors can't be used for everything," because they can't. If there's too many sensors on the robot, the robot could get "confused" and interpret certain readings as something else. Sensors should be used as a supplement to the program, not something the program relies on, much like people take multivitamins every day to

supplement their nutrient intake. However, in the practical real-world applications of robotics, as seen in Google's self-driving car, that car relies on a plethora of sensors that cross-reference each other with several redundancy programs, which the limited sensor kit and programming knowledge of high and middle schoolers prevent that from happening in Botball.

10 Conclusion

So while sensors are incredibly useful and practical, they shouldn't be used for everything. Learning the different commands and loops required to program sensors can come in a lot of handy in other coding situations as well, so it's worth learning. While using sensors takes more time and can be confusing, the benefits are great and bountiful. Don't miss out – learn about sensors today!