Kimberly Mackey

Cit227

03/08/2021

**1) How do the whisker sensors work? What value does a whisker read if it is NOT activated and what does it read when it IS activated? Why does that value change (electrically)? (3 points)**

The whiskers will show a 1 when not activated and a 0 when it is activated. So, it’s a low signal with it is pressed and high signal when it is not pressed. It is electronically monitored because different actions can be taken depending on if the switch is pressed or not pressed. It this case if one is pressed the bot will go in one direction, if the other is pressed it will go in a different direction and if they are both pressed then it will do something different altogether. Also we are able to count and see if there are any patterns in case the bot is stuck in a corner then we can tell it to do something else altogether to get unstuck.

**2) How do the servos work? How do you get them to move and how do you determine the direction? How do you get them to stop? How can you make them move at varying speeds? (3 points)**

The servos are the devices that run the wheels. To make them run you must first attach and send signal to the servos with : “servoLeft.attach(13);servoRight.attach(12);” The left servo moves counterclockwise and the right moves clockwise. You set them separately to run at varies pulses in microseconds, which is also what determines rpms and in which direction the bot will go. To stop the bot you have to detach each sarvo so it stops sending a signal to it with: “servoLeft.detach(); servoRight.detach();”.

**3) How does the robot escape the corner? How does it know it's in a corner and what does it do to prevent the repetition that keeps it there? (3 points)**

The bot gets stuck in a corner it is because it's on a repetitive pattern with the same two commands so it’s stuck in a cycle that it can't break out of. Since we know the pattern of being stuck is it bouncing from one whisker to the other in a never-ending cycle, we change it. We want to count how many times its bouncing back and forth. Once it counts enough times, we know it’s a pattern and we can make an exception and tell it to go in a different direction. Thus, breaking the cycle and its free to roam once more.

**4) What change did you have to make in code to prevent the robot from moving if there is no light detected? (1 point)**

I’m not sure what you mean by change, but I set mine to read the voltage of the light. I used an if statement, so if the photocell didn’t detect enough voltage, then it, I detached the servos. If it did detect enough voltage than it was attached and able to move.

“if (voltage(A3) < 1.5)

{

servoLeft.detach(); // stops servo signals

servoRight.detach();

}

else

{

servoLeft.attach(13); // sends signal to wheels

servoRight.attach(12);

} “

And to calculate the voltage:

“float voltage(int adPin)

{

return float(analogRead(adPin)) \* 5.0 / 1024.0;

}”