**A Bit of Hypothesis**

Let's begin with a few essentials some time recently getting our hands dirty. The self-balancing robot is comparative to an upside down pendulum. Not at all like a typical pendulum which keeps on swinging once given a bump, this modified pendulum cannot remain adjusted on its possess. It'll basically drop over. At that point how do we adjust it? Consider adjusting a broomstick on our index finger which could be a classic example of adjusting an rearranged pendulum. We move our finger within the course in which the adhere is falling. Comparative is the case with a self-balancing robot, as it were that the robot will drop either forward or in reverse. A bit like how we adjust a adhere on our finger, we adjust the robot by driving its wheels within the course in which it is falling. What we are attempting to do here is to keep the center of gravity of the robot precisely over the rotate point. To drive the engines we require a few data on the state of the robot. We ought to know the heading in which the robot is falling, how much the robot has tilted and the speed with which it is falling. All these data can be derived from the readings gotten from MPU6050. We combine all these inputs and create a flag which drives the engines and keeps the robot adjusted.