Introduction

The following documentation will give the reader detailed information about the Teensy BalanceBot Mk1 project.

The goal of this project is to develop a balancing robot from the ground up. This means going through the full process of creating a product/project, everything from System analysis and development of the system to conducting tests to verify that all of the objectives are met and a complete and detailed documentation of all of the steps.

System Analysis

This section covers desired functionality of the system, how the system can be modelled, what solutions currently exist and how they compare to the system being developed.

Desired Functionality

Overview

This is a relatively small project with limited time for development and testing, therefore the desired functionality has been reduced to the bare minimum. The features that have been chosen are enough to create a MVP (Minimum Viable Product) that can be used as a good starting point for future projects that require a balancing robot platform. Future development possibilities and improvements are discussed later in the documentation.

Possible Uses

Despite the small formfactor of the Teensy BalanceBot Mk1, it has a couple of use cases.

* The first use case is in education. It is simple enough to be an introduction to robotics for students, but it has enough complexity for any future development.
* The second use case is in research. Robots are becoming very common in everyday life, and different situations require robots with different sets of features and form factors. With the limited feature set, BalanceBot Mk1 provides an excellent blank canvas for researchers to build upon.

Functionality

* **Angle control** - Angle control is the single most important feature for Teensy BalanceBot Mk1. Angle control goes well beyond staying in an upright position. Consider the situation where the robot is required to move from point A to point B. When repositioning, due to the nature of the robot, it must either lean forward or backward. This requires more consideration when developing the software, because the lean angles when moving are not constant, they depend on the speed of the robot.
* **Position hold** -This feature is not critical to the operation of the robot, but it is still part of the core controllability of the platform. Being able to hold a given position allows for easier testing and tuning of angle control, as well as any future features. As mentioned, this platform can find uses in education and research. In these cases, space is limited, and it is not desirable for the robot to drift away during testing.

System Model

Mathematical Model

Inverted pendulum model

Equations of motion

Controller Options

Full state feedback

Full state estimation with LQG

PID Controller

Modelling Software

Mathematica

MATLAB & Simulink

Comparison with existing solutions

Existing solutions

Implemented functionality

Hardware choices

Development tools

Development

MATLAB Model & Simulation

Non-linear model

Using differential equations

Creating a model with Simulink and Simscape multibody

Controller model

PID Controller model

Automatic PID Tuning in MATLAB

Simulation

Simulation results

Hardware

Microcontroller

Sensors

Overview & Considerations

IMU

Encoders

Actuators

Overview & Considerations

Motors

Main structure

CAD Design

Material Choices

Manufacturing Method

Electronics

System Diagrams

Microcontroller Mount

Connectors, Wires & PCBs

Power

Batteries & Battery Holder

Power requirements

Final Assembly

Fastener Choices

Assembly Strategies

Assembled Robot

Software

Development strategy

Controller

PID Controller structure

PID Tuning

Implemented Functionality

Balance mode

Position hold

Software architecture

Software modules

Final Structure

Testing

Testing strategy

Overall approach

Testing equipment

Test descriptions

Circuit Test

PCB Tests

Wiring harness tests

Encoder Test

Screen Test

IMU Test

Motor & Motor Driver Test

PID Tests

Future Development

Current Issues

Encoder inaccuracy

Issue

Solution

IMU mounting

Issue

Solution

Robot structure & assembly

Issue

Solution

Motors

Issue

Solution

Future Improvements

Hardware

LiDAR Sensor

Sonar Sensor

Radio Control

Functionality

Implementing GUI interface

Adding direction control

Collision Avoidance

Environment mapping using SLAM

Possible use-cases

References/Sources

Heading 1

1. Functional Analysis