Laboratorium Electric Bicycle Cruise Control System Operating and Technical Guide

Introduction

This guide is here to help you understand both how to use the cruise control system and how to understand the technical details behind the design and implementation of the cruise control system.

Cruise Control System User's Guide

The cruise control system maintains the electric bicycle's speed at a user set desired speed in m/s. This action is performed through automatic modulation of the throttle via an Arduino nano programmed with a simple PID controller. The Arduino Nano sits inline with the throttle output signal. When the cruise control is engaged, the Ardino works by emulating the rider's throttle inputs to maintain the speed of the electric bicycle at the user set desired speed.

The cruise control system can be engaged when riding the bike. The following outlines the steps the user needs to take to engage and operate the cruise control:

1. To power all systems the following needs to be performed:
    a. Before beginning, make sure that the dead man's ankle velcro switch is attached to its mount near the foot peg
    b. First power the Arduino nano and dead man's switch by flipping the switch to the NiCd battery pack. The onboard Arduino LEDs will flicker on.
    c. Next power the electric bicycle by flipping the rocker switch on the bottom of the Li+ battery cylinder on the seat tube. To verify the battery switch is in the "on" position, press the battery level indicator button on the top of the battery. There should be an array of **multicolored LEDs** that light up showing the current capacity of the battery.
    d. Finally, press the red switch on the side of the throttle. The **LEDs on the throttle will light up** indicating the bicycle is ready for riding
2. With the dead man's ankle velcro switch **still attached to the bicycle**, tie the switch to the rider's ankle. Another person may be needed to hold the bicycle upright.
3. Start riding the bicycle.
4. Accelerate or deccelerate to around the desired setpoint (target) speed.

5. Engage the cruise control by pressing the two cruise control buttons at the same time. The LCD screen will change to show the setpoint speed and the current speed.
    a. Note that the setpoint speed initializes itself at the speed of the bicycle the moment the cruise control is engaged.
6. Once engaged, the Arduino Nano will take control of the throttle and the user can sit back and steer.
    a. At any time, manual control of the throttle can be returned to the user by simply pressing **all the way down** on the throttle with your thumb (This will deactivate the cruise control).
7. While the cruise control is engaged the cruise control buttons can be pressed to either increment or decrement the setpoint speed.
8. To disengage the cruise control simply press **all the way down** on the throttle. A message confirming that the cruise control has disengaged will print to the LCD.
9. Steps 4-8 can be repeated any time during the ride

Reverting the Electric Bicycle to its Original Powertrain

To revert the electric bicycle back to its original powertrain, the throttle output signal (from its hall effect sensor) needs to be jumped directly to the motor controller. See below for the steps on how to do this.

1. Locate the throttle junction. This is located around the top triangle of the frame right above the head tube. See the picture below for reference:

Note: *The schematic for this junction can be found on sheet 3 of the master electrical schematic*

2. Once located, disconnect the black and yellow wires. (Note the yellow wire is sleeved in a yellow sleeve).
3. To jump the throttle signal directly to the motor controller you will need to add a jumper wire to connect the white wires together. The picture below shows the

jumper wire used to connect both the white wires together.

4. Once this is complete the Arduino should have no effect on the bike's powertrain and it should work normally again.

Cruise Control System Technical Guide

This technical guide is meant to help you understand what went into the design and implementation of the cruise control system. There is a number of documents regarding the technical details and design of the cruise control system; these are outlined here:

1. Project Technical Poster and Paper
    a. These give a good high level overview of the project and its technical details making them a good place to start
2. Davis Instrumented Bike (Wheel Rate and Calibration Sections)
    a. This gives a good description of the instrumented bike platform and how to go about interacting with it
3. Controller Design Blog Post
    a. This blog post outlines the details of the controller design process used to derive and tune the PID controller
4. Project Code Description (See eBikeSpdController Github Repository Read Me)
    a. This Read Me outlines the files contained in the repository
5. Controller Design MATLAB Code

6. Controller Implementation Blog Post
    a. This blog post describes the physical implementation of the PID controller onto the electric bicycle
7. Arduino Code
8. System Electrical Schematic

Reading these documents in the order suggested here should give you a good enough understanding of the technical design and details of the cruise control system in order to make modifications or adjustments to the system.

Technical Troubleshooting Tips

Over the course of working on the bike, a couple common issues have arisen. These are detailed below:

1. **Anderson Power Pole Contacts**
    a. Issue: Arduino won't power on or dead man's switch is malfunctioning
    b. Fix: Adnerson power poles either on the 14 AWG wire from the Li+ battery or the 22 AWG wire from the NiCd battery have loose contacts.
        i. These contacts will need to be manually pressed back into place by pressing on them from the back of the housing with a small diameter screwdriver or Allen wrench
            1. It helps to do this while the two connectors are connected to each other. **CAUTION: DO NOT SHORT THE BATTERY**
        ii. It's a good idea to check that these contacts are fully in place every time the connectors are connected/disconnected as they come loose easily
            1. To do this look at the rear of the connector housing and see if the contact is sticking out further than typical
2. **NiCd Battery Charge**
    a. Issue: Arduino won't power on or dead man's switch is malfunctioning
    b. Fix: Charge the NiCd batteries via the 14.2V terminal
        i. NOTE: I don't believe there is any over/under charge protection built into the batteries and don't trust the charger so it is important to periodically check the battery voltage and temperature while charging these batteries
        ii. The same goes with discharging the batteries. They should be periodically checked to ensure they are not overdischarged while being used to power the Arduino

iii. Read up on charging NiCd Batteries https://batteryuniversity.com/learn/article/charging_nickel_based_batteries

3. **Dupont Connectors**
    a. Issue: there is an open circuit somewhere
    b. Fix: Check the Dupont connectors. Typically the 1x1 Dupont connectors have been found to come loose. There is a 1x1 Dupont connector on the dead man's switch circuit that has come loose before (it is currently wrapped in heat shrink tubing)
4. **Arduino Board Solder Joints**
    a. Issue: Open Circuit
    b. Fix: Check the solder joints on the Arduino board as, over time, these may break
5. **Battery Low Voltage Imbalance**
    a. Issue: Bike motor cuts out after a few seconds of riding
    b. Fix: Check the Li+ battery. If the cells in the battery become out of balance they will trigger the low voltage cut off of the battery management system in the Li+ battery and cut power to the bike. To check this you will need to disassemble the Li+ battery and probe the voltage of each cell on the battery management board. If you find a significant imbalance between cells this will justify buying a new battery
6. **Motor not spinning up**
    a. Issue: Motor won't spin up even though everything else is working/seems right
    b. Fix: Check the brake levers. Sometimes the metal bracket that contacts the brake lever safety switch will get stuck outside the brake lever housing. Reinsert this metal bracket to fix the brake lever safety switch and get the motor to start up again.