University of Pittsburgh at Johnstown
Department of Electrical and Computer Engineering
EE 1195 / COE 1195 Engineering Practice and Professional Development
Final Project Proposal

Cycle Statistic Tool

Date: September 19th, 2020

Submitted By: Cody Klingler
Jacob Hoffman

Instructor: Professor Gabany

Problem Identification:

Currently, there are several wearable devices available for cyclists that offer general data collection about a cycling session; however, these do not offer specific statistic tracking for the user, only general data about the location, duration, and exercise of the cycling trip. The specialized meters that are available to cyclists currently offer very few measurements. Usually they have either power measuring or speedometers and odometers. The power meters that attach to the pedal of a bike start at $350, and only measure power output. Cyclists require a better, more encompassing solution for setting goals and tracking improvement.

Engineering Formulation:

Due to dramatic advancements in technology, small-scale sensors are readily available and may be used to create a compact device which can be universally attached to a cycle. The ability to use a cycle as a housing for components allows for collection of data that is more specific to cycling than other general devices are capable of.

Proposed Solution:

The Cycle Statistic Tool is a data recording device which will be attached to the frame of a bicycle. In this position, several electronic components may be used to record statistics about a user’s cycling trip. The Cycle Statistic Tool will transmit data to a companion application that is installed on a user’s smart phone. This app will utilize the collected information to provide records of a user’s cycling sessions in the form graphs and charts that enable the user to improve their cycling skills. An appropriate Arduino microcontroller will be used as the primary board to connect all sensors and for operation of the device. A secondary Arduino microcontroller will be attached to the pedal shaft to wirelessly relay data from this position to the primary board. Live data will be displayed to the user by a display panel connected to the primary board. The entire device will require a casement which will be modeled and 3D-printed. This casement should be somewhat weather-proof and able to connect to the frame/handlebar of a bicycle universally.

The device will use various sensors to collect data about the user’s cycling session. It will consist of two Arduino Nano 33 IoT [1] microcontrollers with built-in Bluetooth and Wi-Fi technology—this is necessary for data transmission between both the controllers and the cyclist’s phone. Additionally, the microcontrollers have an inertial measurement unit (IMU) on-board which features a 3D digital accelerometer and a 3D digital gyroscope. This component will be used for calculating velocity, incline, and the angular frequency of the pedals. The primary microcontroller will be attached to the main frame of the cycle and the secondary microcontroller will be attached to the cycle’s pedal. The secondary microcontroller is necessary because any physical tethering between the pedals and frame would interfere with pedaling. Sensors for measuring torque and pedals per minute and subsequently power and gear-ratio must be mounted on the pedal shaft. The two sensors mounted here are small and should not interfere in pedaling. A basic diagram of the device's layout is shown in Figure 1.

To enable the user to view the real-time statistics of a cycling session, a built-in display will be included within the design. The information that will be displayed includes current velocity, distance travelled, average speed, incline angle, torque, gear-ratio, power output, pedals per minute (cadence), wind speed, wind direction, calories burned, and the current time. An eInk display [2] will be used due to the range of potential lighting conditions. This display offers dramatically lower power consumption and improved visibility in direct sunlight when compared to traditional displays. An LED controlled by a photoresistor will be mounted above the display to increase visibility in low-light. A breakout [3] will be necessary to buffer the information from the Arduino. Without this component, the Arduino would use most of its memory to drive the display.

To collect accurate measurements of the instantaneous velocity, both a GPS module and the inertial measurement unit (IMU) will be used. The Adafruit Mini GPS PA1010D [4] has been selected due to its compact design and speed at 10 location updates every second. Additionally, the selected GPS module will be used to collect GPS location data, dynamic travel time, and overall trip length.

The IMU will also be used to measure the current incline angle and record the overall elevation change of the trip. In the statistic tool's companion app, overall elevation change will be found using GPS information for higher accuracy.

Wind direction will be found by using a continuous potentiometer, which allows for free rotation of the control knob. This continuous potentiometer will be controlled by the wind through a large plastic fin attached to the knob by a rod. To prevent inaccuracy from riding at an incline, a 2-axis stabilizer will have to be used. The stabilizer will either be a modified phone gimbal, or it will be constructed of ball bearings and rods to reduce cost.

Wind speed measurements will be collected using an anemometer created from either a basic computer fan or some other brushless DC motor. When spun, these motors create a measurable voltage at their terminals proportional to rotations per minute (RPM). All the wind sensing equipment will need to be mounted somewhere that allows for unrestricted movement. The front of the bicycle is ideal for accuracy, but slightly inconvenient for the cyclist. The equipment should be able to mount in multiple places as to accommodate for personal preference.

Using the secondary microcontrollers attached to the pedal shaft, four strain gauge sensors [5] will measure the torque being applied by the cyclist. Strain gauges are thin strips that change in resistance depending on the flex and sheer force of whatever they are applied to. The strain gauges must be tared before each use due to change in temperature, just like an electronic scale.

The IR sensors [6] will be used to measure the angular frequency of the cycle's pedals and rear sprockets. The IR sensors output a signal when there is a bright object close to their surface. A reflective strip will be placed at one point in front of the sensor's path. This information is necessary for calculating power, gear-ratio, and cadence.

The microcontrollers will require enclosures that will be modeled and 3D-printed once the circuitry has been finalized. A waterproofing sealant may be used to protect the internal circuitry and the external sensors during wet conditions.

The mobile app component will store information from the microcontrollers that it receives over Bluetooth. Here, the cyclist can change the settings of the microcontroller, such as units displayed and the user weight used in calculations. The app will allow the cyclist to create routes that are used to compare their trips with their best and average trips. Unobvious trends that become evident in the data will be shown to the user. For example, it might be found that a cyclist tends to move fastest up 5-10% inclines at a 2:3 gear ratio, or that they tend to finish routes 5% faster overall if they pedal faster in lower gearing at the beginning of their trip. The app will aim to make it easier for the cyclist to find areas of improvement, as well as track their abilities and set goals. The app will be available on Android and iOS. C# (Xamarin) and a database in SQL or SQLite will be used for the design. The microcontrollers will be programmed in C++. A flowchart describing the workflow of the device and companion app is shown in figure 2.

Drawings:

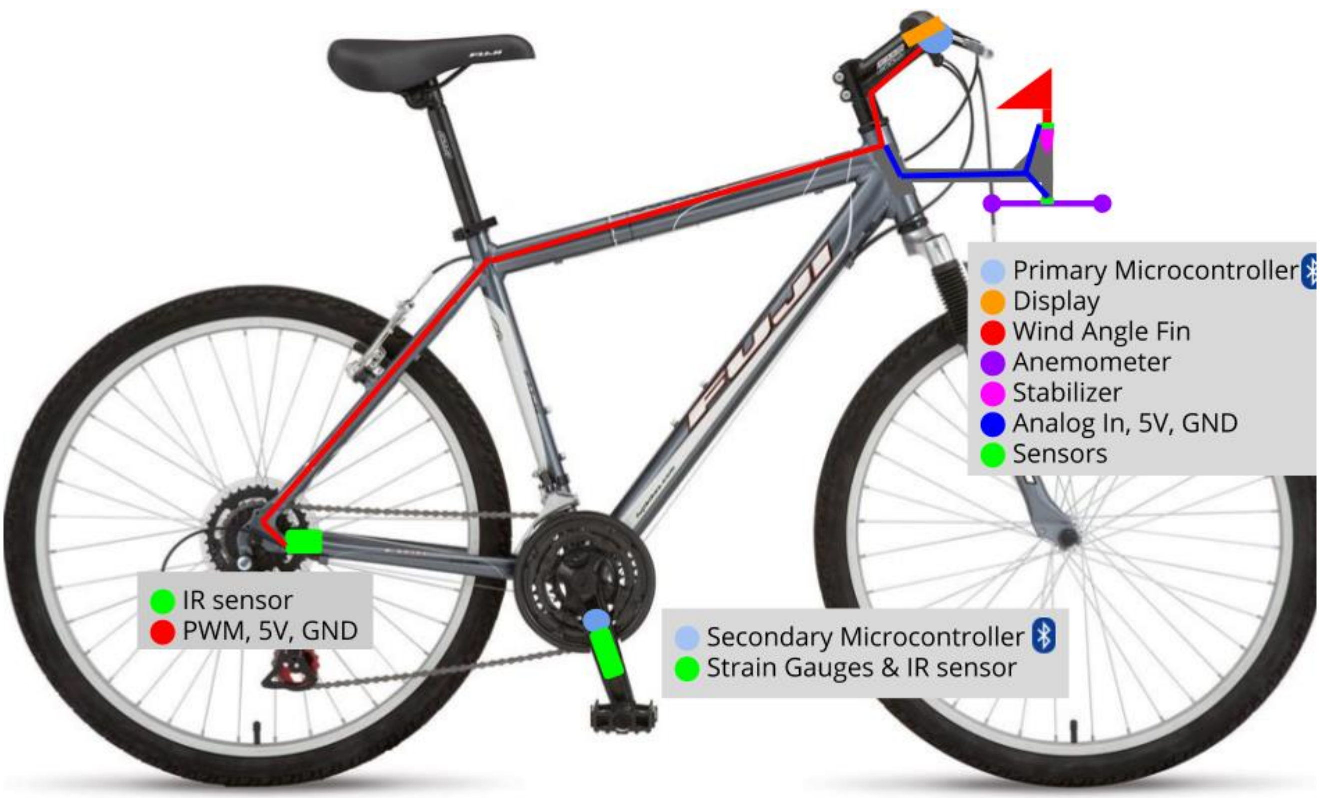

Figure 1: Basic Diagram

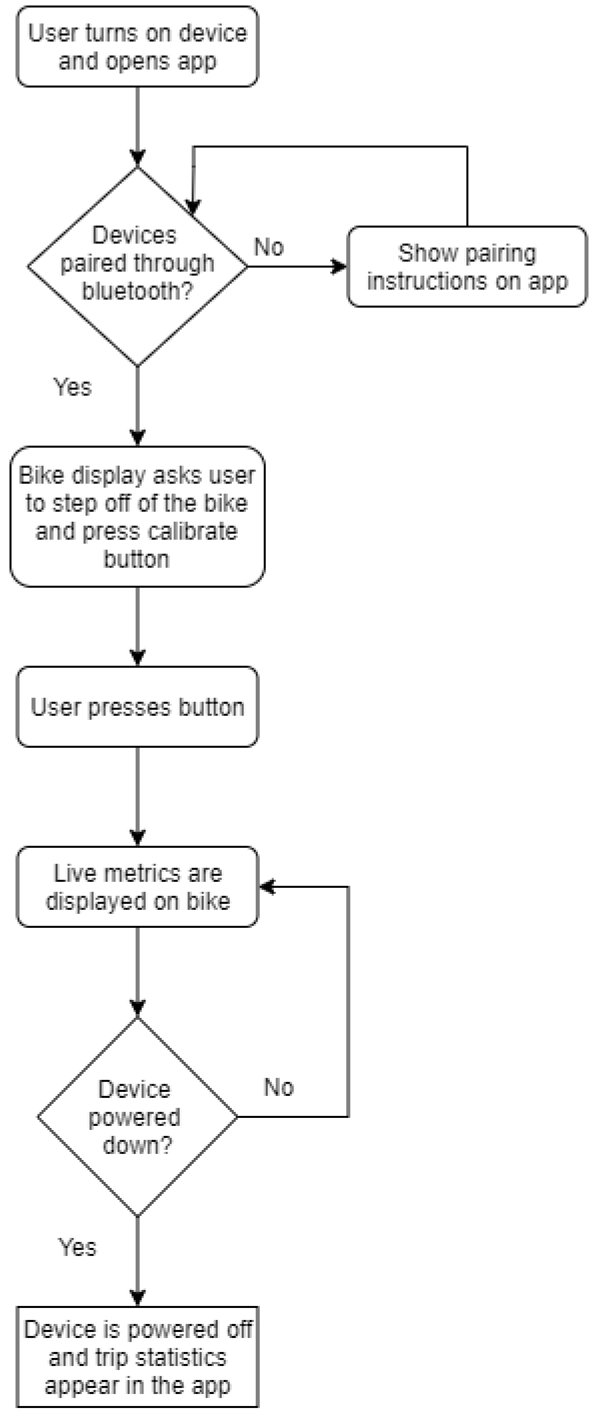

Figure 2: Flowchart

Facility Requirements:

Digital Multimeter

Computer

Computer Software

- AutoCAD
- Arduino IDE

3D Printer

Estimated Cost:

Utility	Component(s)	Quantity	Cost
Bicycle	Fuji Odessa 1.0	1	$0
DC Power	Lithium ion batteries	4	$20
Micro Controller	Arduino Nano 33 IoT	2	$40
Display	2.9" eInk Display	1	$30
	Display Breakout	1	$10
	White LED and photoresistor	1	$0
	Buttons	4	$5
Gear Ratio	IR Sensors	2	$15
Torque	Strain Gauges	4	$30
Wind Direction	Continuous Potentiometer	1	$20
	Gimbal Stabilizer (or create one)	1	$50
Wind Speed	Computer fan/DC motor	1	$5
Global Location	GPS	1	$30
Shipping			$30
Misc materials	cables, silicon, plastic, solder, etc		$100
		Total:	**$385**

Estimated Man Hours:

Proposal - 10 hours

Functional Specifications - 10 hours

Time Schedule - 10 hours

Reports - 10 hours

Presentations - 10 hours

Paper Design of Device/General Design of Device Layout - 20 hours

Ordering Parts - 5 hours

Assembly - 40 hours

Hardware Programming - 30 hours

Online Application Programming - 30 hours

Testing/Modifications - 30 hours

Final Documentation - 15 Hours

— TOTAL - 220 HOURS PER PERSON

Conclusion:

The Cycle Statistic Tool will provide an experience with more features than competing products at the same price point. In the current market, pedal shafts that measure power start around $350 [7], but they are limited to only measuring power. Currently there are no products that measure wind direction, wind speed, or gear ratio.

Knowledge from the following courses from the University of Pittsburgh-Johnstown Computer Engineering curriculum will be used to support this project: Analysis and Design of Electronic Circuits (EE0257), Advanced Programming Concepts (CS0457), Data Structures and Files (CS0458), Physics 1 (PHYS0151), Signals and Systems (EE1152), and Computer Architecture (EE1541). The following areas will need to be studied/researched for this project: App Development, Data Analysis/Management, Arduino Proficiency, and IoT networking.

This project will exercise technical skills taught in the University of Pittsburgh-Johnstown Computer Engineering curriculum, as well as giving the project team an opportunity to learn additional knowledge in specific engineering topics. Additionally, this will be an opportunity to gain hands-on experience with team-oriented tasks and project development.

References

[1] Arduino Nano IoT - https://store.arduino.cc/usa/nano-33-iot

[2] 2.9" eInk display - https://www.adafruit.com/product/4262

[3] eInk breakout - https://www.adafruit.com/product/4224

[4] Adafruit Mini GPS PA1010D - https://www.adafruit.com/product/4415

[5] Strain Gauges (gages) - https://micro-measurements.com/shear-pattern-strain-gages

[6] IR Sensors - https://store.arduino.cc/usa/grove-infrared-reflective-sensor-v1-2

[7] Relevant cycle power meters - https://road.cc/content/review/98411-stages-power-meter