University of Illinois

Senior Design Project Proposal

Acoustic Spoke Tensiometer for Bicycle Wheels

Group 14:
Xi Li
Andrius Bobbit
Sakeb Kazi

TA:
Tom Galvin

February 8, 2012

Contents

I. Introduction

1. Objectives

This project aims to design a tensiometer for bicycle wheels based on the audible frequencies emitted by the spokes when they are being struck. Currently available techniques require clamping of the spokes individually in order to determine the tension based on the physical deflection of the spokes. This method is time consuming and highly dependent on the proper calibration of the meters. This project was created to measure each bicycle spoke quickly without making individual measurements to each spoke.

The goal of this project is to measure the effective spoke length and input other parameters (butted/non-butted spokes) to calculate an optimal tension for each bicycle spoke. The device will pluck each spoke consistently so that the resonating frequency can be accurately measured and used to determine the tension in the spoke. This device would be ideal for individuals that want to make adjustments to their bicycles and bicycle repair shops that need to make measurements quickly and accurately for a multitude of customers.

Benefits

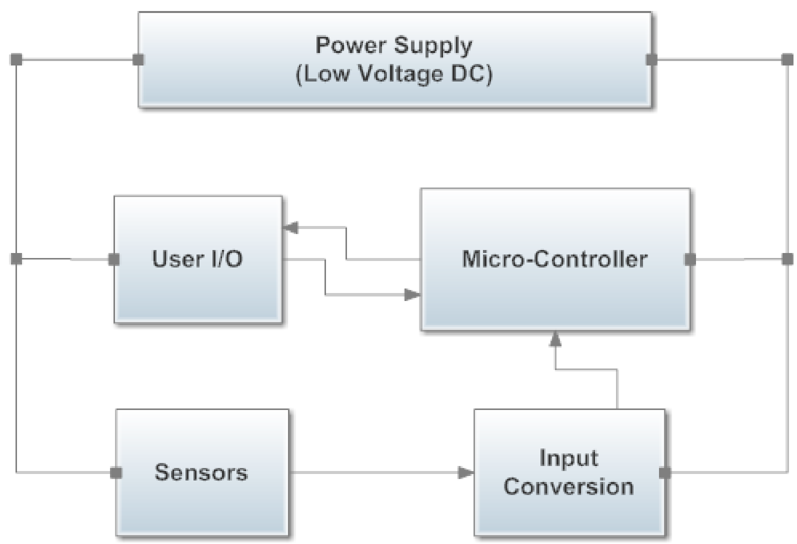
- Consistent and accurate measurements
- Calculates optimal tension of the spokes for the user
- Does not require frequent recalibration
- Quickly measure the tension for each spoke in a wheel

Features

- Convenient measurement of spoke length
- Built in plucker to ensure clean striking of the spokes
- User controlled precision range
- Real-time intuitive visual display for readings
- Stores readings for an entire wheel in memory

II. Design

1. Block Diagram



2. Block Description

Sensors: This portion of the project will read in all the measurements from the bicycle wheel. The first sensor will be an infrared sensor used to determine the length of the bicycle spokes. The infrared sensor will output a certain voltage that is proportional to how far the object is (the hub of the bicycle wheel). Our second sensor will be a microphone that takes in the frequency emitted by the resonating bicycle spokes.

Input Conversion: This block will convert the analog signals from the sensors and convert it into a digital signal that can be read by the micro-controller. It will also filter out unwanted frequencies that the microphone is picking up. The signal from the microphone and the infrared sensor will be manipulated by an analog to digital decoder which will be converted into a digital signal to be read by the micro-controller.

Micro-Controller: The micro-controller will take in the parameters from the User I/O and the Input Conversion blocks to calculate the optimal tension that each bicycle spoke should have. The micro-controller will assess the microphones readings and store each individual spokes tension into memory. The micro-controller will determine if the spoke is within the desirable tension range or not and display all results back to the User I/O.

User I/O: The User I/O will consist of 3 seven segment displays, 3 LEDs, a butted/non-butted switch, a scrolling button, an On/Off button and a keypad. The switch and keypad are needed to allow for users to enter important information about their bicycle such as number of spokes and desired accuracy. The LEDs will display whether the tension of the spoke is too low, too high, or within range. The seven segment displays will display the exact tension of each spoke on the wheel.

Power Supply: A circuit will be designed to utilize a 9 volt battery as the power source. Each block component of this project will require a slightly different voltage or current rating, which will be accommodated for through techniques such as buck/boost converters to allow for DC voltage stepping and so on. Using a 9 volt battery allows for portability in the design.

3. Performance Requirements

Infrared Sensor: Measures accurate spoke length within 10%

The tension of the spoke is primarily based on the spoke length so accurate tension requires an accurate length measurement.

Microphone: Takes measurements 60 times per second

The microphone needs to take measurements fast enough for each spoke on a spinning wheel.

Ambient Noise Tolerance: Filters unwanted noise

The micro-controller should filter out only the desired frequency of spoke and not record outside noises such as human speech or music in the background. After hitting the record button only frequencies within the optimal frequency range are detected.

User I/O: Responsive Keypad and Display with Informative LEDs

With a single press of a key the correct digit should appear on the 7-Segment Display. LEDâs indicating power and the tension relative to the perfect tension should be lit at appropriate times.

User I/O: Display Recorded Tensions

After all frequencies have been measured and the data compiled, the user should be able to recall the tension of each spoke on the 7-Segment Display in a sequential order.

Micro-Controller: Calculate Correct Tension

Receiving a frequency value the corresponding accurate tension will be displayed.

Overall Performance: Record Correct Tension

The unit should arrive at a tension within 5% of the value read by a calibrated mechanical spoke tensiometer.

III. Verification

1. Testing Procedures

IR Sensor Test: The sensor apparatus will be setup to measure the target board at a distance of 20 and 25 and 30 cm. It will be hooked up to a multimeter which will display the voltage output values. If working correctly, a voltage within 5% of the voltage indicated on the datasheet will be displayed. A non-working sensor will show no correlation between the distance and the voltage.

Microphone Response Test: The output of the microphone will be hooked up to an oscilloscope displaying the voltage versus time. A short audio file, with rapid spikes in the pitch, will be played on a computer speaker in front of the microphone. If operating correctly a graph with discernible spikes will be displayed. A graph that does not resemble the noise of the audio file indicates a failure.

Ambient Noise Test: The output of the audio filter will be hooked up to an oscilloscope displaying the voltage and the mic will be placed in front of a speaker that plays a range of frequencies from 300-900 Hz. Meanwhile, background speakers will play an audio file of people talking. Correct operation will yield a graph barely affect by the talking with a highly noisy graph being a failure.

Keypad Test: LEDs will be connected to each output pin of the keypad. When a button is pressed the appropriate LED will light up. If the keypad is not on or not working no lights should turn on.

7-Segment and LED Display Test: A preset program in the micro-controller will be run set to display certain values on the 7-Segment Displays as well as light up certain LEDs. If working, correct values and lights will turn on. If not working nothing should be displayed.

Tension Display Test: A preset list of tensions will be loaded on the micro-controller which will be output to the displays. If working, the user should be able to scroll through the list using the keypad. A failure will consist of an non-ordered list or no keypad response.

Tension Calculation Test: A frequency (or voltage corresponding to a frequency based off of our microphone) will be supplied to the input of the micro-controller. Correct operation yields in a tension corresponding to that frequency based off the algorithm used. Any other value is a failure.

Overall Performance Test: When setup with the proper user input the apparatus should record the frequency of each spoke that is struck and the micro-controller will convert that to a tension value. These tensions will be stored in memory and accessible on the display using the keypad. The user will also be be informed if the tension is less or greater than the desired tension. A tension value outside the range of $\pm$ of the value acquired using a mechanical tensiometer will be considered a failure.

2. Tolerance Analysis

It is important to determine the range of frequencies the microphone can detect. The microphone should be able to detect frequencies as low as 370 Hz and as high as 900 Hz. The microphone should not pick up ambient noise or other unwanted frequencies such as previously resonating spokes. The microphone should also have a quick response rate to allow for the measurement of each spoke as the wheel spins.

We will introduce low frequencies to the microphone with a signal generator connected to a speaker and see if it can detect the signal to establish the lower bound of the frequency range. We will repeat the process with higher frequencies to determine the upper bound of the frequency range. In order to determine the response rate of the microphone, we will vary the signal on the signal generator and measure the time it takes for the microphone to output a different voltage. Using an IF Filter will prevent other resonating frequencies from other spokes to interfere in proper signal acquisition. The final report will indicate the range of acceptable frequencies and the response rate at which our microphone can operate on.

IV. Cost and Schedule

1. Cost Analysis

Labor

Employee	Labor
Xi Li	12 hrs/week x 2.5 x 12 weeks x $30/hr ($60k Salary) = $10,800
Andrius Bobbit	12 hrs/week x 2.5 x 12 weeks x $30/hr ($60k Salary) = $10,800
Sakeb Kazi	12 hrs/week x 2.5 x 12 weeks x $30/hr ($60k Salary) = $10,800
TOTAL	($10,800/person x 3 persons) = $32,400

Parts

Part	Quantity	Provider	Cost
IR Detector	1		$15.00
Microcontroller	1	Estimated Cost	$30.00
A/D Convertor	1		$5.00
Mic	1		$8.00
7-Segment Displays	2		$2.00 (4 Digits)
LED Diodes	3		$0.35 each
Key Pad	1		$4.00
Push Buttons	1		$5.00 (2x2)

NOTE: Costs found from Sparkfun.com

Total $72.05 + $32,400 = **$32,472.05**

2. Schedule

Deadline	**Week**	**Responsibilities**	**Team Members**
	2/3	Background research for proposal	Xi
8-Feb *Proposal*	4	Ensure that the proposal is correctly formatted and completed.	Andrius
13-Feb *Design Review Signup*	5	Research appropriate parts for project	Sakeb
20-Feb *Design Review*	6	Complete Design Review	Andrius
	7	-Order Parts -Assemble Power Supply	Xi
	8	-Assemble Sensors -Tolerance Analysis	Sakeb
14-Mar *Individual Progress Report*	9	Assemble Input Conversion	Andrius
	10	Assemble User I/O	Sakeb
26-Mar *Mock Up Demo Signup*	11	-Assemble Microcontroller -Ensure proper communication between entities	Andrius
2-Apr *Mock Up Presentation Signup* *Mock Up Demo*	12	Debugging	Xi
	13	Calibration	Andrius
	14	Testing	Sakeb
23-Apr *Demo*	15	Presentation Preparation	Xi
30-Apr *Presentation* *Final Paper*	16	Complete Final Paper	Andrius