Auto-Truing Bicycle Stand

03/15/2013
ME 405-03
Dr. Ridgely

Trevor Jones
Hamilton Little
Sean Green

Section 1: Introduction

1.1 Background Information

Under normal operating condition, bicycles experience heavy static and cyclic loading. In addition to the rider's weight, the bicycle's wheels experience an equal and opposite reactionary force from the ground. Over time this loading causes the bicycle rim to deform from its axis of symmetry, decreasing performance and eventually rendering the bike un-useable unless repaired.

Riders address this problem by tightening or loosening the spokes on the bent wheel, forcing the rim to move towards a common axis of symmetry. Fixing a bent wheel requires both truing and dishing. While dishing is relative simple, truing is tedious and risks error that can completely ruin the wheel. Publically available truing methods and stands do exist; however, professional level truing still requires training and experience the average enthusiasts does not possess.

1.2 Target Market

The auto-truing bicycle stand aims to reduce the time and effort required to properly true a bicycle wheel. Avid cyclists generally have the experience and equipment required for truing while professionals have crew members to perform the task. Therefore this machine is marketed towards fledgling enthusiasts and newly competitive cyclists who do not yet have the experience to properly true their wheels. It is also designed with bicycle repair shops in mind, reducing the overall time and skill level needed for a worker to repair a bent rim.

Section 2: Specifications

2.1 Customer and Engineering Requirements

The overall goal of the truing stand is to reduce the amount of time it takes a non-professional rider to true their wheel. Thus customer requirements and the subsequent engineering specifications center around the fact that the auto-truing stand must be fast and accommodate a fairly wide variety of wheels, as shown in Table 1.

Table 1 - List of Customer Requirements and Engineering Specifications

Customer Requirement	Engineering Requirement	Maximum	Minimum
Reduce overall truing time	Total Truing Time	5 Minutes	N/A
Determine spokes that need tightening			
Accept most common bike wheels	Acceptable Wheel Diameter	29 inches	22 inches
	Acceptable Spoke Number	40	N/A
Prevent further damage to bicycle rim	Initial from True Axis	5mm	N/A
Useable with quick-release and standard axles	Final Displacement from True Axis	.1mm	N/A

2.2 Expected Results from Auto-Truing Bicycle Stand

To meet these requirements, the auto-truing bicycle stand measures the displacement from multiple points along the rim of an out-of-true bicycle wheel. These measurements are used to determine the degree to which the wheel is out of true.

Using these values, the auto-truing stand calculates the maximum and minimum allowable displacements and creates a tolerance band for the bicycle wheel measurements. It then alerts the user which spokes need to be tightened or loosened to bring the rim's displacement into this tolerance band. Shown below are the expected results before and after use of the auto-truing stand.

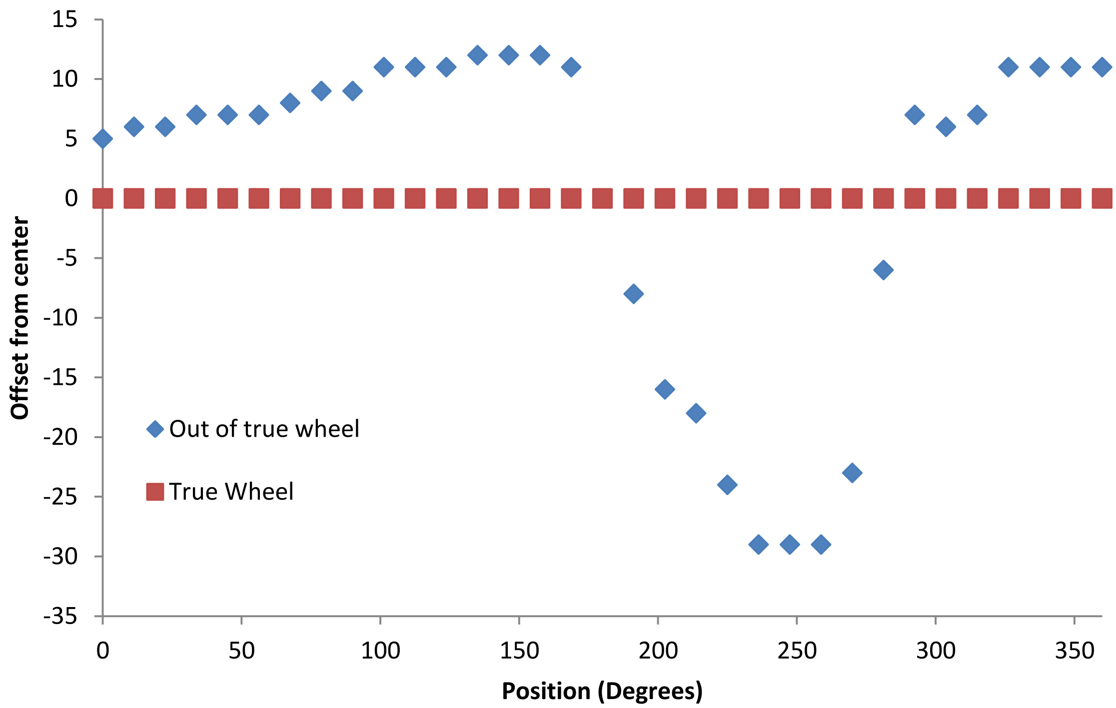

Figure 1 – Expected rim displacement of generic out of true wheel and true wheel

Figure 1 displays the expected displacement measurements taken about the rim of a perfectly trued bicycle wheel and a bicycle wheel that is out of true. While the figure does not include units, it reflects a general trend that the auto-truing bicycle stand will be able to measure. More displacement measurements reflect increased precision and will lead to better results.

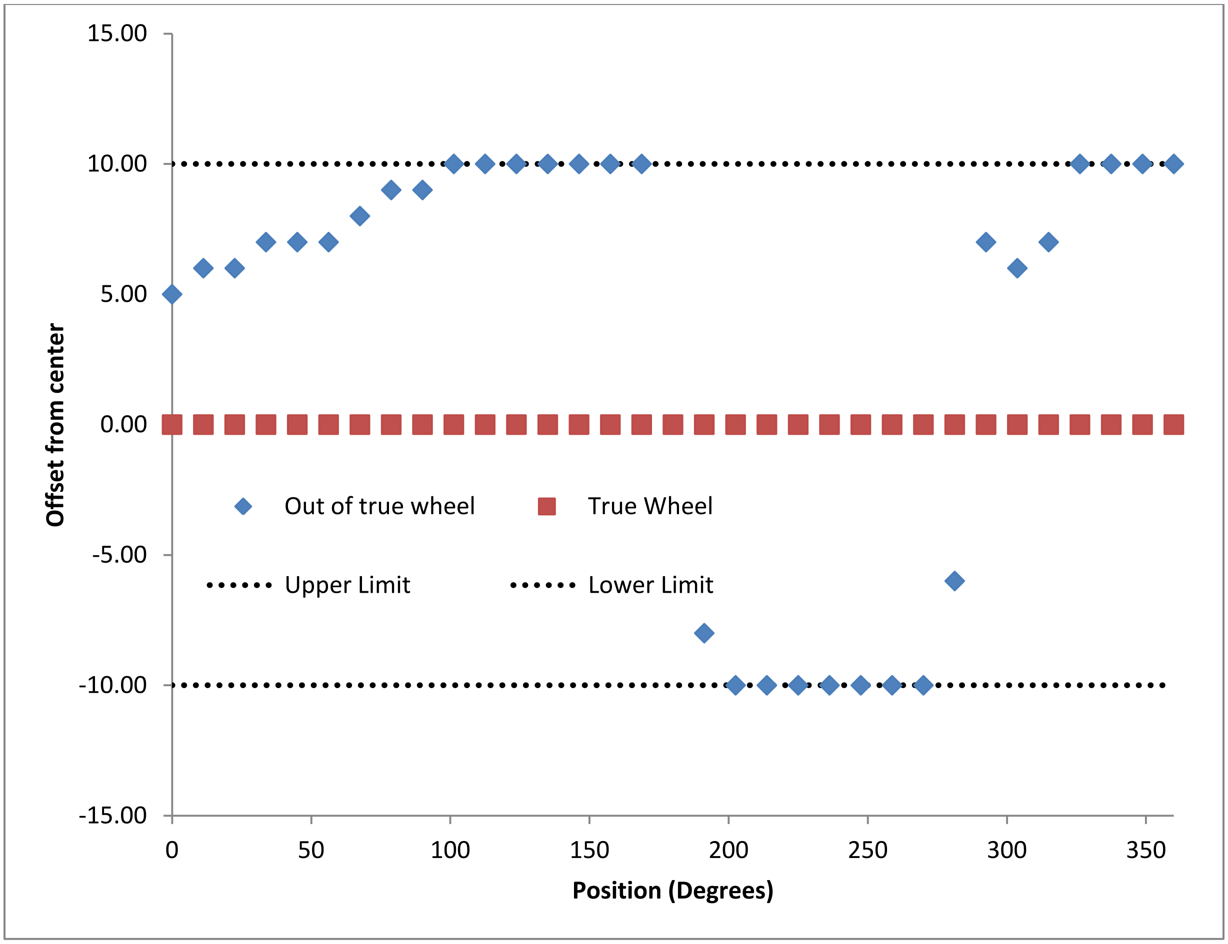

Figure 2 – Expected rim displacement of perfectly true wheel and out of true wheel after tuning using auto-truing bicycle stand.

Figure 2 displays the expected displacement measurements of the previous bicycle rim after using the auto-truing stand. Data points previously outside the upper and lower bounds have been adjusted to meet tolerance requirements.

Note that in this model, measurements originally within tolerance are not affected by wheel adjustments. In reality, any adjustments to the wheel affect the entire rim's displacement. However, this displacement is difficult to represent and predict. The auto-truing stand takes this into account by re-measuring the rim displacement after each adjustment, ensuring that individual measurement is out of tolerance.

Section 3: Design Development

Following a general system layout, hardware was selected to meet a specific need of the design. Once a piece of hardware was selected, software was designed to implement that component into the system. These drivers were then integrated into the tasks diagrams to show the relationships between each hardware object and their corresponding place in software.

3.1 Hardware Design

Hardware design consisted primarily of five different components: a bike stand, friction roller mounted on a motor, encoder, follower-potentiometer assembly, and the laser-phototransistor circuit.

Table 2 provides a cursory overview of the function of each of the major hardware components of the system. The mechanical assembly shown below displays the physical integration of the hardware into the overall system.

Table 2 - List of Primary Hardware Components

Part	Description	Qty.
Bike Stand	Rigid fixture to lock wheel in place and allow rotation	1
24V DC Motor	Friction roller mounted on motor to spin wheel	1
Encoder	Encoder to determine direction of wheel rotation	1
Linear Slide Potentiometer	Variable resistor that outputs voltage proportional to displacement	1
Follower wheel	Locks onto wheel as it spins and translates when wheel moves out of true	1
Laser and phototransistor	Interruption of beam triggered an external interrupt to say a spoke has passed	1

Figure 3 - Rendering of mechanical assembly of truing stand

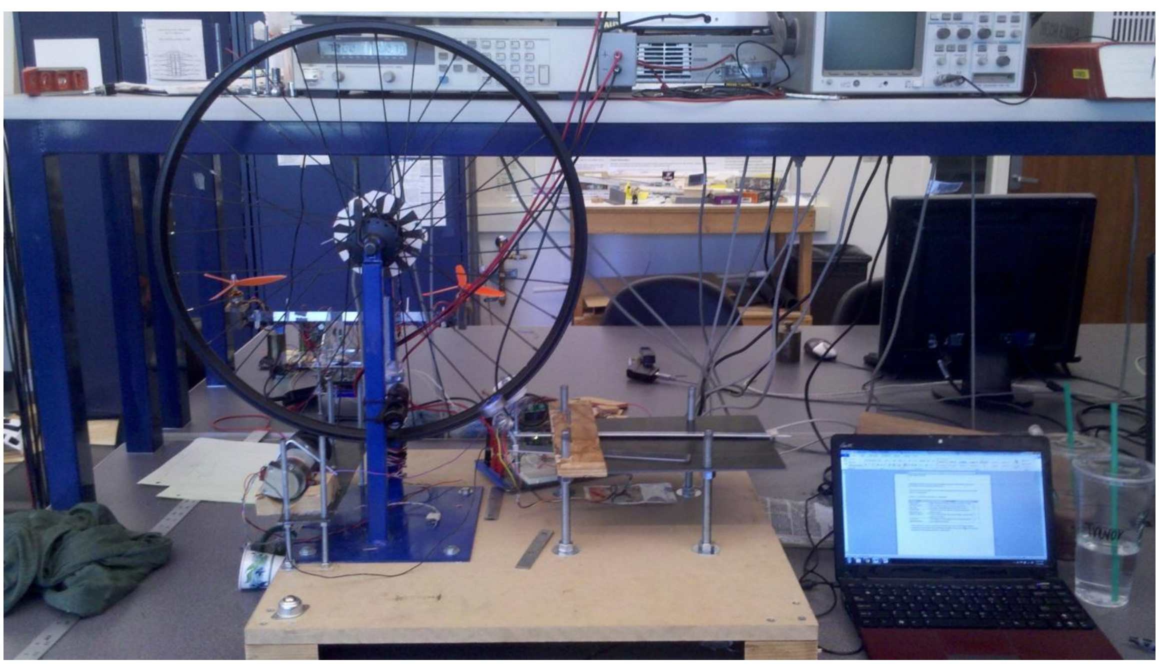

Figure 4 – Actual System

3.1.1 Follower system / encoder / potentiometer

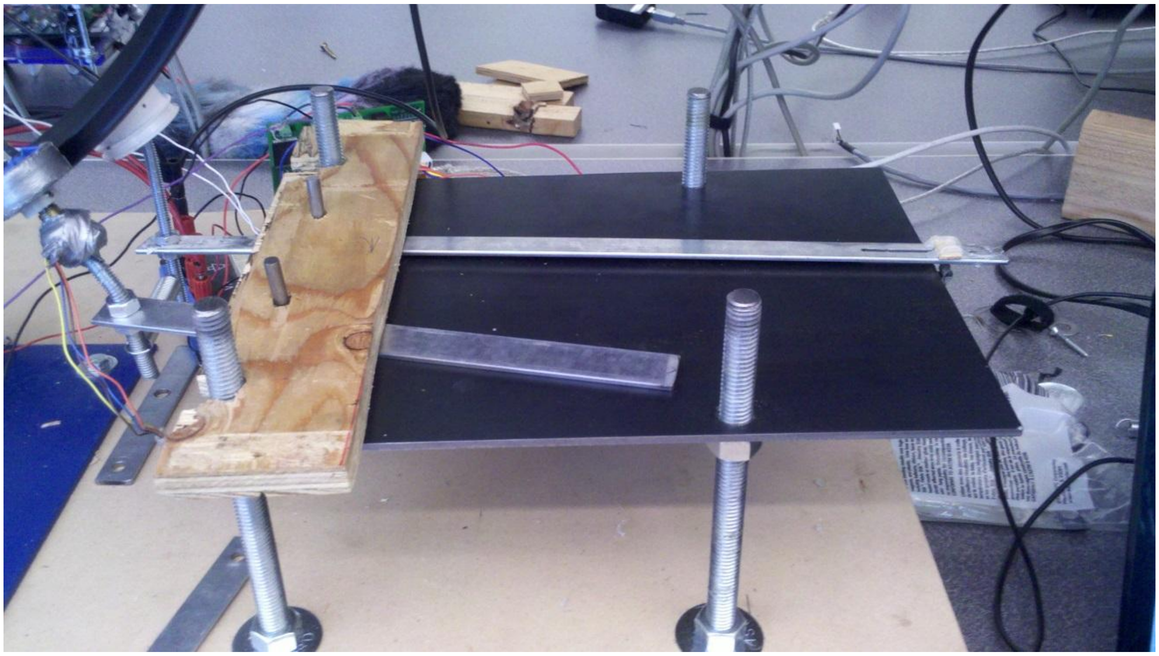

Figure 5 – Follower system final assembly. Note that the slide potentiometer is not visible in this picture

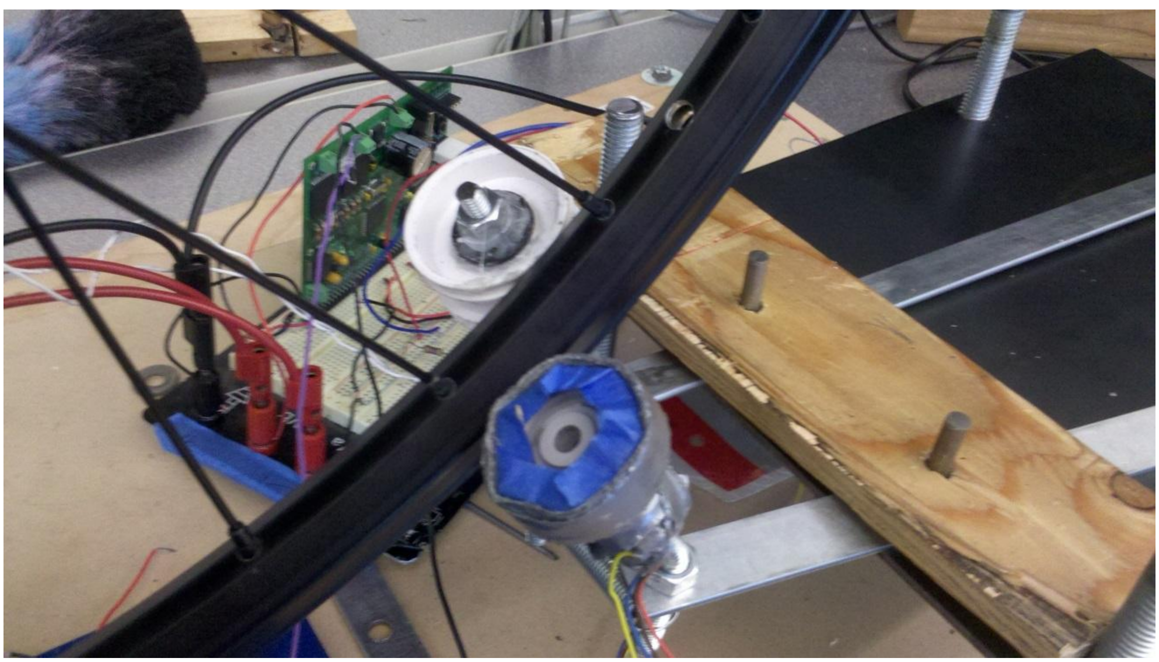

Figure 7 – Close-up of wheel followers. Note that the blue follower is an encoder for measuring the direction of rotation.

As shown in **Figure 5**, the follower system mounts two wheels on either side of the bicycle rim. A spring provides the tension force required to keep the followers in constant contact with the rim. Wheels mounted at the bicycle rim contact reduce friction forces that oppose the bike's rotation (**Figure 6**). Note that the blue wheel is actually the encoder used to determine the bicycle wheel's direction of rotation.

Overall, the follower system moves laterally as the rim displaces from the true axis. The bars shown then provide a 4:1 displacement to the slide potentiometer mounted at the end of the plate (**Figure 7)**. This physical gain increases the displacement experienced by the slide potentiometer for a given rim displacement, providing higher resolution measurements for the system.

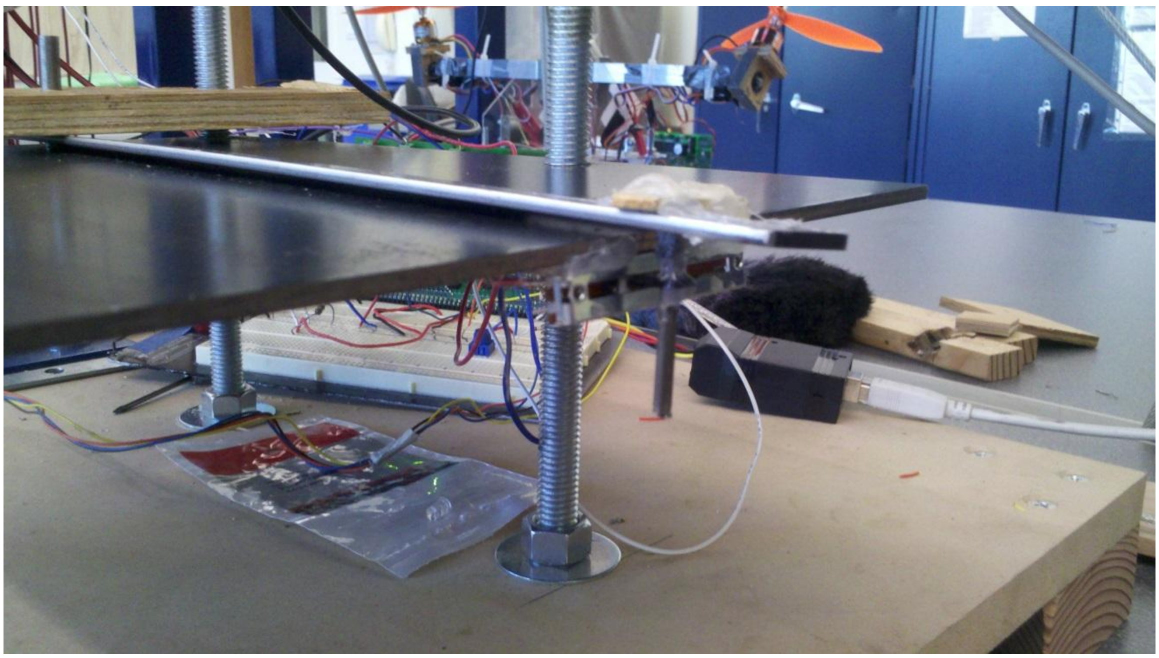

Figure 7 – Close-up of slide potentiometer at the end of follower-potentiometer assembly

3.1.2 Laser –Phototransistor

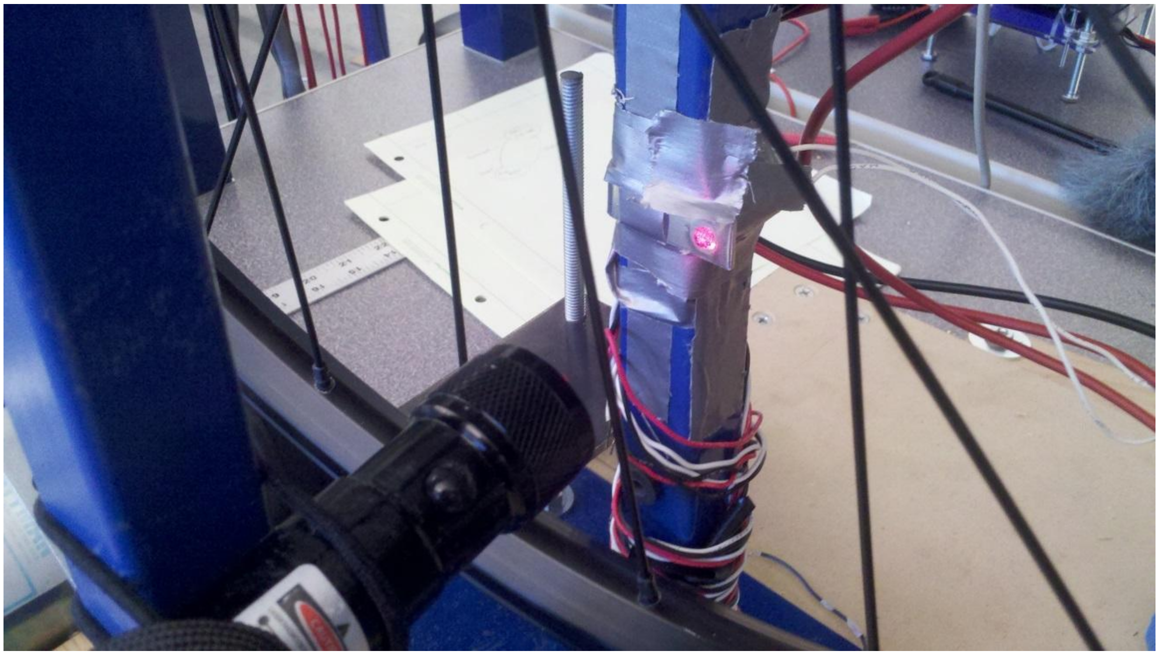

Figure 8 – Laser-Phototransistor set-up. Rotating spokes interrupt the laser beam, creating a readable change in voltage measureable from the ME 405 board.

This setup determines whether a spoke has passed through a certain point in the system. The laser constantly points at a photo-transistor which outputs a voltage based on the amount of light it receives. When a spoke interrupts the laser, the voltage output from the photo-transistor dips below the normal level. This action triggers a hardware interrupt request on the microcontroller. Based on the wheel direction read from the encoder, the spoke count is either incremented or decremented.

3.1.3 Friction Roller – 24V DC Motor

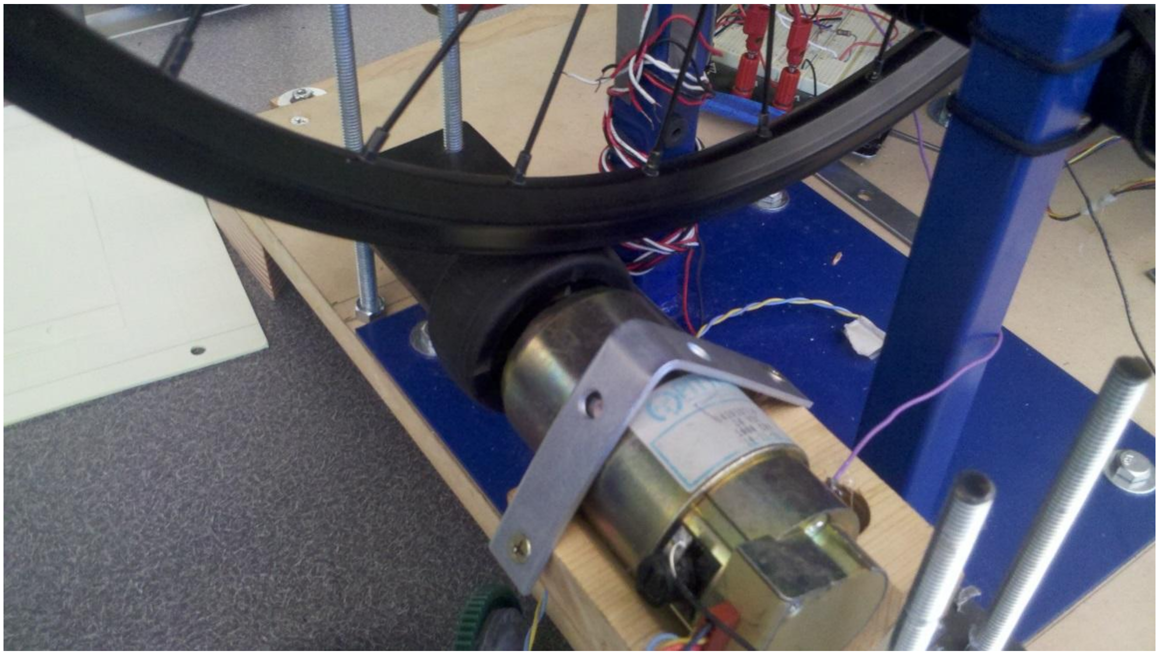

Figure 9 – Friction roller used to change wheel position. Nuts underneath (not picture) adjust the height of the mounting platform to accommodate larger wheels.

The friction roller system component provides position control for the bicycle wheel during measurements and tuning. The rubber wheel mounted to the shaft of the 24V DC electric motor allows the system to grip bicycle wheels with or without tires. In addition, the height of the entire platform is adjustable to accommodate smaller or larger bicycle wheels.

3.1.4 Miscellaneous Mechanical Design

The wheel is held in place by a standard bike wheel holder, which allows the wheel to freely rotate about its center axis. Steel mounting rods prevent deflection and provide stiff support to the steel mounting plates for the potentiometer and friction roller assemblies. This prevents false readings on the potentiometer while ensuring the friction roller maintains contact with the wheel at all times. Finally, the entire setup is bolded to a rigid baseboard for extra support.

3.2 Software Design

The software drivers are written to directly correlate with each of the physical subsystems. They are designed with modularity in mind and allow each component to communication with each other as needed to rotate and track the wheel while measuring and recording its displacement at the given positions.

3.2.1 User interface

The user interface task communicates the wheel's location and offset after each round of measurements and prompts the user to either loosen or tighten a specific spoke. After tightening a spoke, the user provides input to signal they are finished. The UI then notifies the mastermind task to re-measure the wheel. Based on the results, the UI will either prompt the user to adjust another spoke or notify them that the wheel is within tolerance.

During measurement, the UI also displays the ADC readings, the average of these readings and the calculated offsets for a given spoke position. This allows the user to follow the displacement trend and determine whether the wheel is salvageable.

Future implementations of the UI will prompt the user for the number of spokes on the wheel being measured and on which side of the wheel the first spoke attaches to the rim. Taken together, this will allow the truing algorithm to accurately inform the user whether a spoke needs to be tightened or loosened for truing purposes (see Appendix A).

3.2.2 Spoke Counter

```
ISR(INT4_vect) {

      // if wheel direction is true, increment. Else Decrement
      if (wheel->get_direction()){
            count++;
      }else{
            count--;
      }
}
```

To measure whether a spoke has been passed, a task for incrementing or decrementing position based on which spoke a user is at, has been deployed. Using the encoder that rotated alongside the wheel in the follower setup, a shared wheel direction variable is updated. When a spoke breaks the laser beam, the photo-transistor voltage changes, triggering a hardware interrupt request. This interrupt vector reads wheel direction from the encoder and count, is either incremented or decremented appropriately.

```
spoke_counter::update()
```

To prevent errors, a separate method in the `spoke_counter` class updates the shared variable `spoke_count` separately from the variable `count`. This prevents errors from other methods accidentally writing directly to `count` and affecting the system.

3.2.3 Position Controller

To effectively navigate to a specific spoke on the wheel or rotate the wheel for measurement, PI control is necessary. The controller reads the `desired_spoke` that's updated by the mastermind task and reads the current value of shared variable `spoke_count`. The position controller computes the error from these two values and runs it through proportional and integral gains. The result of these computations then provides an actuation voltage to the 24V DC motor to update the wheel's position.

```
void pos_controller::update() {
    int8_t pos_act = spoke_count;
    int8_t pos_des = desired_spoke;
    int32_t KI_control = 0, KP_control = 0;
    int16_t control = 0;
    int8_t e = pos_des - pos_act;

    // motor braking if we are at desired spoke
    if(!e) {
        motor->set_power(0);
    }
    else {
        // Integrator control, with limiting
        if(ABS(e) <= limit) {

            // integrator clamping
            if(esum + e > 127) {
                esum = 127;
            } else if (esum + e < -128) {
                esum = -128;
            } else {
                esum += e;
            }
            KI_control = esum * FF_gain * KI;
        }
        else {
            esum = 0;
        }
        // Proportional control
        KP_control = e * FF_gain * KP;

        // Control saturation (prevents actuation signal overflow)
        if(KP_control + KI_control > 32000) {
            control = 32000;
        } else if(KP_control + KI_control < -32000) {
            control = -32000;
        } else {
            control = KP_control + KI_control;
        }

        // update the motor actuation signal
        motor->set_power(control);
    }
}
```

3.2.3 Mastermind

As the bicycle wheel rotates, the follower system displaces the linear potentiometer, which outputs voltages that are converted by an ADC. The displacement at each spoke is placed into an array with each index corresponding to the spoke position (ie. 0 to 31 for a 32 spoke wheel). An average of all displacements are taken to define the center of the wheel, and from that, offsets are calculated at each spoke. The greatest offset is calculated, and a desired position is sent to the PI controller, which subsequently navigates to the given spoke so that a user can tighten or loosen it appropriately.

```
int16_t *mastermind::con_to_offs(int16_t meas[], int16_t avg) {
    uint8_t ndx;

    // for each element in the array, convert it from an absolute
measurement to an
    // offset by subtracting the avg value passed in.
    for(ndx = 0; ndx < max_spokes; ++ndx) {
        meas[ndx] = (int16_t)(meas[ndx] - avg);
    }
    return meas;
}
```

Section 4: Results

4.1 Working System Overview

Overall, the test results follow the projected results and match the measured trends of the test bicycle wheel. The Automatic Truing Stand accurately measures the displacement trends of the rim from center axis, calculates the truing axis, positions the wheel to the appropriate location and prompts the user to adjust the spoke.

Once a user has tightened or loosened the suggested spoke, the system correctly re-measures the wheel and again prompts the user to tighten or loosen the worst spoke. This continues until all offsets are within a predefined tolerance at which point the program congratulates the user and ends.

Because the stand constantly tracks wheel position, the stand will return to the correct spoke if the user accidently jostles wheel during tuning. Furthermore, the system has demonstrated precision by returning to the same spoke repeatedly if no wheel adjustments are made.

4.2 Testing Results

As shown in **Figure** 10, the trend observed in the physical measurements of the wheel's displacement match the data trends collected by the truing stand. The physical test results are scaled slightly to allow the curves to overlap. This scaling is necessary because of deflection imprecise construction of the pivot rod that actuates the slide potentiometer. It currently does not provide the expected 4:1 scaling ratio and instead closer to 4.5:1 scaling.

Adjusting for this error, the two curves mimic each other and show that the truing stand correctly measures the out of true wheel's offset. Both the truing stand and the physical test show the same spoke as being most out of true and indicate that there is a minor bump in the wheel (a fairly common case for out of true wheels).

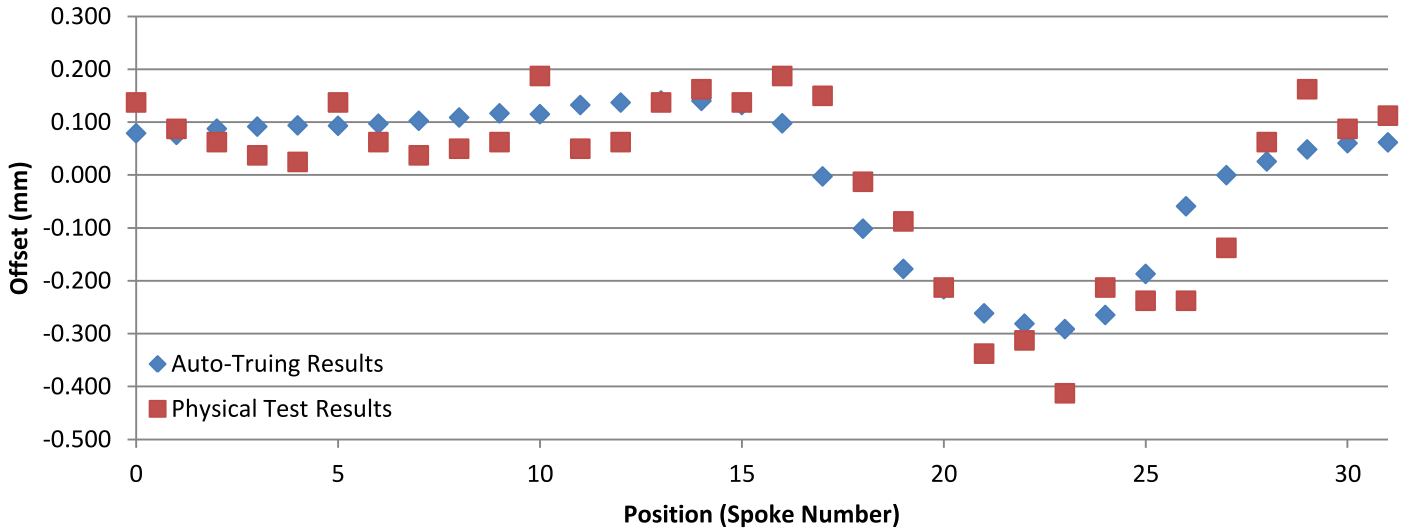

Figure 10 - Comparison of displacement trends from physical testing and auto truing stand output.

4.3 Current Problems

While the system works well, issues with manufacturing currently prevent the physical system from giving completely accurate results. Wobble in the follower wheels and friction between the pivot bolts and actuating rods prevent the follower system from reading the true displacement of the rim. In addition, the follower displacement also torques the actuating. This prevents the linear potentiometer from achieving the desired accuracy for high-level bicycle truing.

4.4 Future Improvements

Future iterations will most likely use bearings in the place of pivot rods and a rotary potentiometer in place of the linear potentiometer. In this case, the physical reduction would come from a gear train instead of a rigid bar. The benefits of reducing system size and eliminating the need for rigid actuating bars greatly outweigh the redesign needed for this proposed system. While gear backlash might be an issue, proper design or the use of worm gears can alleviate possible error.

On the software side, functionality would be added to refine the truing algorithm. The current system relies on the user to read the spoke displacement and adjust the wheel appropriately. Future forms will measure the current position and inform the user directly to tighten or loosen the given spoke.

Section 5: Conclusion

With the exception of the aforementioned issues, this project met most of the engineering requirements stated in the proposal. The system accurately measures and outputs the displacement trends of an out of true rim. It correctly navigates to the worst spoke and informs the user to adjust appropriately, then waits for user input before re-measuring and repeating until the wheel is within tolerance.

Overall, we consider this project a tremendous success and are potentially looking to continue work in the upcoming months in hopes of making it either commercially viable or publicly available.

Appendix C: Code

The full source code is hosted at:

http://wind.calpoly.edu/ME405/mecha01/AutoTruingStand/

The documentation has been sent along with this report. Alternatively, it can be downloaded from the above link (it is in /doc).