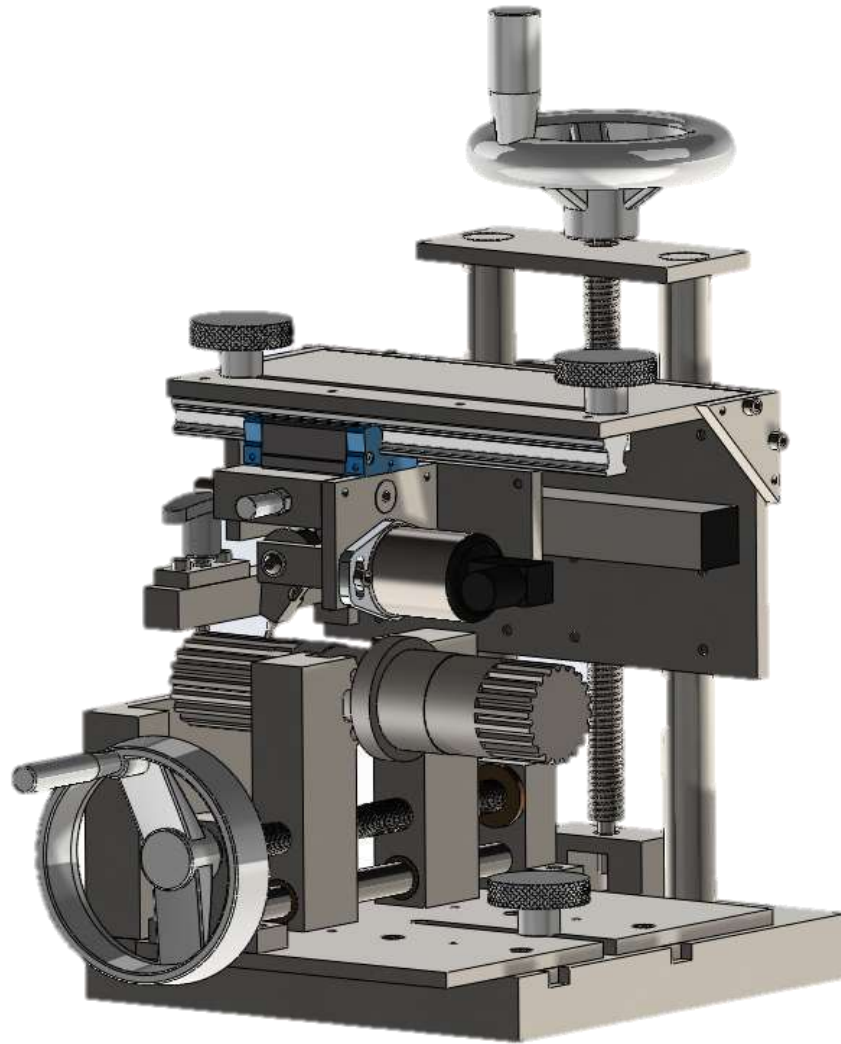




Shaft Twist Detection Tool



Bryce Rickard
Thomas McAtee



Agenda



- Overview
- Requirements
- Considered Designs
- Initial Design
- Final Design
- Schematics
- Algorithm
- Testing
- Results
- Conclusion



Overview

- Power Take-Off (PTO) Shafts
 - Part of the differential
 - Can twist and break if overloaded
 - Unusable/unreliable when twisted



└ Splined shaft in U.S. Army combat vehicle



↑ Twist

└ Failed fuse section





Overview (Cont.)

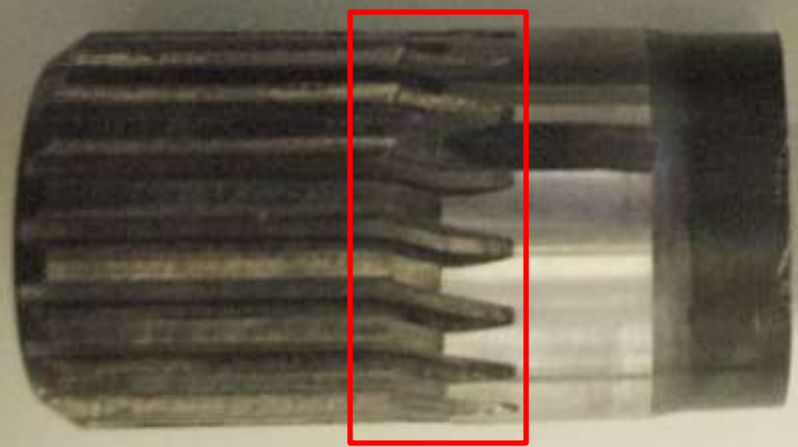


- Inspection
 - Visual inspection on twisted shafts
 - Unreliable and timely
 - Other techniques needed
- Solution
 - Develop tool to measure twist
 - Eliminates ambiguity
 - Saves time and money



Good Shaft

Bad Shaft





Requirements

- Go/ No Go Indicator
 - Red/green LED
- Calibrated
 - Microcontroller
- Fast Measurements
 - Less than five minutes
- Detect \leq One Degree
 - Any less is undetectable by eye
 - Precision to tenth of a degree
- Portable
- Cost Efficient
 - Tool cost and life
 - Needs to save money



Requirements (Cont.)



- Very Durable
 - Machine shop environment
 - Oil/water resistant
 - Shock absorbent
 - May need lights
 - Tactical environments
 - Approx -40 Fahrenheit to 126 Fahrenheit
 - Shock absorbent
 - Water resistant
 - Lightweight



Requirements (Cont.)



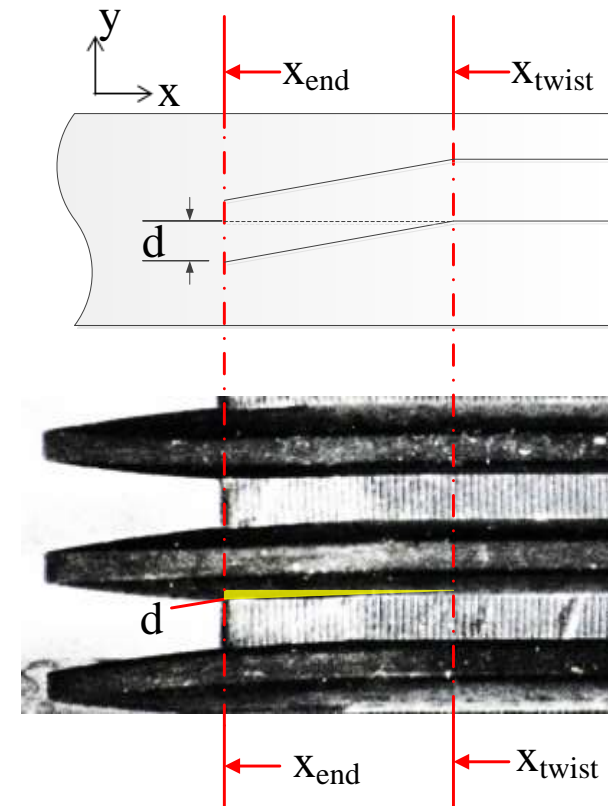
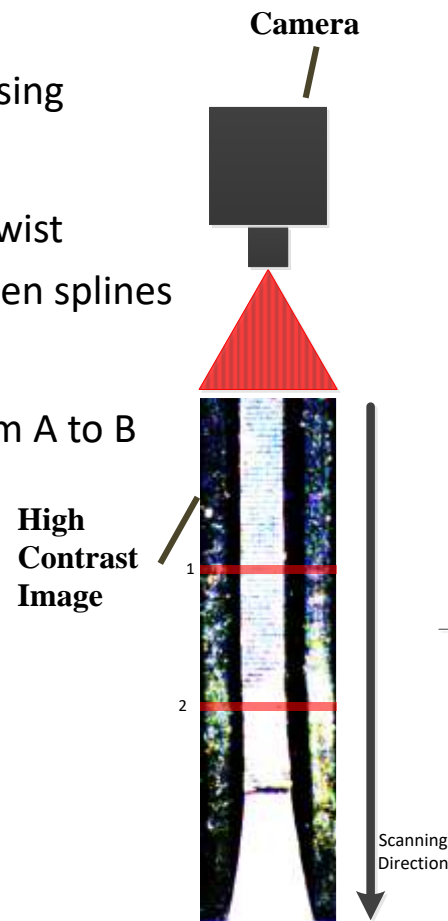
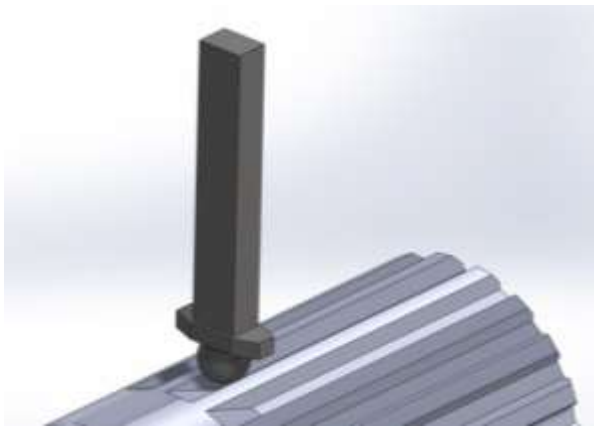
- Following are optional:
 - Digital display
 - Adjustable for different diameters



Considered Designs

- Designs Include:

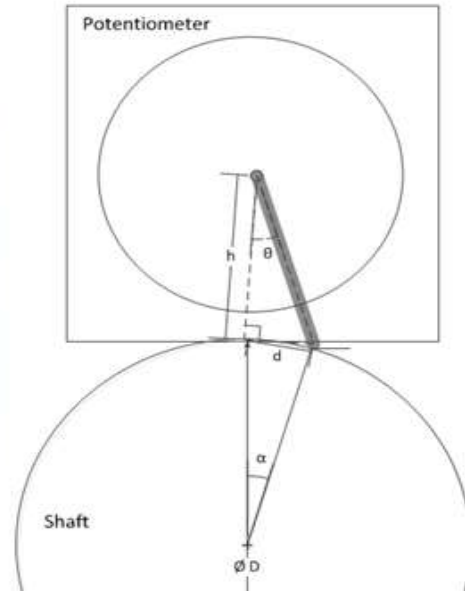
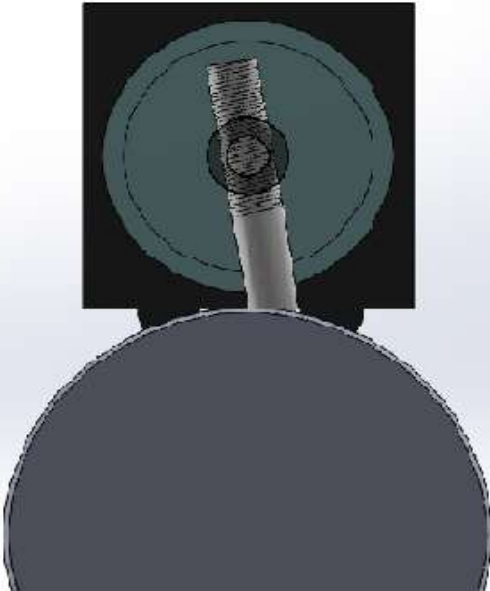
- Optical
 - Camera with Image processing
- Ball Plunger
 - Physical measurement of twist
 - Measures the width between splines
- Planar Displacement
 - Measure displacement from A to B



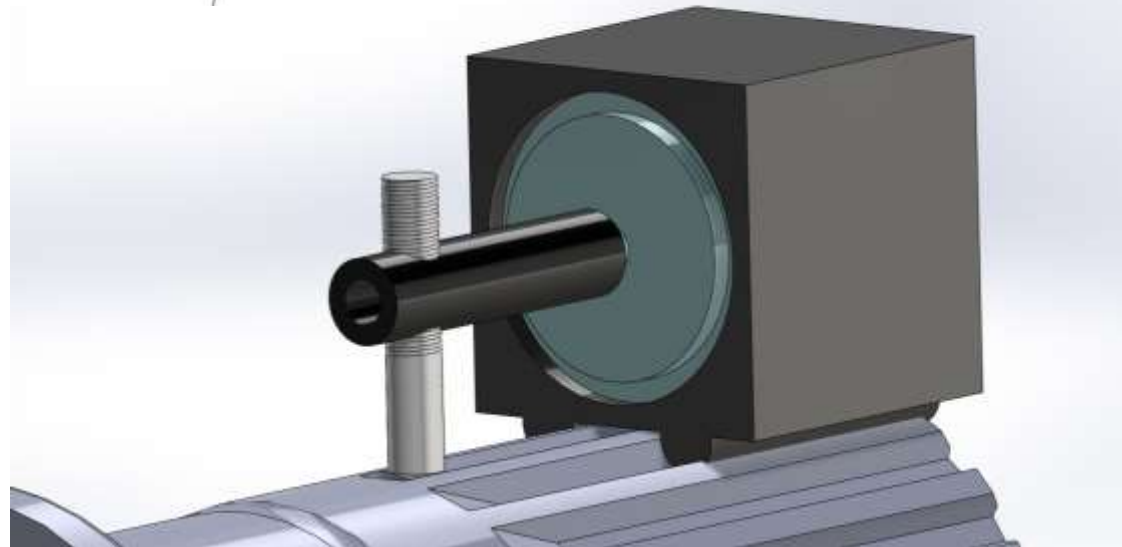


Initial Design

- Rotary/Linear Sensor Combo
 - Takes angle sample respect to position
 - Uses algorithm
 - Thousands of samples

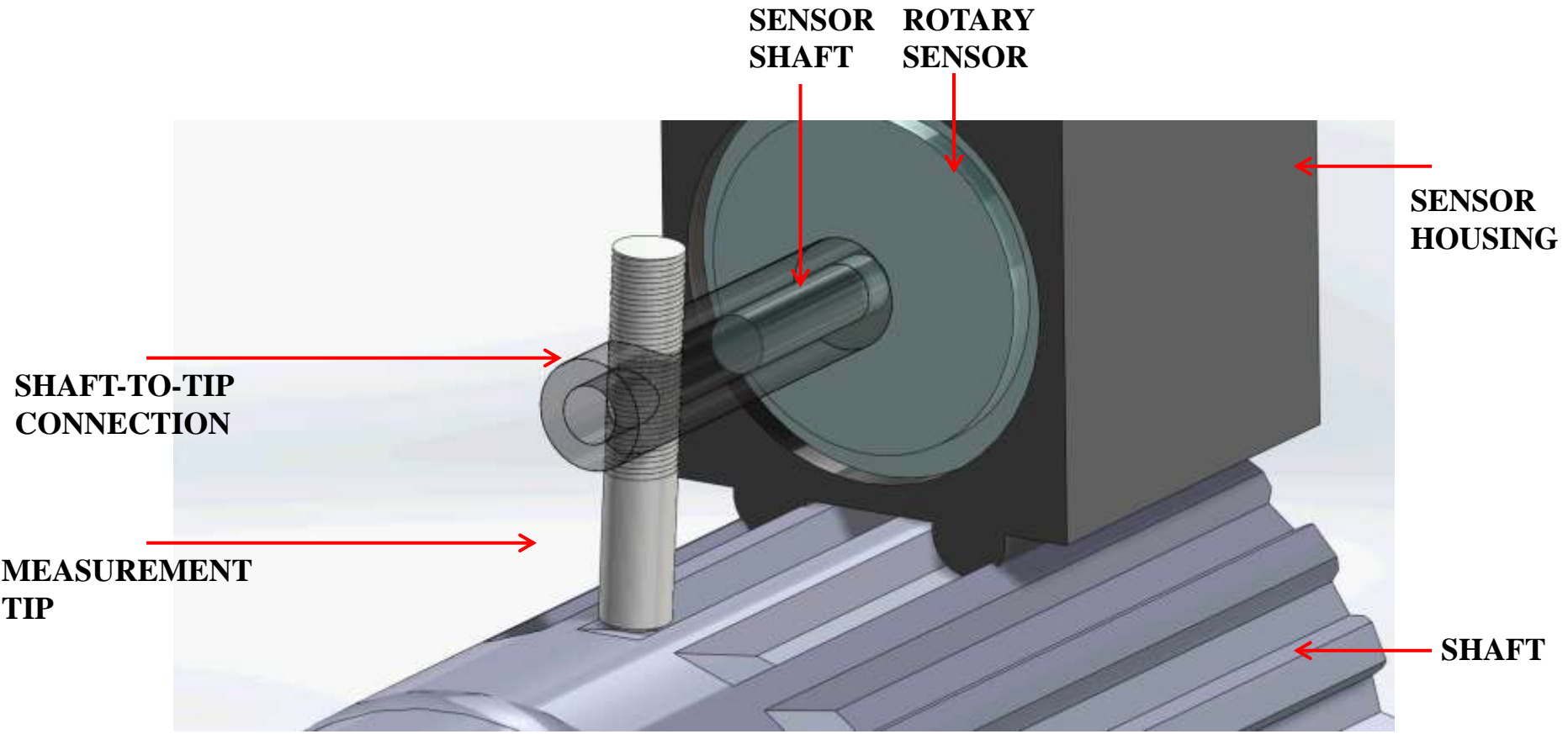


$$d = h_{tip} \tan \theta$$





Initial Design(Cont.)





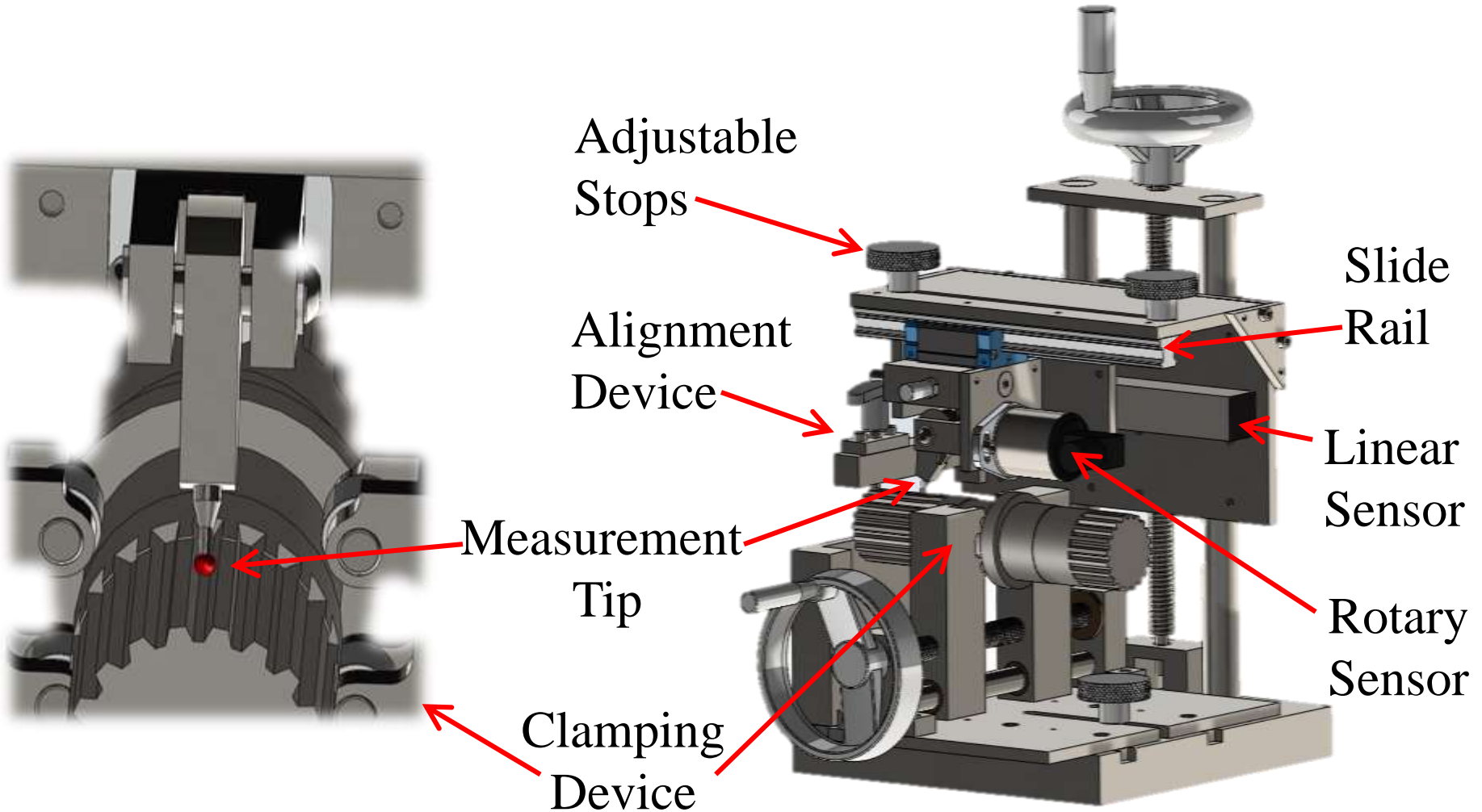
Initial Design (Cont.)



- Computation
 - Microcontroller contains algorithm
 - Rotary sensor collects samples
 - Samples respect to position
 - Input to algorithm
 - Algorithm computes twist
 - Display the twist

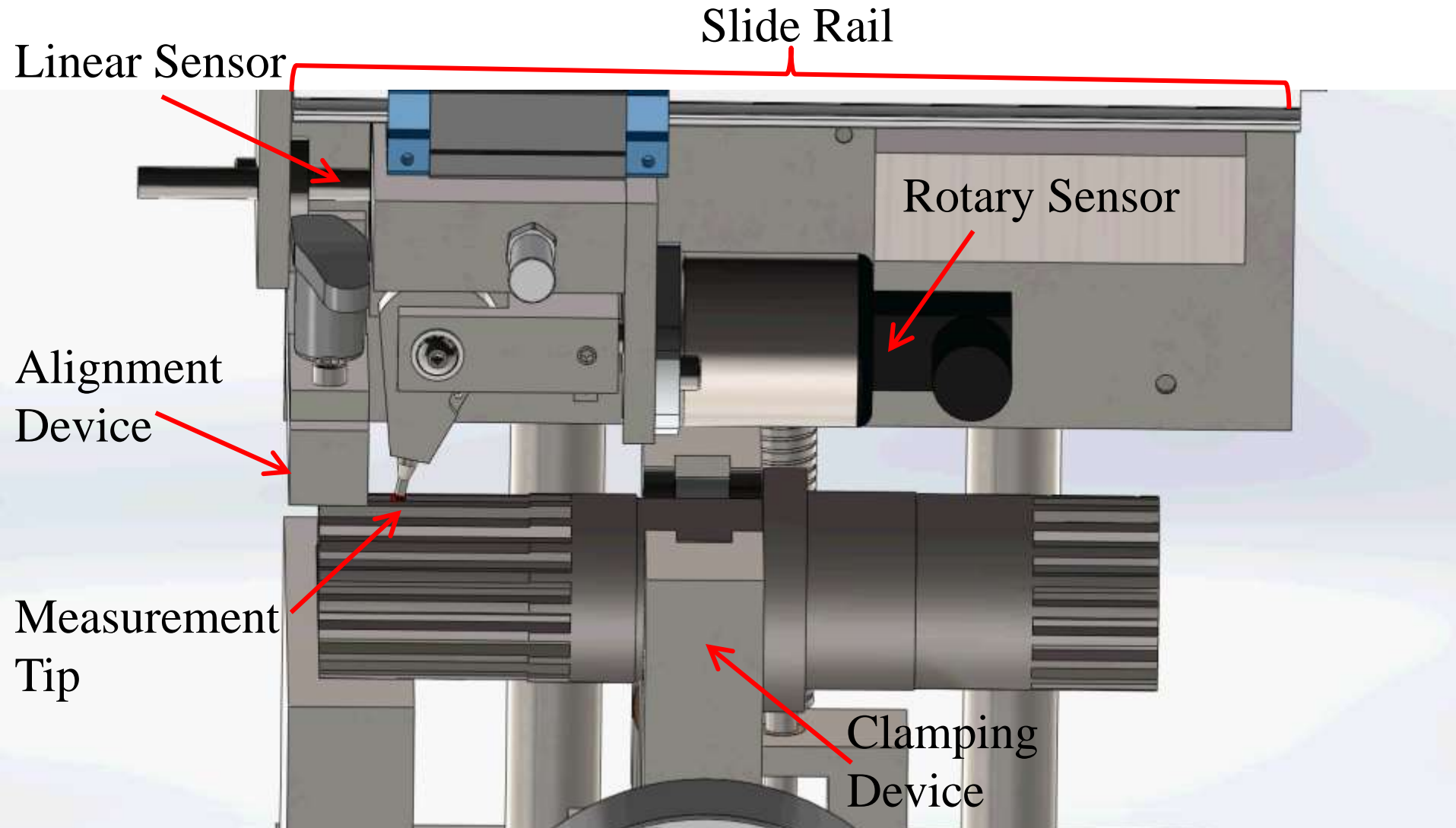


Final Design





Final Design (Cont.)





Final Design (Cont.)



- Sensors
 - Inductive rotary sensor
 - Resistive linear sensor
- Microcontroller
 - PIC32MX795F512L
 - 32 bit
 - 128 KB RAM
 - 512 KB Flash
 - 76 MHz
- External 12-bit ADC
 - ADS7800KP
 - Parallel
 - Quantizes linear sensor



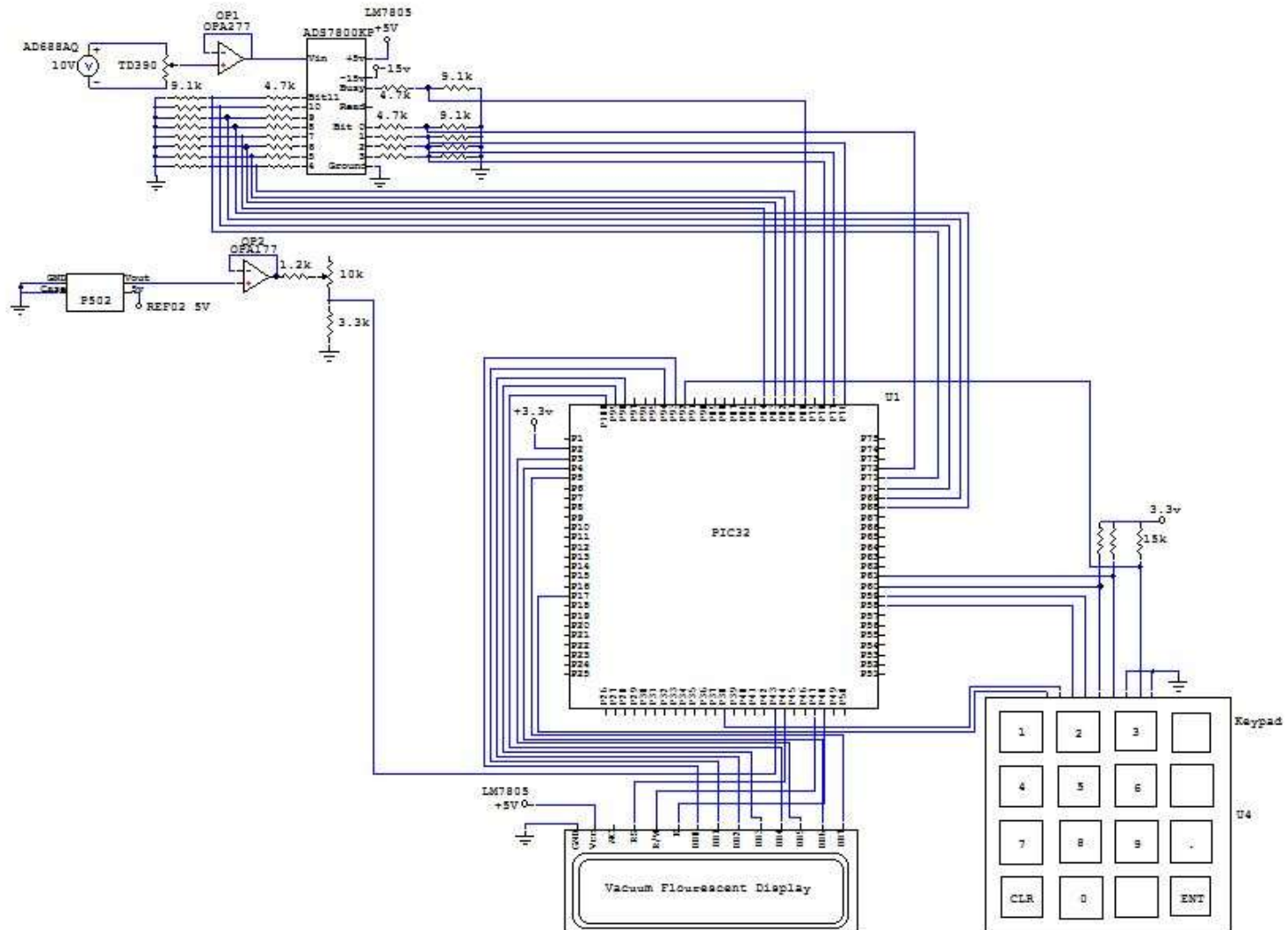
Final Design (Cont.)



- Internal 10-bit ADC
 - Onboard PIC
 - Quantizes rotary sensor
- Keypad
 - Matrix
 - Active low
 - 8 pins
- Vacuum Florescent Display
 - Higher storage temperatures
 - Operation identical to LCDs

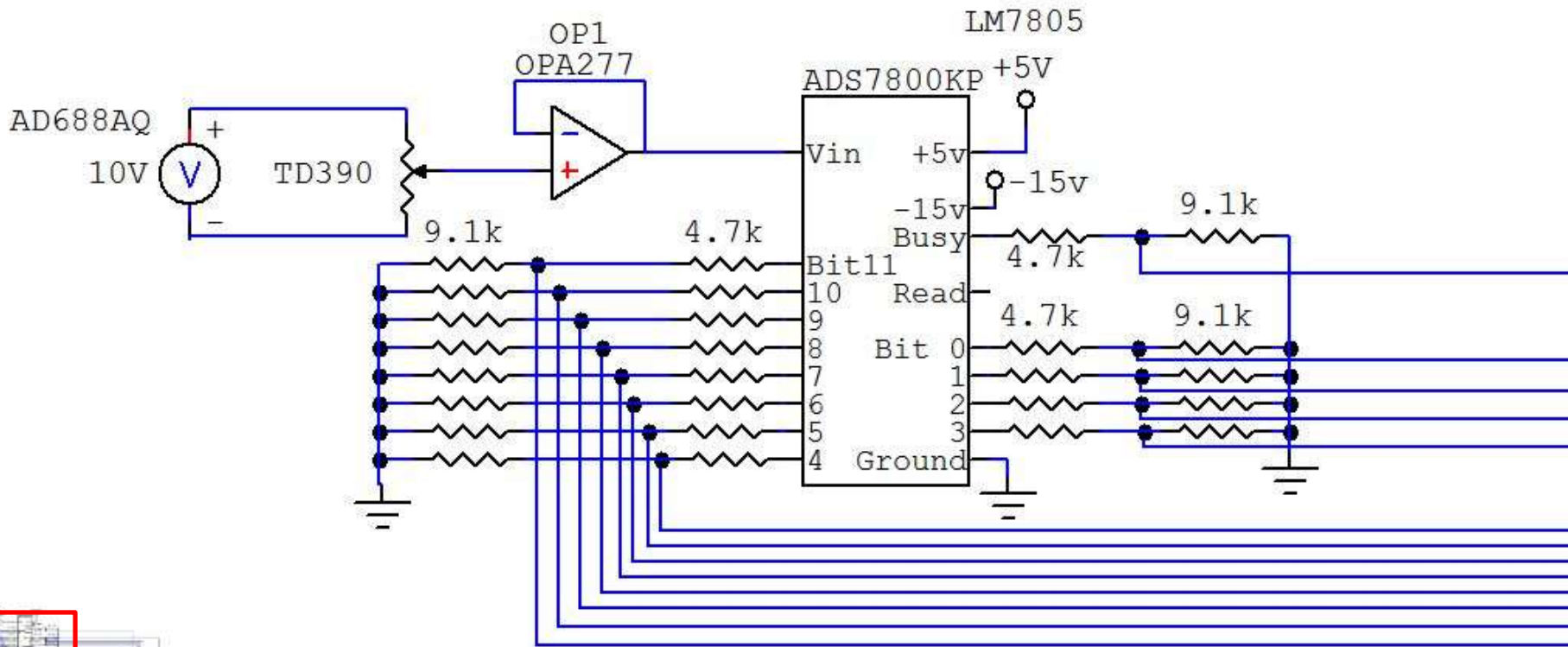


Final Design (Cont.) Schematics



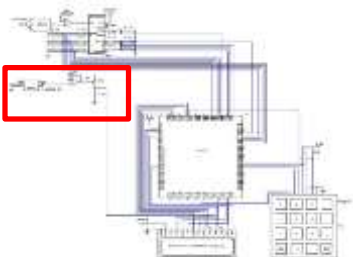
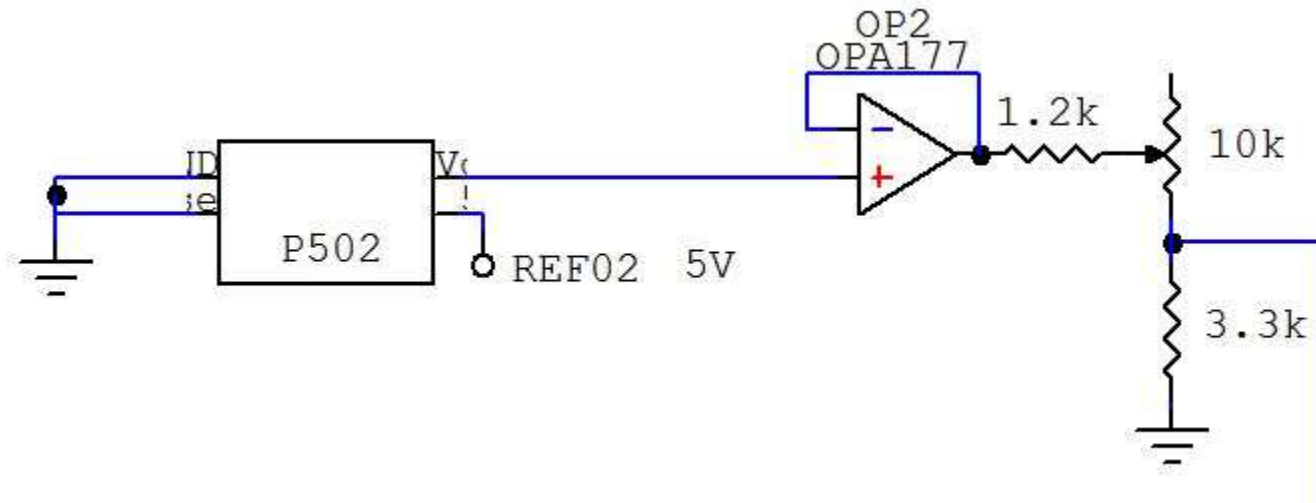


Final Design (Cont.) Schematics

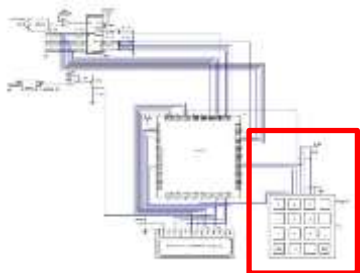
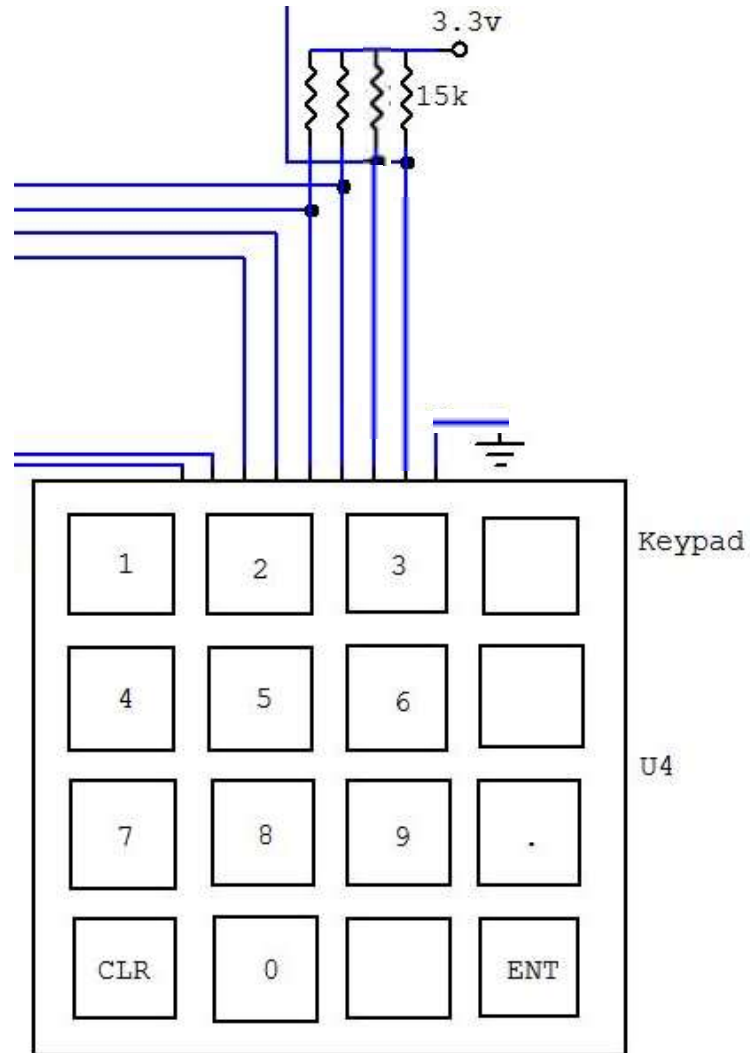




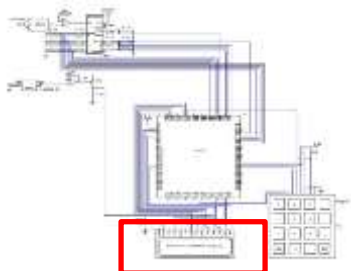
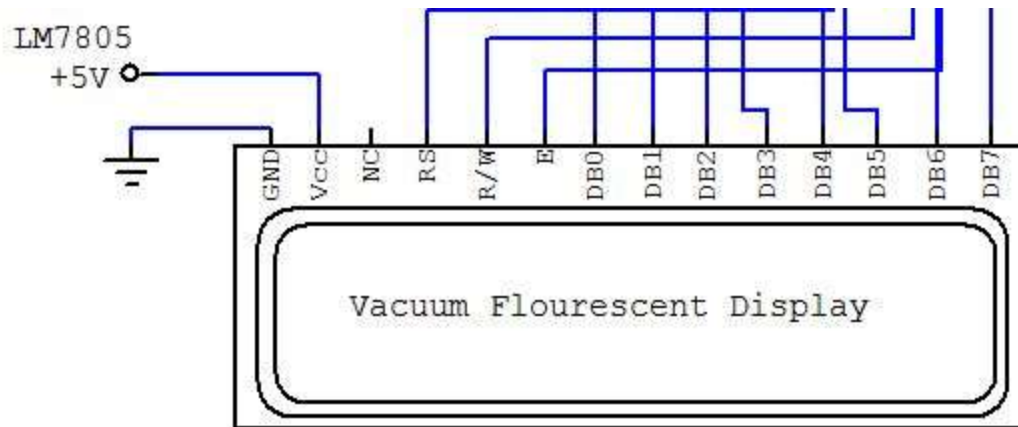
Final Design (Cont.) Schematics



Final Design (Cont.) Schematics



Final Design (Cont.) Schematics





Final Design (Cont.)



- Startup
 - Obtain diameter from user via keypad
 - Obtain location of twist from user
- Sampling
 - User sets up tool and shaft
 - Press enter to activate ADCs
 - Move rotary sensor to take sample
 - Press enter when done



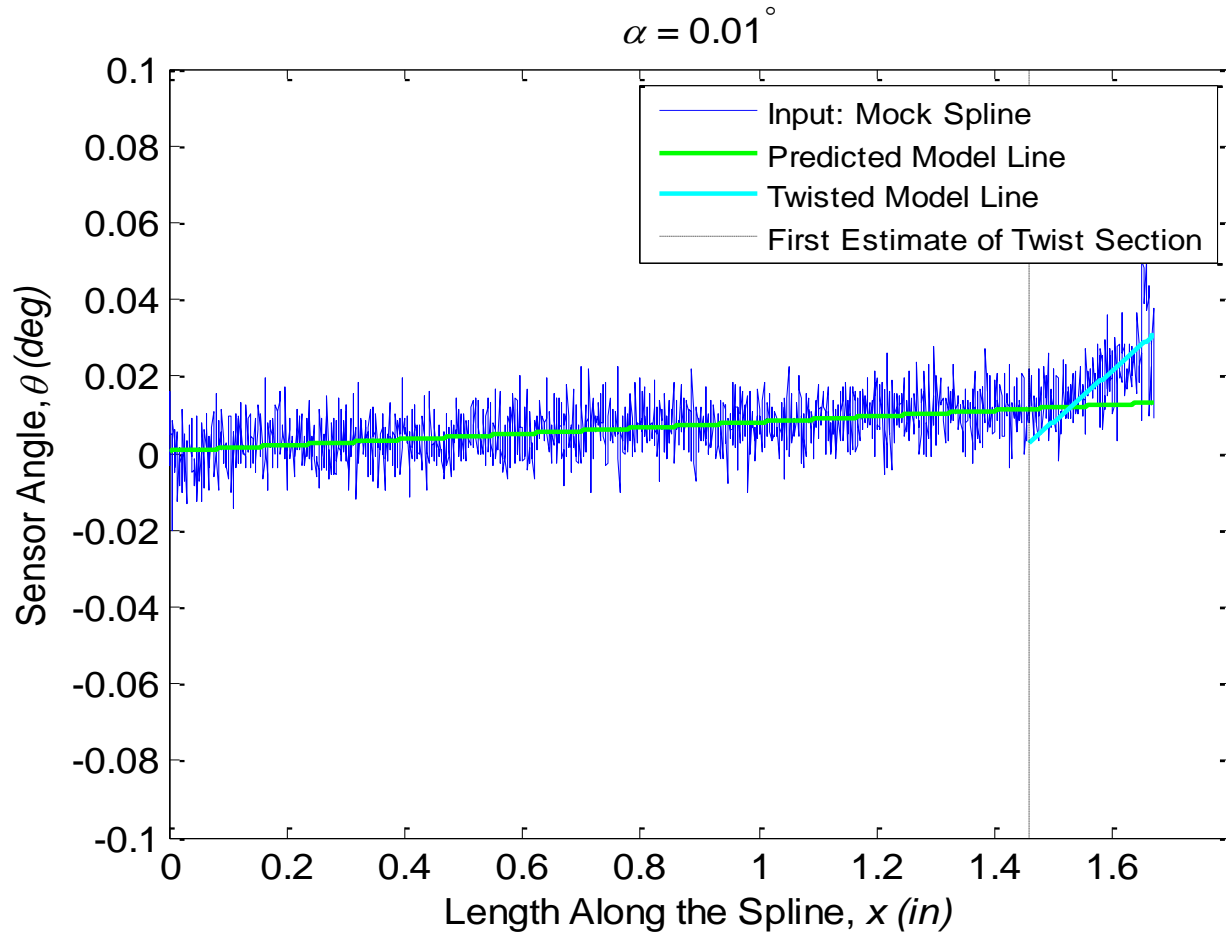
Final Design (Cont.)



- Calculation
 - PIC stores both sensor's data as vectors
 - Algorithm takes both vectors as inputs
 - Displays result in degrees
- Further Calculations
 - Option to take another sample
 - Max of 5
 - Returns average

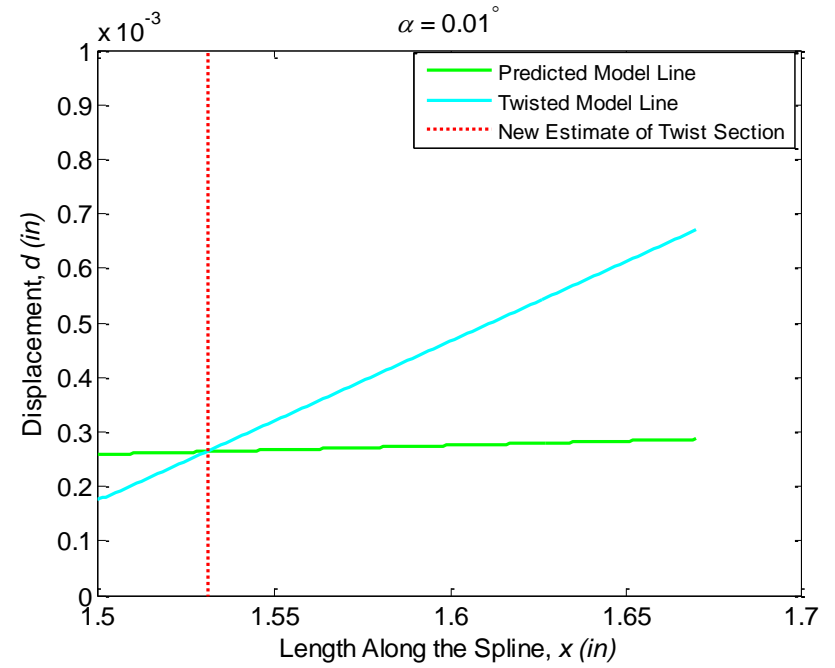
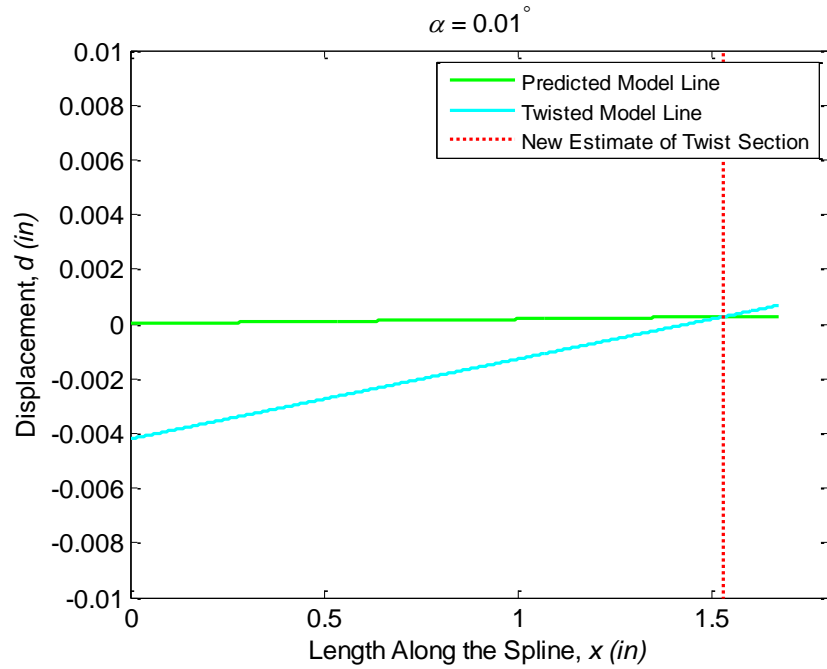


Algorithm





Algorithm





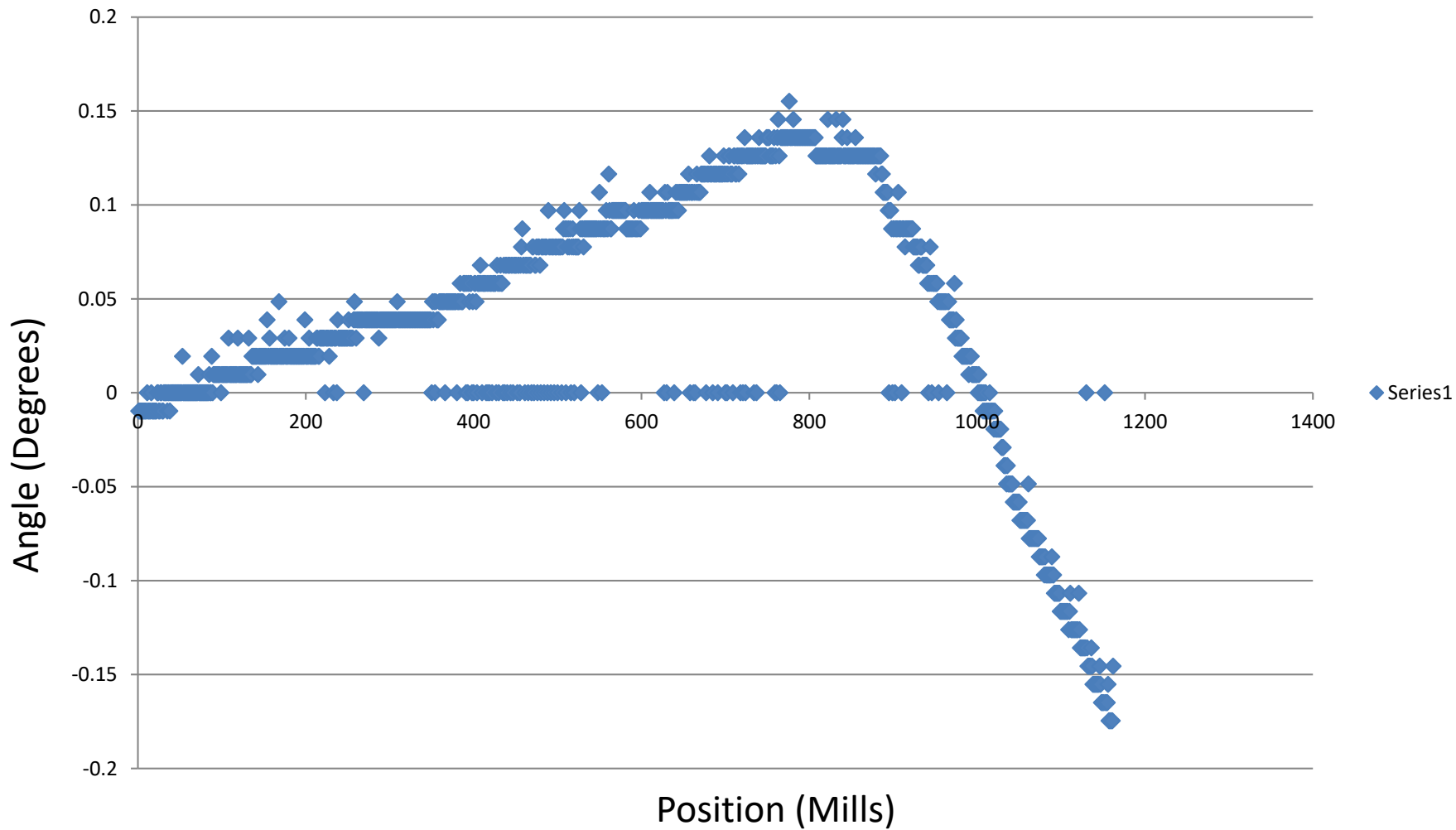
Testing



- Tested Two Shafts
 - PTO and Combat Vehicle shaft
 - Tested 5 times on 5 splines
 - Pulled raw data from ADCs
- Algorithm Testing
 - Used data for MATLAB and C codes
 - Observed differences in results

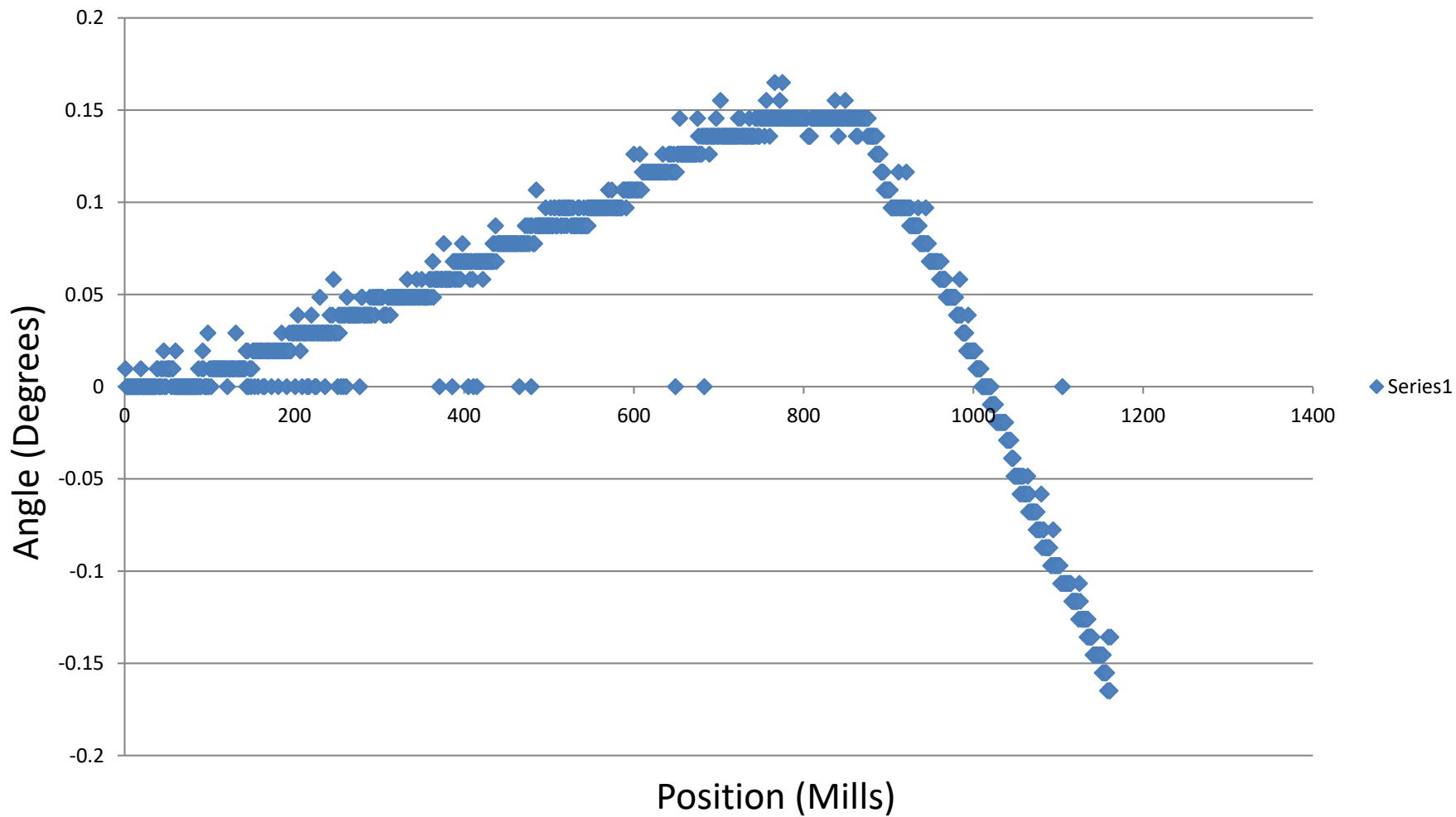


Testing (Cont.)



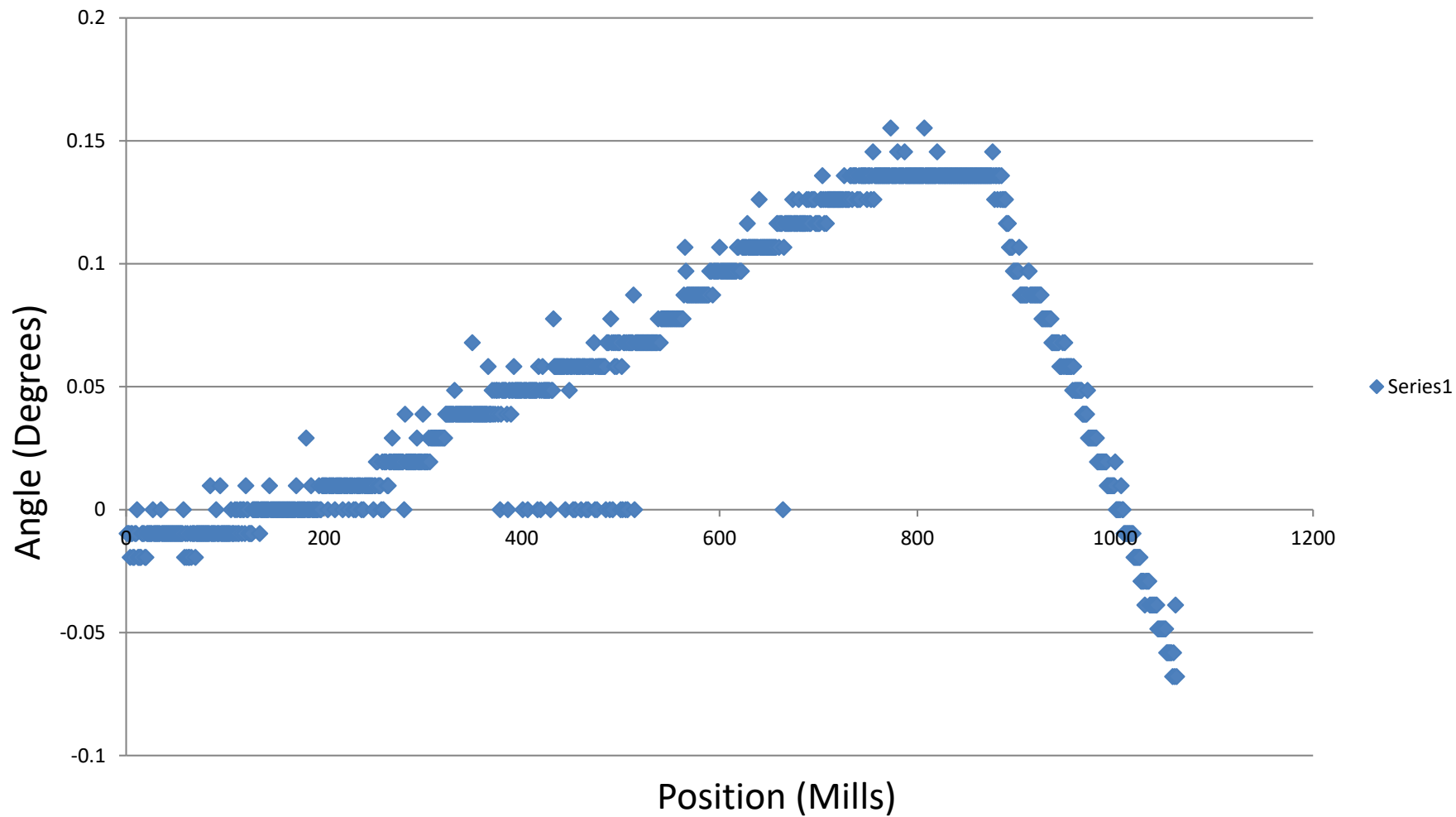


Testing (Cont.)



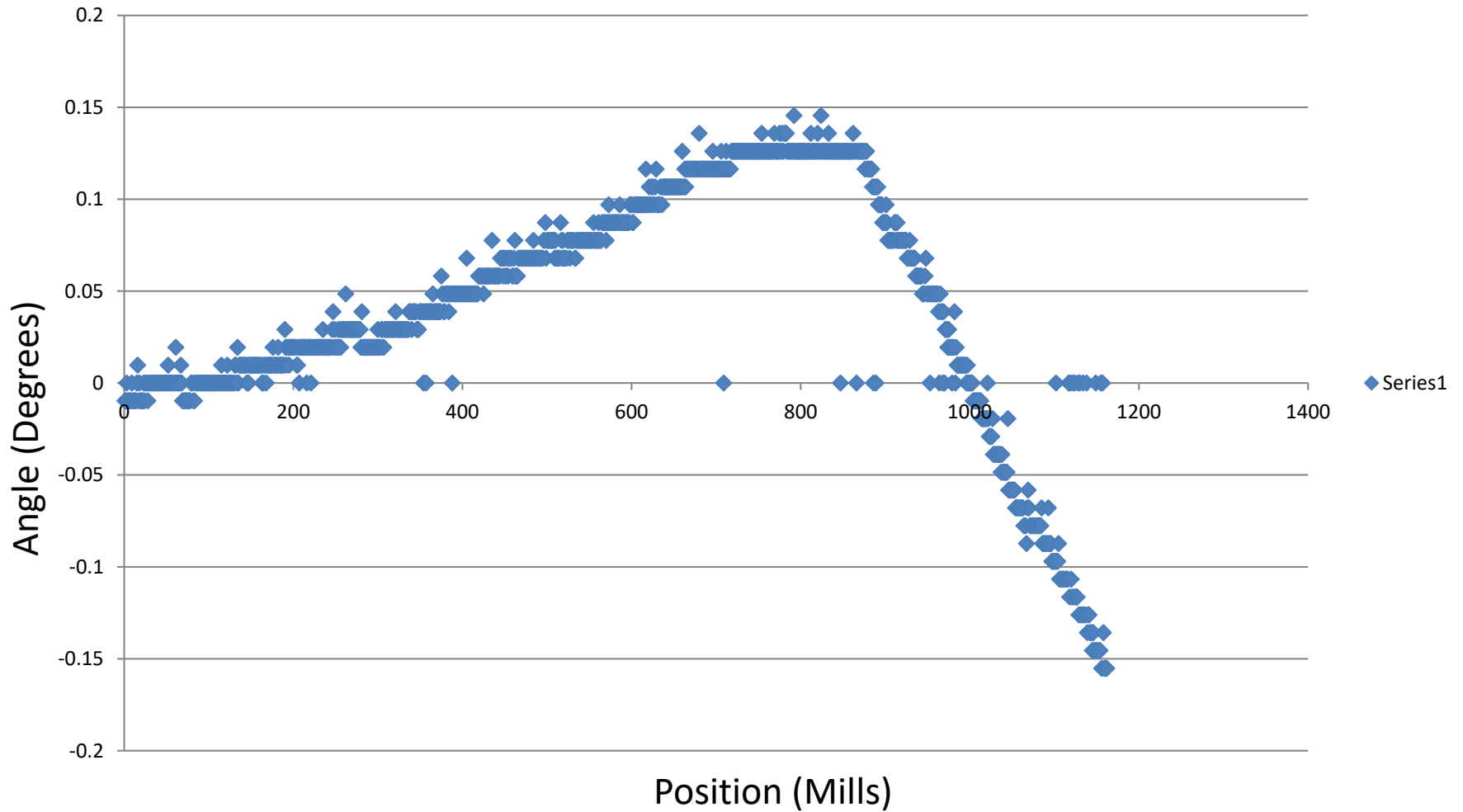


Testing (Cont.)



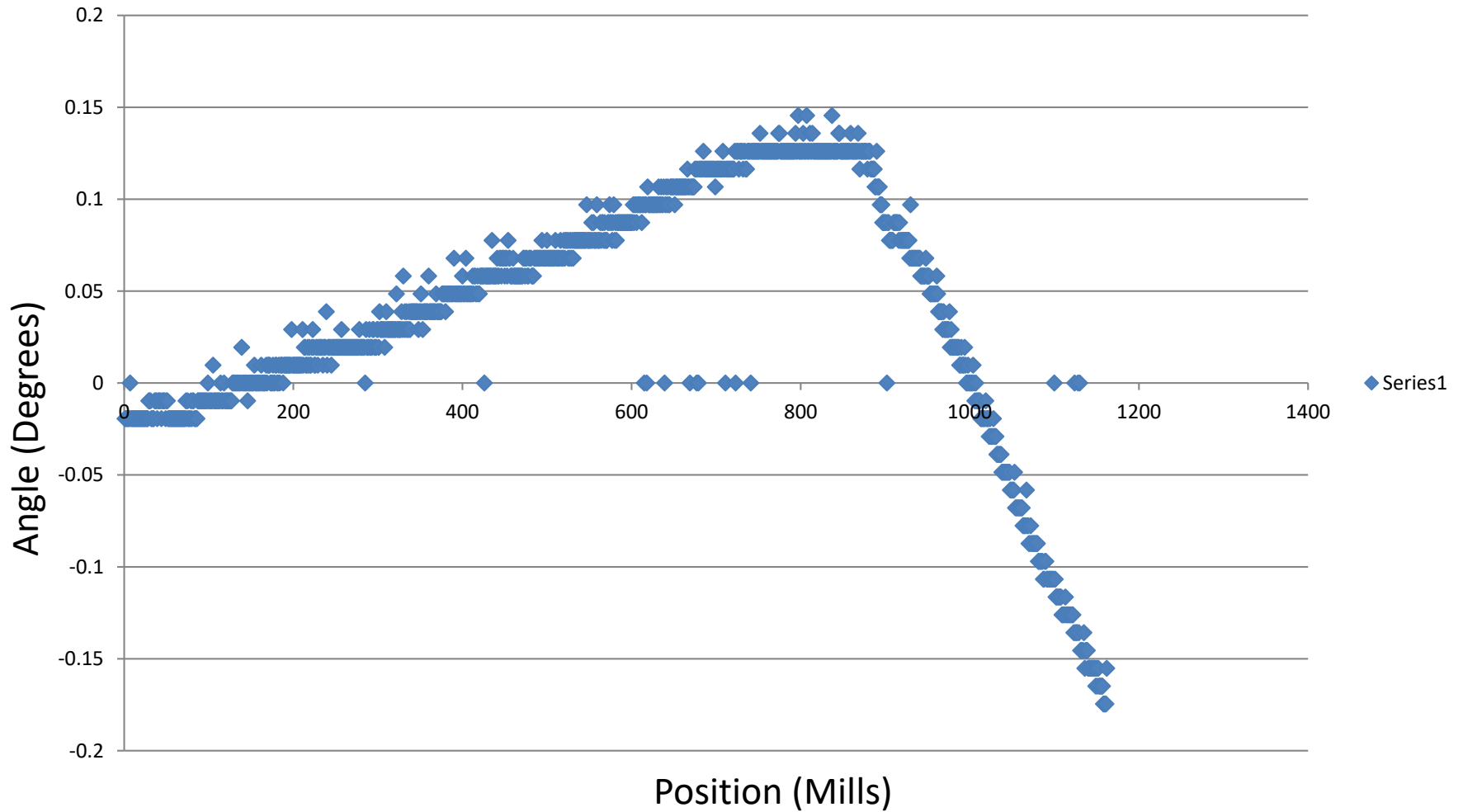


Testing (Cont.)





Testing (Cont.)





Results



- Raw Data
 - Have some zeros clustered together
 - Corrected in algorithm
 - End of sample experiences noise
 - Ignore the very end
- Algorithm
 - Going from MATLAB to C
 - Calculated correct values for gamma
 - Fluctuating results for alpha



Conclusion



- Fast
 - 5 second calculation time
 - Precise to a hundredth of a degree
- Met Requirements
 - All required
 - Plus additional
- Working Concept
 - May be used for further refinement



? Questions ?

