ELECOMP CAPSTONE DESIGN PROGRAM SPRING 2025 SEMESTER

Homework #15

DUE: Sunday, April 6th, between 8 and 9pm, via Google Form attachment; to Mubariz Saeed.

Each Team will compile a word doc, with the information below, for approval by your Technical Directors, for the Summit Poster. You MUST use the Template provided at the end of this section, and also include the 4 Figures:

- Project Motivation (~150 words)
- Anticipated Best Outcome (ABO) (~100 words)
- Implications for Company & Economic Impact (~100 words)
- *Total of 350 words, for these 3 sections (+, or – 10-word flexibility in each section)*
- Project Outcome: *(ABO achieved or not)*
- Key Technical Accomplishments (600 words): use bullet format, with a few words title in bold, followed by a few lines of explanation. See the example in the template provided at the end.

Label the doc as: "Team Name: Summit Doc: SPR2025"

We also need 4 diagrams/pictures/etc, ***in color***, for inclusion in the Summit Poster. Label them as **Fig. 1** through **Fig. 4,** and refer to them in the Key Technical Accomplishments, in the word doc, in **bold**. Each figure must have a brief caption and should be below the figure. Low resolution figures can be included in this doc. (see example template below)

(When compiling the Summit Poster, you must use high resolution figures. More details at this Link:

https://web.uri.edu/elecomp-capstone/files/High-Resolution-Images-Guide.pdf

The Final Poster that you will submit, by using the Summit Poster Template on the website, will look like:

https://docs.google.com/presentation/d/1Nkdco87nGq3DZCJ6Ar3Yv4nlRLQKkvtUHj85aP9zxfc/edit#slide=id.p3

The Submission date is April 8th, 2025 (to be confirmed, after approval from your TDs)

Team Cambridge Technology: Motion Control: Summit 2025

Project Motivation:

Cambridge Technology manufactures Galvo based laser scanners which allow for easy laser control. The ScanMaster Controller (SMC) is a proprietary hardware system used for generating programmable control signals which can interface with many common lasers. The controller also can be integrated with the customer's own software application or can be commanded through Cambridge Technology's ScanMaster Designer (SMD) software. The purpose of this project is to add to the robustness of the SMC by integrating the ability to control external microstep motors. When a scanning head is mounted above the target plane, there does not currently exist any native way to expand or adjust the region that is "scannable" by that device. By introducing one or multiple stepper motors to the system, the customer will have the ability to dynamically change the area in which they can etch or drill their designs by moving the actual scanning head directly from the SMD and SMC.

Anticipated Best Outcome:

As per the original specification of the project, the Anticipated Best Outcome was to deliver a platform, embedded within the SMC's dual-core Xilinx Zynq FPGA+ARM Cortex-A9 Processor environment, that manages the hardware and software interconnect to control a step-and-direction motor along a linear slide using precise motion control. The overall system should have maximized the operation of three integral facets: the FPGA logic necessary for step/direction pulse delivery, an intelligent software API for generating motion profiles and interfacing with the hardware fabric, and a demonstrated integration into the full Scan System product.

Implications for Company & Economic Impact:

The addition of motor driving capabilities from the SMC will greatly expand the application space of Cambridge Technology's Galvo based laser scanner solutions. It will allow for scanning systems to have a dynamic working range and be less limited by its stationary XY plane. Naturally, this imbues the product with more value to customers, and it will expand the potential market for the product because of the newfound ability to handle a wider breadth of issues faced by production companies and OEMs. A rough estimate for the impact of this feature on the revenue of the company is in the range of $250k - $500k from additional sales.

Project Outcome:

The Anticipated Best Outcome of the project was achieved.

Key Technical Accomplishments:

- **Custom Logic Implementation:** A black-box custom logic module for motion control has been successfully integrated into the SMC's main logic element. We have also added writing support for the previously unused stepper motor ports on the SMC's official I/O header, establishing a gateway to the outside world.
- **Step and Direction Pulsing via SMC:** Established and tested the ability to generate pulse streams in the logic and drive an external stepper motor with those signals in either direction. Full control of the linear slide/cart from the ScanMaster Controller FPGA logic has been accomplished by parameterizing the different variables needed to translate distance into the necessary number of step pulses. These variables are further utilized when attempting to control the motor via the CPU cores.
- **Motion Profiling Algorithms:** Developed rapid mathematical algorithms for generating velocity slopes based on settable acceleration parameters. We have written scripts to simulate our algorithms ahead of time, allowing us to know how a move will behave before even powering on the board. **Fig. 1** shows two examples of graphs generated by our algorithms when passed through our simulator.
- **The Motion Control Software:** We have implemented the motion profiling and translation algorithms in C++ and provided an API for setting move parameters from upper layers. All parameters necessary to describe a motion control environment are encapsulated inside of a custom software structure - therefore multiple motion axes are readily supported in the future. From the ScanMaster Designer, our underlying software is easily actuated via the program shown in **Fig. 2**.
- **The Motion Control Hardware:** Developed a robust pulse delivery hardware system which communicates with the control software via the AXI interface. The software sends the period and pulse count commands into a FIFO from which our stepper state machine requests the 32 bit command that tells the motor how fast and how far to move. The state machine then provides the necessary parameters for motion to the pulse delivery servicer. The servicer, as its name implies, delivers the necessary number of pulses to the external step port at the commanded frequency. The speed of pulse delivery is managed by the pulse edge timer which signals the pulse delivery servicer when to toggle the step signal.
- **Motion Profile and Control Register Command Formats:** We defined two completely custom command formats, depicted in **Fig. 3**, to interface directly with our hardware from any software application. The Motion Control Software has been specifically designed to translate the mathematical motion profile into a set of *motion profile commands* that follow the above standard and send them off to the FPGA fabric. Similarly, the software utilizes the *control register command* format to perform administrative interfacing with the hardware on an auxiliary, 2-way communication channel.

- **Integration into the Full System:** As per the Anticipated Best Outcome of the project, the Motion Control Software and the Motion Control Hardware have been successfully ported into the full ScanMaster Controller. This means that the customer is able to configure a motion control axis to their particular environment and send move commands directly from Cambridge Technology's ScanMaster Designer GUI. See **Fig. 4** for the flow of information produced as a result of this project.
- **Robustness and Quality Checking:** Performed extensive testing of the hardware and software under a variety of edge cases and operating conditions to make the framework more robust (improper command handling, maximum speed and acceleration testing, extended runtime stability, etc.). Several adjustments, performance optimizations, and bugs were identified and taken care of.

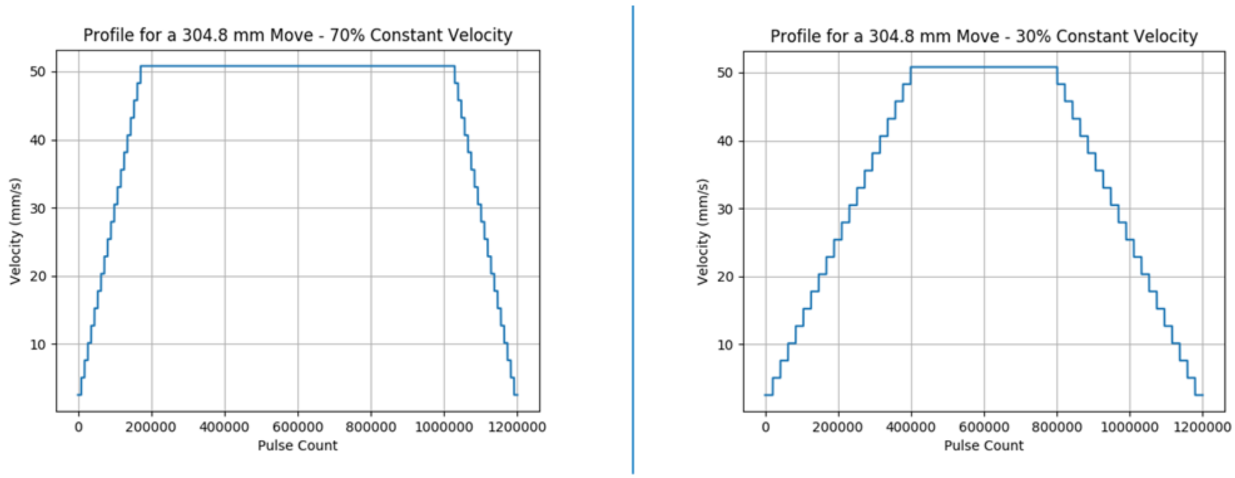

Fig. 1: The simulated results of our motion profiling algorithms. The programmer has control over how quickly the motor accelerates the platform. When 70% of the move is specified to be at constant velocity (left), the slope is quite steep, and the acceleration is high. When 30% (right) is commanded, the slope is noticeably less dramatic.

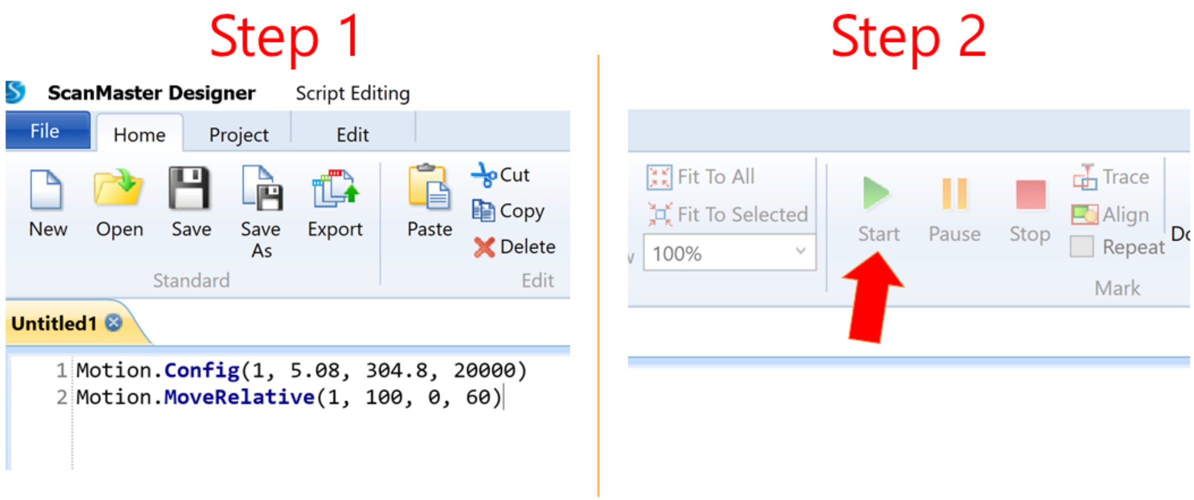

Fig. 2: Controlling our stepper motors and linear slide using Cambridge Technology's proprietary ScanMaster Designer graphical user interface. The following 2 lines of commands in the SMD are all that is needed to set up and activate the underlying Motion Control Hardware and Motion Control Software:

Motion.Config(<axis>, <pitch>, <slide length>, <steps per rev>)

Motion.MoveRelative(<axis>, <distance>, <direction>, <velocity>)

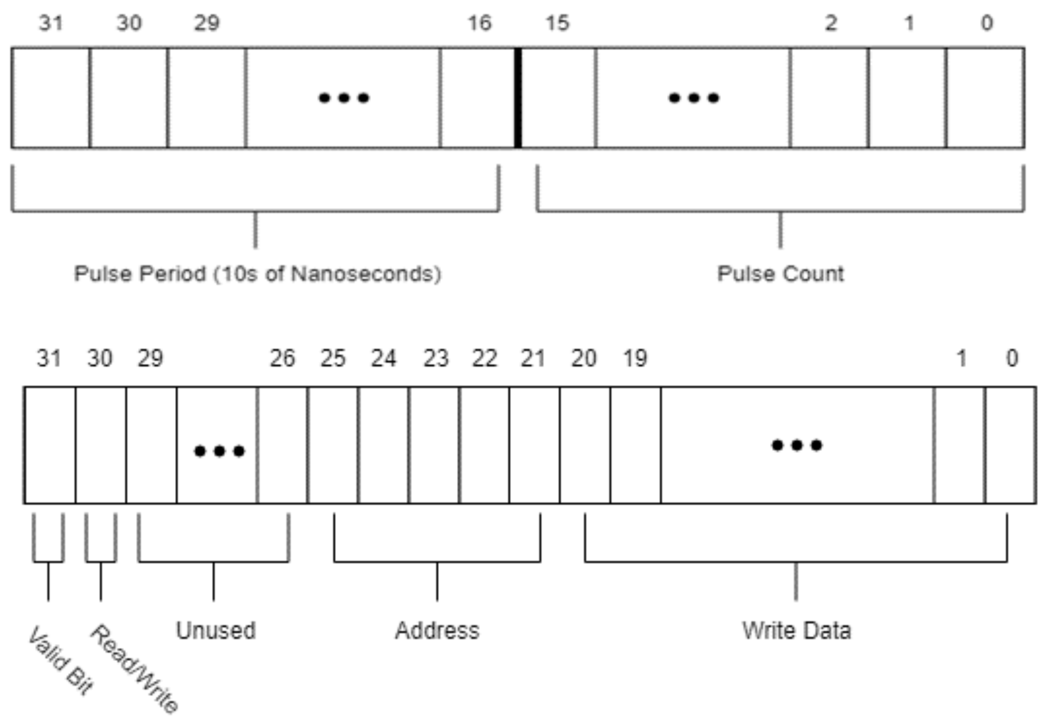

Fig. 3: Our designs for command formats. The motion profile command format (top) allows for precise control over the velocity, acceleration, and distance of each movement command. The control register command format (bottom) was implemented for easy communication between the software and the state machine core of the hardware.

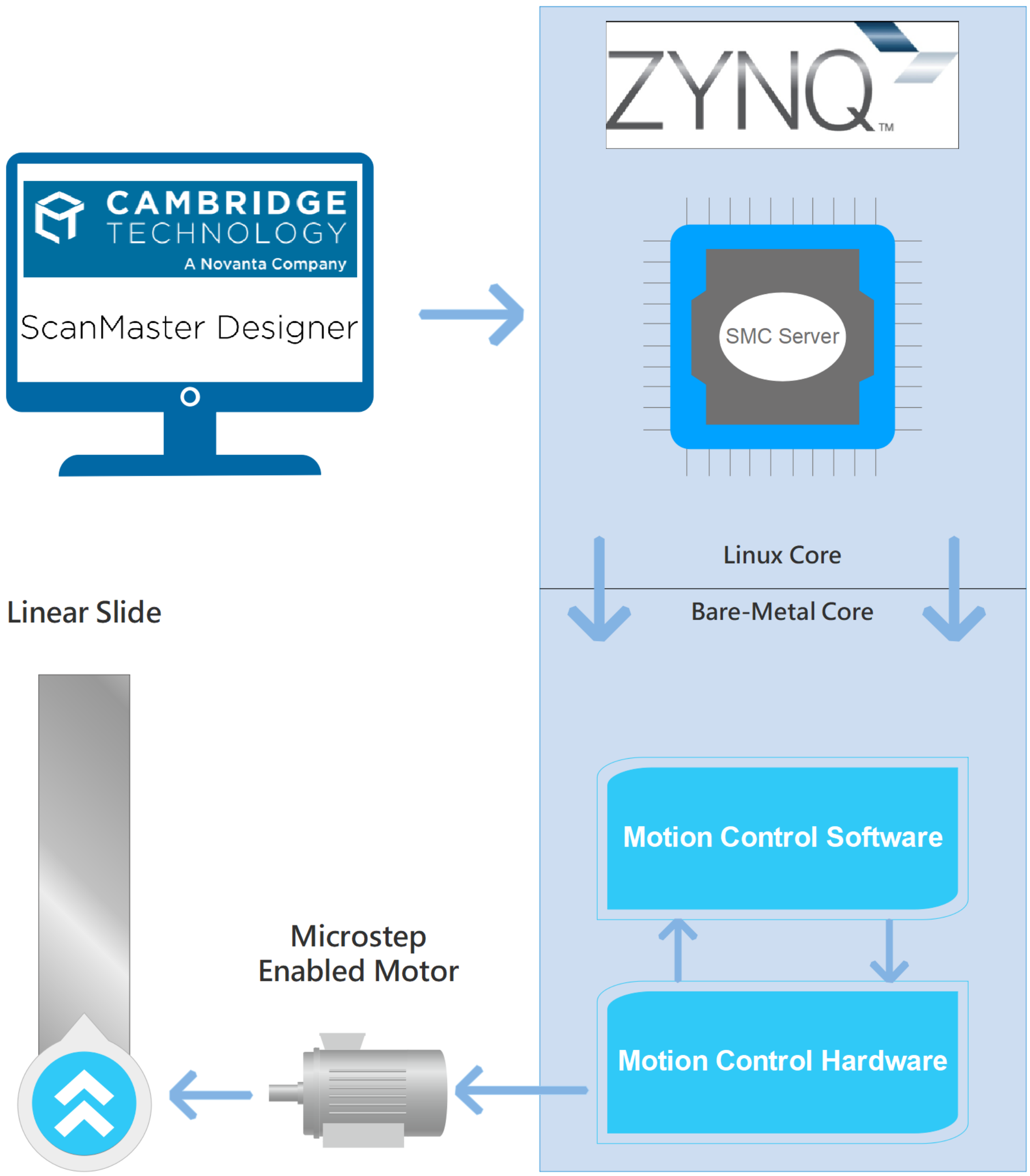

Fig. 4: A visual representation of the flow of control for our stepper motor positioning system under user input.