MSP430 Motor Control Card

Progress Report 2

ECE480 Design Team 4
Roy Dong
Micajah Worden
Andrew Kleeves
Mark Barnhill
Dave Seaton

Facilitator: Professor Elias Strangas
Sponsor: Tim Adcock, Texas Instruments

Introduction

Since our last progress report, many provisions have been made to get our project back on schedule. Most importantly, we have been allowed an extended budget to get our redesigned DIMM card fabricated and populated. There have been limited modifications to hardware, but software is under development at a rapid pace. In addition to hardware and software changes, the motor is now fully characterized with simulations of both basic inputs and PI control. However, the project has encountered some minor delays. Backorders and space constraints of the MSP430G2231 have prevented us from moving ahead as scheduled. The workarounds that we developed are further detailed below.

Hardware

The motor card has not evolved much since its original revisions. The mechanical drawing and schematics have been fully updated to reflect any additional design changes. However, our initial budget of $500 proved to be not enough to both fabricate and assemble a chip with 89 components requiring surface mounting. Initially, we tried to find an option within our budget, which would have been $350 for fabrication and kitted assembly, after parts. Then, we realized that this was likely impossible, especially within the context of a semester-long time frame, and worked with Professor Shanblatt to restructure our budget. We received approval to fabricate the board for under $1300.

As a precautionary step, we have all reconfirmed that the connections on the MSP430 DIMM card are indeed connected to the corresponding C2000 pins. Recently, the circuit has been shipped out to Sunstone Circuits for PCB fabrication and assembly. Also, according to TI, the DRV8412 motor driver card revision itself has been modified; however, we have been assured that any changes will only require modifications to software and not hardware. We are likely not receiving a new version of the DRV8412 before Design Day.

We have also found a way to optimize our time usage as our hardware is being created. We plan to develop software on the MSP430 while waiting for our motor card. We were provided with a MSP430G2231 experimenter board, known as the MSP430 LaunchPad. However, the MSP430G2231 did not have enough memory to implement a PI controller. We have ordered an MSP430F5438 experimenter board, and plan to implement our entire project on the MSP430F5438. The MSP430F5438 experimenter board was shipped out yesterday. TI originally did not have any in stock, but a Spartan at TI managed to get one shipped to us, expedited, from Europe. This MSP430F5438 will interface with the DRV8412 through a simple DIMM PCB card which routes connections. We are currently in the process of making this DIMM card. This solution will allow us to develop software while we wait for our hardware fabrication and assembly.

In addition to actual hardware design, the DC machines that were given to us by TI must be characterized and simulated before used, especially with a PI controller amplifying the input signal. A ramp control module in software prevents a sudden spike in the reference value, protecting the motor from sudden bursts of current. The simulations confirm this. After acquiring the characteristics of the motor, a simulation was run in MATLAB and then Simulink® to emulate ramp and step motor inputs, as well as PI control of the motor. Parameters were found using root locus methods, and will be implemented in the software.

Software

As mentioned in our last progress report, the MSP430 does not have any high-precision, fixed-point arithmetic library. We contacted our sponsor to determine what math libraries we should be using, and he promptly replied with some MSP430 code. Looking into some MSP430 code designed to perform PID control of a fan, we found that this code was able to implement PID control using basic C variables, such as ‘int’ and ‘double’. We decided to use basic C variables as well, given the unavailability of specialized

math libraries, and a time frame that would not allow the development of a robust high-precision mathematics library.

After doing so, we have implemented a ramp control module, as well as a PID regulator module on the MSP430. The implementation was inspired by similar code for the C2000; however, it had to be re-written to use basic C variable functions and methods instead of the specialized high-precision functions and methods provided through the IQMath library. The nature of digital microcontrollers require that any PID implementation be done digitally, so when we use the PID parameters from a continuous-time simulation, we will have to scale the integral and the derivative gains based on the frequency of the algorithm.

Currently, we have successfully written code for the control of two brushed DC motors. When the appropriate hardware arrives, we will test this code thoroughly. We are also looking into a stepper motor control implementation while we wait for hardware to arrive.