2^{nd} Semester Project Proposal

Joshua Wise

February 11, 2006

When I started thinking about this project, I had a lot of ideas. There were a lot of projects that I've already gone through the first couple stages of[1], such as my build-a-UPS project and my portable desktop computer, but most of those seemed like they'd be done a lot sooner than three months (and for over $50). For this project, I was looking for an excuse to do something that I haven't really done before (and get some grades for it), and I decided on control theory. On the net a while ago, I saw a few descriptions of people who have built their own Segways, and it looked interesting, so for this project, I decided that building my own self-balancing scooter would be an interesting idea – and an interesting challenge to keep it within the time and budget limitations.

The basic idea behind such devices is to use a gyroscope to determine which way's down, and which we we're pointing, and an accelerometer to correct for the long-term skew on the gyroscope. When the calibrated systems determine that we're not facing straight up and down, then it will move the wheels appropriately to place them back under the scooter. The wheels are moved by a pair of motors that are modulated by a pair of H-bridges.

In this project, a key part of the design process is keeping the device within the $50 limitation. This initially seems somewhat unreasonable – one implementation of this project used a pair of $250 motors, along with a $500 motor controller and a $100 gyro. But by sacrificing some accuracy and performance (and by using some free samples), we can get a much more reasonable device. Tackling the most expensive element first, we have the motor controller. This is trivially replaced by a set of 8 MOSFETs and gate drive transformers, both of which are sampleable. Total cost thus far: $0.

The next device that needs to be chopped out of the equation is the gyro. Luckily, a google search for "gyro sensor sample" gives a MEMS gyro from Analog Devices. The chip is fairly trivial to connect to a microcontroller using its built-in SPI interface. Likewise, Freescale produces an X-Y-Z accelerometer that also does pretty much exactly what we want. Total cost: $0.

Motors are, unfortunately, not sampleable, as far as I can tell, but luckily, a certain type of motor is dirt-cheap. Motors from "Power Wheels" cars (you know, the plastic motorized things that little kids get ...) and their gearboxes go for about $20 for a pair on eBay. They're incredibly powerful, as well: at 12V, they draw 140A stalled (although they're not doing any work like that). However, from previous explorations with robotics, they still are very powerful when actually doing work. Total cost: $20.

Microcontrollers are similarly sampleable; a pair of PIC18F2550s should manage this easily. For power, one will likely want a pack of sealed lead-acid batteries. Those are available for free at the dump, as long as one is willing to spend some time determining which ones are functional, which ones require desulfation, and which ones have holes blown in the plates. A charging circuit can be built for $5 or so. Total cost: $25.

There will probably be various other mounting devices that cannot be sampled from the manufacturers, that may cost up to $25 at Home Depot under this budget. A PVC frame will probably be used with some low-cost metal to mount things. Bicycle tires from the dump will likely be used as wheels. This completes the $50 Segway budget. (Yes, I'm serious.)

So what's going to take three months? In addition to assembling this system, I will need to write firmware for the microcontroller(s) to read input and control the motors. This is non-trivial, but should not be unacceptibly difficult. In scheduling projects, I find it helpful to set milestones for various bring-up tasks. Some of the processes that can be executed in parallel, with their various milestones, are:

- Microcontroller bringup
  - Reading (and accumulating) the gyro and reporting back to the PC

[1] thinking about a design, shooting down the initial design, thinking about a new design, making a first pass implementation, exploding some parts, shooting down the next design, ...

  - Reading the accelerometer and reporting back to the PC
  - Pulse width modulation based on gyro position
    * At this point, the first tests may be able to be performed
  - Steering control for two motors based on gyro position
  - Calibrating the gyro automatically based on the accelerometer
  - Data output to a small 20x4 LCD that I have
  - Battery sensing?

- Electronics bringup
  - Controlling one motor with one transistor
    * Dealing with inrush current
  - Controlling motor with four transistors
    * Dealing with inductive kickback
    * Forward motion
    * Reverse motion
    * Finding optimal pulse width modulation frequency
    * Connecting to microcontroller
  - Dealing with unexpected circumstances?
    * Low battery – reacting safely
    * Avoiding MOSFET shoot-through
    * Sanely limiting current through FETs
    * Overheat

- Hardware bringup
  - Creating a basic frame
  - Mounting motors and gearboxes
  - Mounting gyro and accelerometer
  - Mounting electronics
    * Heat sinking and other thermal concerns
  - Mounting batteries
  - Mounting controls in top of unit

- Testing
  - Eh, that's what helmets are for

Obviously, I have my work cut out for me. But, the project should be highly rewarding when complete.

The final deliverable will be a self-balancing device in some form. Ideally, I'd like to ride it to school for the presentation, and depending on its state of safety, allow others to take control of it, although I'm not sure how a homemade device will handle actual New England road conditions. We'll see how it plays out.