



Low Level Design

Electrek

**EE Senior Design
February 18, 2011**

**Katie Heinzen
Kathryn Lentini
Neal Venditto
Nicole Wehner**

Table of Contents

Table of Contents	2
1 Introduction.....	3
2 Problem Description and Proposed Solution	3
3 System Description and Block Diagram.....	4
4 System Requirements.....	5
4.1 Overall System.....	5
4.2 Subsystem and Interface Requirements	6
4.3 Future Enhancement Requirements	8
5 Low Level Decisions	8
5.1 Alternator / Current Sensor Subsystem.....	8
5.2 Battery/Inverter	11
5.3 Microcontroller and Software	15
6 Major Component Costs	20
7 Conclusions.....	21
8 References.....	21

1 Introduction

As renewable energy technologies continue to develop, new ways to generate electricity have been implemented. The owner of Outpost Sports in Mishawaka would like to generate electricity from his bicycling training facility. The facility allows eight riders to bring in their bikes to train during the winter. The back wheel of the bike is held in a frame in contact with a braking system to provide resistance. During the 30 minute training session, the average power produced by each rider is 250 watts, with peaks of up to 400 watts. Currently, this is all lost to heat. This project is an opportunity to use energy that is already there in a useful way. If all of the energy from the eight riders could be converted to electricity, it would be enough to power ten 100 W bulbs for an hour. ElecTrek's goal is to work with the current system to achieve this conversion by adding components that capture the energy as electricity and store it in a battery or use it during the session to power a device such as a fan or an iPod.

2 Problem Description and Proposed Solution

The customer for ElecTrek is the owner of Outpost Sports, J.V. Peacock. Mr. Peacock identified a way to improve his bicycle training facility and to save him money on electricity. The goal of this project is to route the mechanical energy generated by the riders into electrical energy that he can use elsewhere in the store. Of the eight set ups, only one will be altered for the project, but if successful, Mr. Peacock can implement the design on all eight systems. There is also the option of contacting the company that produces the systems, RacerMate, about adopting the design and improving it to market with their products.

The training facility uses a system called CompuTrainer. CompuTrainer controls the resistance seen by the rider. It allows individuals to program their own rides, via a controller on the handlebars, or to use a preprogrammed course that is controlled from an external computer. The software includes videos of international races that can be projected on a screen to enhance the riding experience. An Outpost store employee enters various parameters for the rider (weight, age, gender, etc) as well as the desired training course into the computer so the only control the rider will have is how fast he pedals during the 30-minute session.

The main goal of indoor cycling at Outpost Sports is for the rider to experience a realistic racecourse, and the CompuTrainer system provides this simulation. ElecTrek's design works with the system to keep the same training conditions. The solution is one that allows for variable resistance and one that does not make any modifications to the mechanics of the braking system or to the coding of the software.

The design uses an alternator to convert the mechanical energy of the braking system to electrical energy. This electrical energy will then either be used to power a device such as a fan during the training session through an inverter, or can be stored in a battery as

chemical energy for later use. A major goal is to make the system as non-invasive as possible to prevent voiding the warranty on the CompuTrainer system and altering any of the commercial software. The alternator will be connected to the rotating shaft on the brakes. We will use an alternator because it operates at a speed compatible with the bike and outputs 12 VDC. There will be no additional gears between the alternator and the bike wheels.

The DC voltage will be fed into a battery and be converted to AC with an inverter, so that devices can be powered during the training session. Current flow into the battery will be regulated with a circuit to avoid damaging the battery. Sensors will monitor the current out of the alternator and the voltage of the battery. This power and battery information will be displayed for the user with an LCD screen. The screen will also display the instantaneous power, peak power, average power, and total energy generated.

A microcontroller will control the LCD. It will also interpret the information from the sensors to compute power production and battery charge level. The microcontroller will dynamically control the resistance the brakes apply based on how much power is being drawn from the alternator. This is important because depending on the load, the alternator's internal resistance will change, affecting the overall resistance on the rider.

3 System Description and Block Diagram

The new addition of ElecTrek's system will interface with the existing software and resistance system. We will not alter how the bike is anchored down. The direct contact between the wheel and the brakes will stay the same. The alternator will be attached to the braking system on the rotating shaft on the CompuTrainer system. All other interfaces in the system are wired. Figure 1 shows the contacts between the bike, CompuTrainer, and alternator.

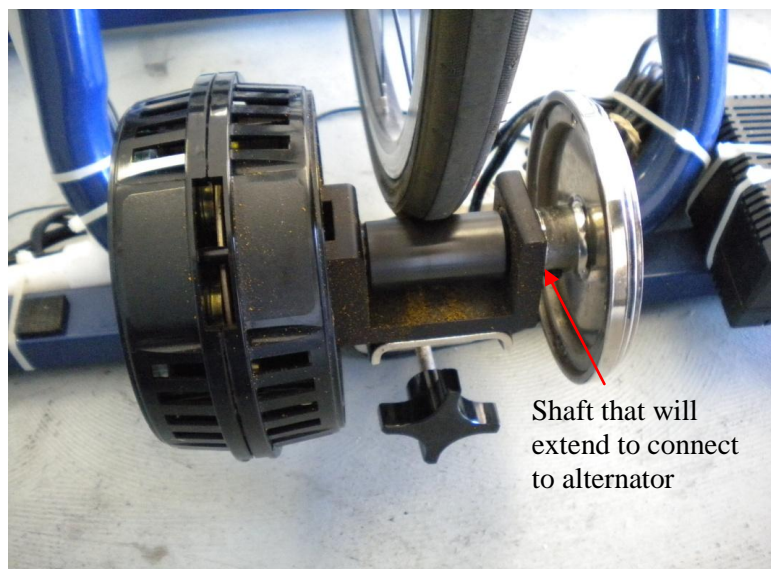


Figure 1. The interface between bike tire and the CompuTrainer system.

The microcontroller is the heart of our design. The microcontroller will intercept the signals from the CompuTrainer software that indicate how much resistance to apply to the wheel. This connection will use the existing cable to the brakes, and the existing cable from the computer, with a DIN-8 connector on each that is wired to the microcontroller. The microcontroller is also wired to the current sensor located after the alternator. All of this information will be displayed on an LCD that is connected to the microcontroller with an I²C interface.

The auxiliary board is the interface between the alternator and the battery. The information from the current sensor will be communicated to the microcontroller via a wired connection. The microcontroller will disconnect the battery from the alternator when the battery is full and will disconnect the battery from the inverter when the battery is depleted. The complete system, with interfaces, is illustrated in Figure 2.

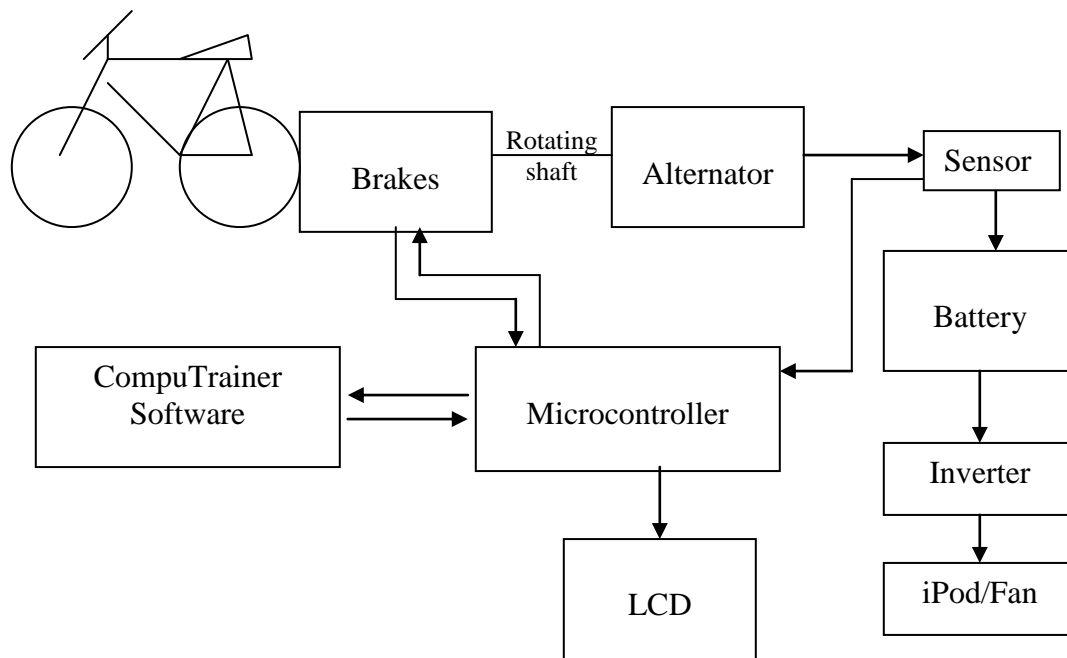


Figure 2. Block system diagram

4 System Requirements

4.1 Overall System

Component	Requirements
Overall System	<ul style="list-style-type: none"> - Take mechanical energy of stationary bike and turn it into usable electrical energy - Added components should not interfere with rider's experience

User Interface	- Should not change from existing user-handheld device or user-personal computer interfacing (i.e. non-invasive)
Size	- Additional components should not interfere with rider's experience
Weight	- Additional components must be able to be easily moved from one set up to another if needed.
Safety	- Tripping hazards may exist – extra wires will be taped down, other potential hazards will be flagged so users are aware - No exposed wires or connections - Battery voltage will be regulated
Power	- Will be powered by battery
Cost	- Additional 120 VAC connection to wall outlet for testing - Must fit within \$500 budget

4.2 Subsystem and Interface Requirements

Component	Requirements
CompuTrainer	- Pre-existing component - Operates essentially as a “black box” - Uses electromagnetic brakes to provide resistance to the rider based on signals received from the software package on the personal computer - Includes PC software package
Alternator	- Generates a 12 VDC output - Increases the mechanical resistance presented to the rider by producing electrical power - Must connect to CompuTrainer shaft with a shaft coupler, that should be grooved for better fit and must be able to accommodate two different diameter sizes, if necessary - Alternator must vary RPMs based on altered resistance signal coming from microcontroller - Alternator must be able to handle at least 1000-3000 RPM and peaks of 400 W (up to 1kW) - Alternator must contain charging circuit, with built-in voltage regulator for 12V, deep-cycle, lead acid battery - Alternator must contain a fan to provide self-cooling - Sensors must monitor the current coming from the alternator so power generation can be determined
CompuTrainer-Alternator Interface (Flywheel)	- Smooth connection joint between CompuTrainer and alternator shafts—needs to compensate for two different diameters - Alternator should rotate freely with CompuTrainer
Charging Circuit	- Included in alternator
Alternator-Sensors	- Monitor the power coming from the alternator by

Interface	measuring the voltage and current at the alternator output
Sensors-Battery-Disconnect	<ul style="list-style-type: none"> - Report these statistics to microcontroller - Once the sensors determine the battery is full, an automatic disconnect needs to prevent current from flowing into the battery
Battery (Lead-Acid)	<ul style="list-style-type: none"> - Operates at 12 VDC - Two functions: charging or powering a load - Must be able to handle several amps of direct current - Must withstand deep cycling and intermittent charging
Enclosure	<ul style="list-style-type: none"> - Moisture resistant, preferably a plastic box (although there could be heat issues) or a grounded metal box (dealing with high currents and voltages, though wires should be insulated)
Inverter	<ul style="list-style-type: none"> - Converts DC into AC to power a load - Convert from 12 VDC to 120 VAC
Load	<ul style="list-style-type: none"> - Variable, e.g. fan, lamp, cell phone charger - Connects via standard wall socket
LCD screen	<ul style="list-style-type: none"> - Runs off 5 VDC power supply - Compatible with I²C interface - Needs enough display space to show the total energy generated; the instantaneous, average, and maximum power generated during the training session; the battery charging level; and whether or not the battery is connected to the system
Microcontroller (μC)Software	<ul style="list-style-type: none"> - Runs multiple functions and interfaces, e.g. intercepting signals from the CompuTrainer software; sending signals to the LCD, CompuTrainer, and alternator; and polling the various voltage and current sensors in the system (see interface descriptions below) - Must have at least 2 analog-to-digital converters
μC-Software Interface	<ul style="list-style-type: none"> - Intercept signals determining required resistance and distribute to CompuTrainer and alternator - Acquire statistics (e.g. power readings) from software so it can be displayed on the LCD screen
μC-LCD Interface	<ul style="list-style-type: none"> - μC sends signal to LCD telling it what statistics to display
μC- CompuTrainer Interface	<ul style="list-style-type: none"> - After μC intercepts signals from software which indicate the required resistance, μC apportions part of that resistance to the CompuTrainer by sending a signal to it
μC-Alternator Interface	<ul style="list-style-type: none"> - Use voltage and current readings from sensors to determine power being produced by alternator and then the resistance the alternator is providing to the rider

4.3 Future Enhancement Requirements

Component	Requirements
CompuTrainer/Alternator Interface	<ul style="list-style-type: none">- Will be one system instead of two integrated components. The system will use purely electrical resistance from an applied load to make the alternator harder to spin, thereby generating mechanical resistance. This would make for a more efficient combined training and energy capture system.
Energy Capture Subsystem (Alternator-Battery-Load)	<ul style="list-style-type: none">- Will include a resistive element that dissipates heat generated by the system to the outside rather than in the building- Must be used with the single CompuTrainer/alternator system mentioned above- Charge multiple batteries in parallel

5 Low Level Decisions

5.1 Alternator / Current Sensor Subsystem

Alternator

The alternator being used is a Toyota Celica 2005 replacement alternator, part number REPT330125 on www.car-stuff.com, and it costs \$58.94. The shaft of the alternator will connect through a shaft coupler to the shaft of the CompuTrainer resistance device. (This will require removing the flywheel from the CompuTrainer.) The shaft coupler will have to connect the 0.75 in CompuTrainer shaft to the 0.5 mm alternator shaft. (Shaft couplers can do English-metric couplings.) This means the alternator shaft will rotate with the CompuTrainer at the same rpm. As the alternator rotates, it generates electricity, which will be fed through the alternator's built-in voltage regulator and charging circuit to the 12V, deep-cycle, lead acid battery. This should be a straight-forward connection given that this is the same electrical set-up as in a car. The primary difference is that the set-up for this system with the CompuTrainer will not use an ignition switch (Figure 5). Instead, the field #1 connection (Figures 3, 4) will be wired to the positive terminal of the battery through a set of resistors. The field #1 terminal connects the battery to the field coil and thus provides the initial field current to run the alternator.

The charge on the battery will kick start the alternator by providing the initial field current. This will allow the alternator to get up to its stable, desired rpm range and provide an output voltage of around 14.4V, which will charge the battery. The voltage regulator controls this output voltage. Once the alternator speed levels out, the diode trio will take over providing the field current, leaving the alternator free to charge the battery.

This setup will use the ground, battery, field #1 and sensing #2 connections pins (Figures 3, 4). The field #1 pin connects the positive terminal of the battery to the diode trio, which is the self-sustaining source of the field current for the alternator. The sensing #2 connection ties the positive terminal of the battery to the voltage regulator to ensure that the voltage going to the battery is within the limits of what the battery can handle. The battery pin on the alternator also connects to the positive terminal of the battery. The metal casing of the alternator acts as ground and is connected to the negative terminal of the battery.

Two critical components of the field #1 pin-to-battery connection are having both a switch and resistive elements in between the two terminals. It is critical that a switch is in place between the field #1 pin and the positive battery terminal so that the battery can be disconnected from the field pin when the alternator is not in use. If this is not done, the field coil of the alternator will continue to draw charge from the battery and drain it, even if the alternator is not running. Furthermore, merely shorting these terminals causes a current to flow through the field coil that is too high for this component to handle and can damage the alternator. This design will not use the typical alternator warning light as the resistive element, in part because the alternator will be running at lower-than-average rpms. (Typical alternator rpms are in the 2000-10000 rpm range, whereas this system will run at 1000-3000 rpm.) The alternator warning light could provide too much resistance. One source recommended using a parallel combination of 10W, 22-Ohm resistors. It is key to note that the lower the desired \ alternator rpms, the higher the initial current the battery must provide to the field #1 pin.

The testing plan for the alternator consists of connecting a 1 HP motor to the shaft of the Delco-Remy alternator by means of a pulley. The alternator will also be wired to the battery. A tachometer will be used to ensure the motor is spinning at least 1000 rpm. Once the alternator is running at the same rpm as the motor (determined again by the tachometer), the voltage across the battery terminals and the current going into the battery will be measured with a multimeter, just to get baseline readings. Readings will be taken at a variety of rpms. In addition, an inverter will be attached to the battery so that loads with large power draws (e.g. a fan) can be connected to test their effect on alternator operation and fluctuations in voltage and current in the alternator/battery setup.

Alternator Diagrams

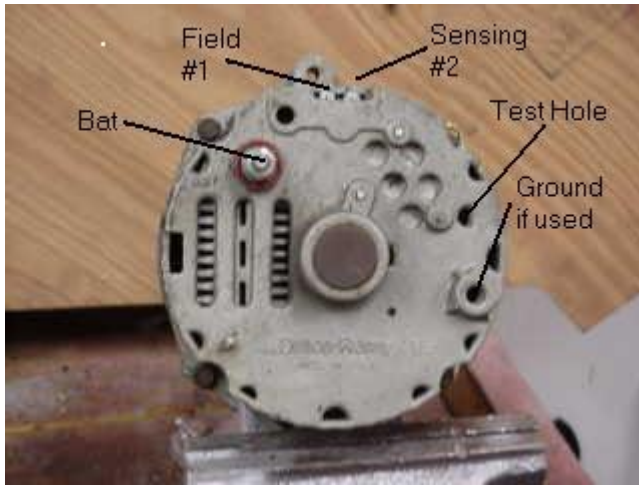


Figure 3. Alternator electrical pin diagram.

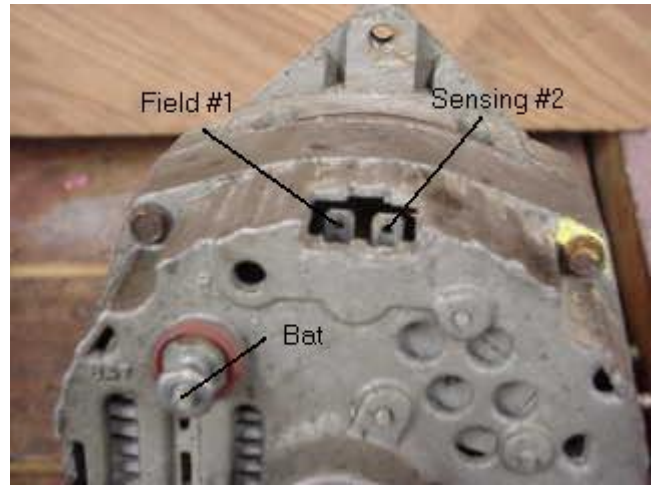


Figure 4. Close-up of field and sensing pins.

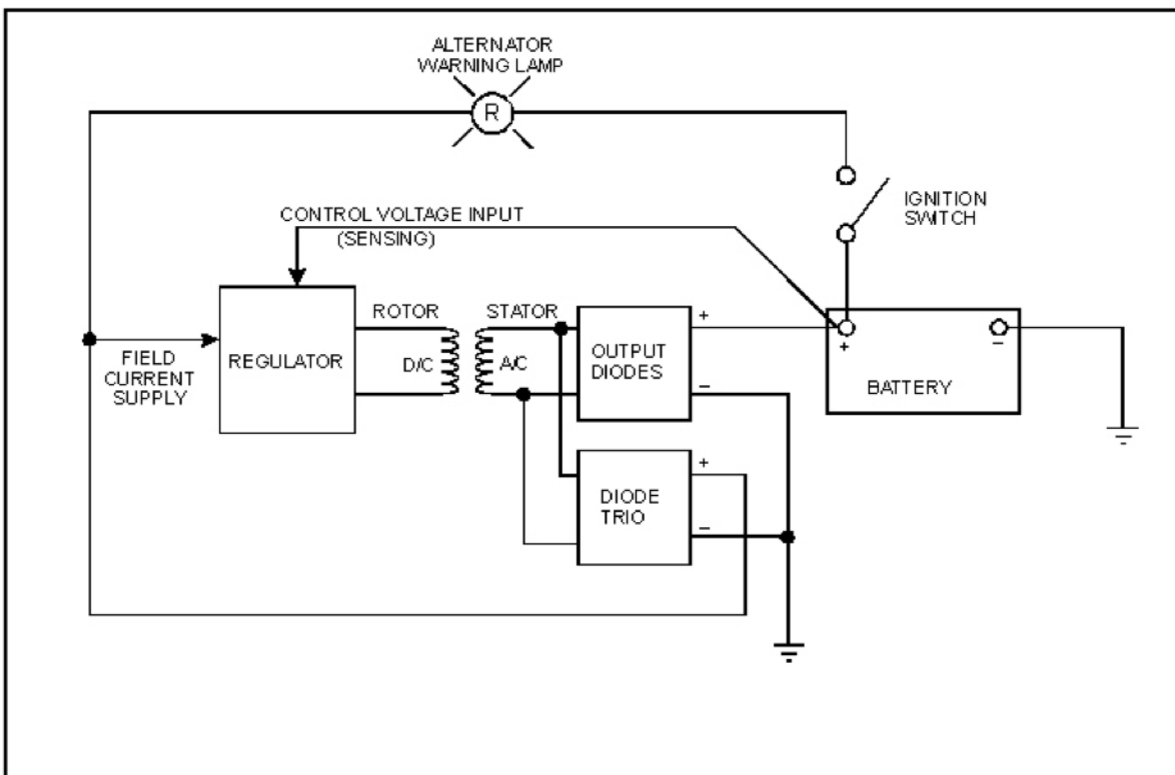


Figure 5. Alternator functional diagram.

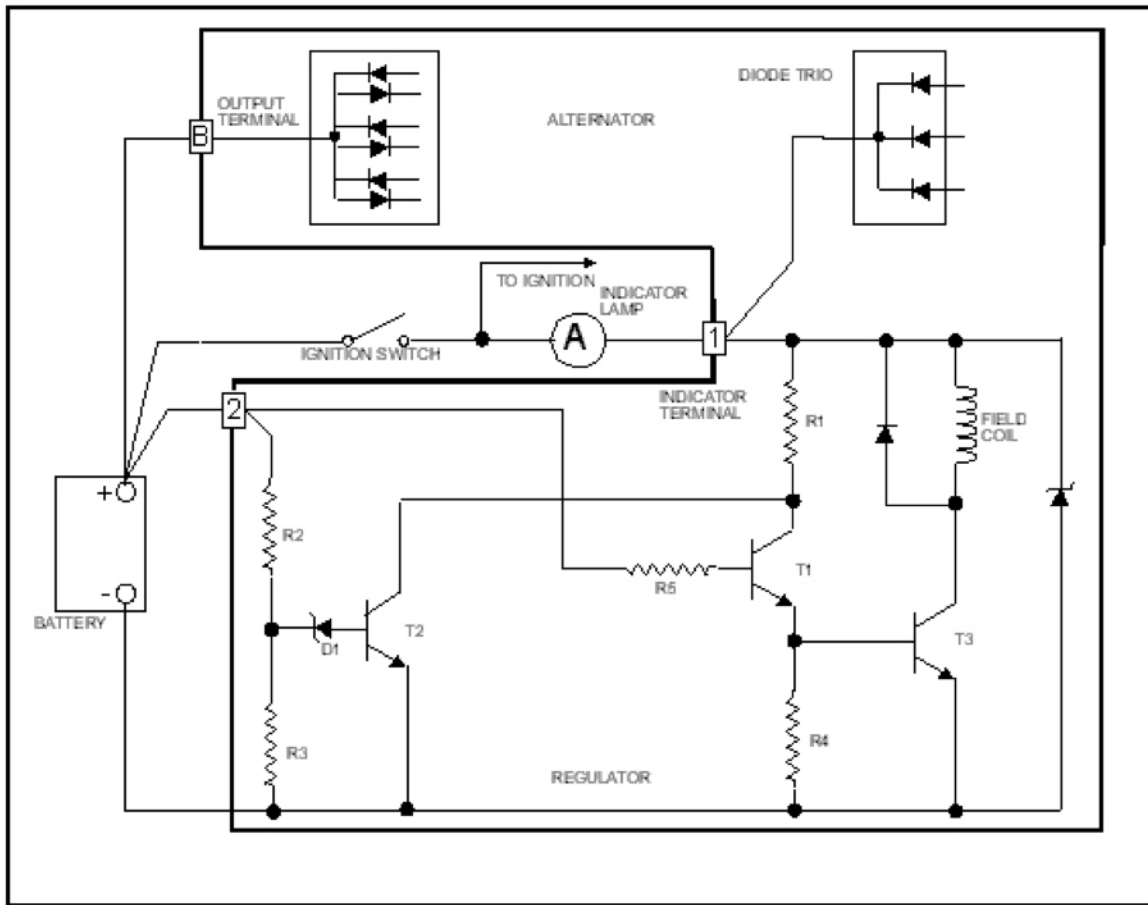


Figure 6. Alternator circuitry.

Current sensor

This design will use a current sensor from Allegro Microsystems rated for 20A. The part number on Digikey is 620-1226-1-ND, and the unit price is \$4.73. The datasheet can be found at http://www.allegromicro.com/en/Products/Part_Numbers/0714/0714.pdf. This current sensor will be employed between the alternator and the positive terminal of the battery so that the current flowing into the battery can be monitored, and so the output power of the alternator can be determined.

5.2 Battery/Inverter

The main hardware components of this subsystem are the 12V deep cycle lead acid battery and the inverter. This subsystem also includes the MOSFET control circuit and the current sensor on an auxiliary printed circuit board. The two main signals of this subsystem between the battery and inverter are positive and ground. The battery and the inverter will be connected in parallel in the final system. See Figure 7 for a block diagram of this subsystem and Figure 8 for a functional block diagram of the auxiliary board.

The inverter that matches our specifications is the AIMS PWRI30012S. The input voltage to this inverter is rated for 10-15V. It is also rated for 600W for power surges and 300W for continuous operation. This inverter has a pure sine wave output waveform. There are two options for inputting the signal into the inverter: alligator clips or an

automobile cigarette lighter plug. The inverter has two types of output connectors: two three-prong plugs and a USB port. The battery that matches our specifications is the Sportsman Marine Battery DC27. This 12V deep cycle lead acid battery is rated for 105 AH.

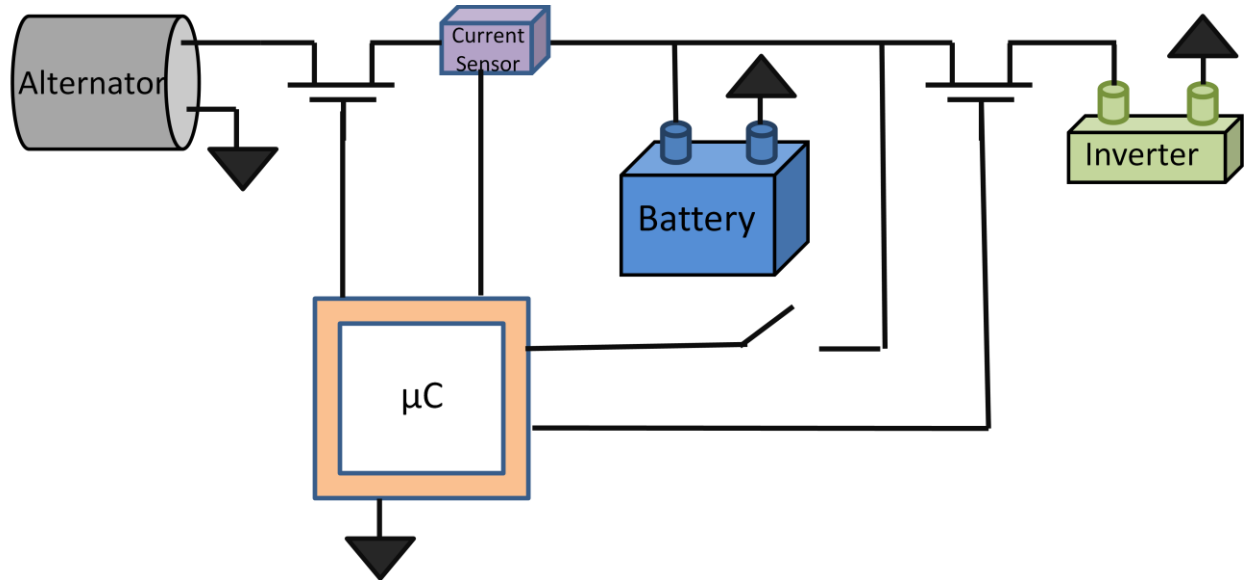


Figure 7. Subsystem Block Diagram.

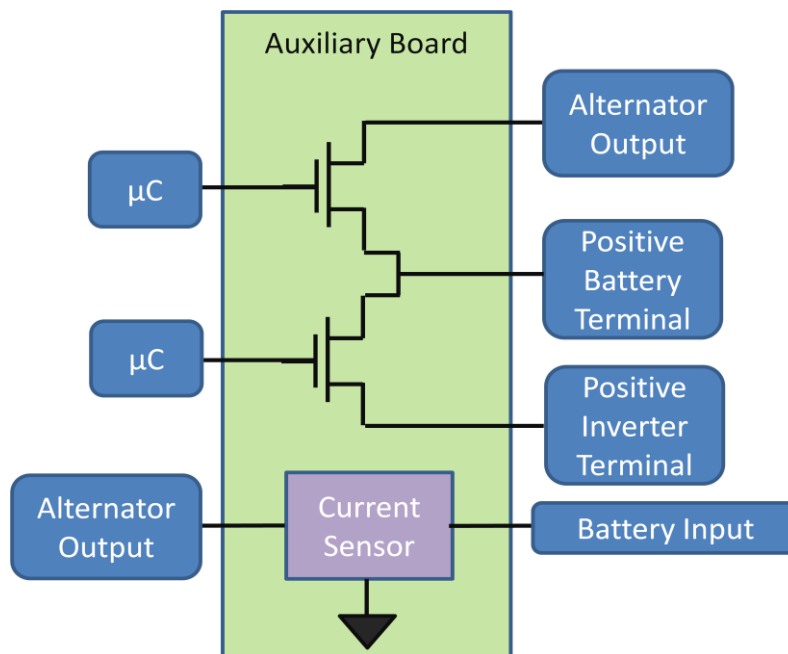


Figure 8. Functional Block Diagram of Auxiliary Board.

The auxiliary board will have connections between the alternator, the battery, and the inverter. Two MOSFETs will be on the auxiliary PCB and will be connected to the microcontroller on the main board to maintain a common ground. The auxiliary board will also contain a current sensor. This sensor will allow the microcontroller to monitor

the current feeding into the battery. Ribbon cables will be used for the signals from the auxiliary board to the microcontroller. This connection method requires ports on each of the two boards and the ribbon cables themselves.

The MOSFET that matches our specifications is the Infineon Technologies BSZ067N06LS3G. This part is available from DigiKey under the part number BSZ067N06LS3GINCT-ND. It is rated for a maximum drain to source voltage of 60V and a maximum power of 69W.

Subsystem Interfaces

The battery and the inverter will be connected by wires. The terminals of both the battery and inverter have a screw and nut. A wire tab can be placed over the screw and the nut can be used to tighten the wire against the screw. This will provide a secure mechanical and electrical connection. Precautions will be taken to prevent someone from touching the bare terminals. The chosen inverter comes two cables with alligator clips and an automobile cigarette lighter plug. One method of connection uses alligator clips internal to the inverter to clip on to the battery terminals. Another option is to use the automobile cigarette lighter plug and a mating socket. Since the inverter is originally designed for use in a car, one of its input connection options is a cigarette lighter plug. A mating socket can be wired to connect the inverter with the battery terminals. It is also possible to use other cable with a wire tab connector to connect to the inverter instead. The output interfaces of the inverter are standard. There are two standard 3-prong 120 VAC plugs and a USB port. These can be easily used to discharge the battery and connect common devices, such as fans or iPods.

The input terminals of the battery will be connected to the output of the alternator. At the output of the alternator, there will be a current sensor to keep track of the current flowing into the battery. The current sensor will be read by the microcontroller on the main board.

All interfaces in the subsystem are standard. The auxiliary board port connecting the alternator output and the board input and the port connecting the board output and the battery terminals will be a standard port. The same is true for the port connecting the auxiliary board and the main board. The battery terminals and the inverter will be placed in parallel using the connections provided with the inverter.

We will monitor the battery status by the terminal voltage. Table 1 can be used to equate terminal voltage with battery charge.

|

Table 1. Relation between Terminal Voltage and Battery Charge.

State of Charge	12 Volt battery	Volts per Cell
100%	12.7	2.12
90%	12.5	2.08
80%	12.42	2.07
70%	12.32	2.05
60%	12.20	2.03
50%	12.06	2.01
40%	11.9	1.98
30%	11.75	1.96
20%	11.58	1.93
10%	11.31	1.89
0	10.5	1.75

Source: http://www.solarnavigator.net/battery_charging.htm

To ensure that the battery is accepting charges and discharges, the terminal voltage can be periodically measured during charging or discharging. The change in terminal voltage will indicate the desired battery behavior. In the final system, the current entering the battery from the alternator will be constantly monitored by the current sensor. The microcontroller will regularly check the current flowing into the battery and the terminal voltage. If the current is too high, the microcontroller will control the MOSFET between the alternator and battery so that the current going into the battery is reduced. By keeping track of terminal voltage, the microcontroller will also track the charge level of the battery. The microcontroller will use the second MOSFET to disconnect the battery and inverter and prevent further discharge of the battery if the battery voltage is too low. These behaviors can be tested using the microcontroller software. The inputs into the microcontroller, such as the current going into the battery and the terminal voltage, can be simulated, and the software of the microcontroller can be tested to see if the microcontroller set the correct control signals to the MOSFET.

There is not much testing that will be required on the inverter. It essentially operates as a black box in this system, and it will be purchased from the manufacturer to perform a specific task: take input of 12-14VDC from the alternator and output a constant 120VAC. The output ports of the inverter can be tested with an oscilloscope to ensure a standard 120VAC, 60 Hz signal. The USB port on the inverter can also be tested with an oscilloscope to check the expected 5VDC output. If either of these output signals are not correct, then the input signals need to be checked to make sure that they conform to the ratings specified by the inverter manufacturer. Checking the input signals is the main troubleshooting option for the inverter.

The power MOSFET circuit has more options for testing. The output current and voltage from the alternator should be tested to make sure that they are within the ratings of the MOSFETs and the other components on the board. The voltage reading and the current sensor output can be verified using a multimeter. One important scenario to test occurs when the battery charge drops below 30%. This can be achieved by discharging the battery using a high power consumption device, such as a soldering iron. Then, the battery can be connected to the system. If the entire system is functioning correctly, the microcontroller should read the terminal voltage of the battery and recognize that it is below the desired threshold. The microcontroller should send a signal to the auxiliary board, and the second MOSFET should completely disconnect the inverter and the battery. This means that the devices connected to the battery will not be allowed to discharge the battery below a certain level to extend the lifetime of the battery.

Another important situation to test occurs when the rider experience low-resistance courses when the battery voltage is low. For example, when a rider is coasting down a steep hill, there is little to no resistance felt by the rider when he is pedaling. When the battery voltage is low, the current coming from the alternator into the battery will be high. This means that the resistance provided by the alternator will also be high. In this situation, the resistance provided by the alternator is larger than the resistance requested by the CompuTrainer software. Therefore, it is necessary to have control over the amount of current flowing into the battery. This will ensure the correct resistance is seen by the rider and will also help protect the battery from dangerous charging conditions. The microcontroller should recognize this situation and direct the first MOSFET to limit the current going into the battery. The auxiliary board will reduce the current going into the battery to 1-2 Amps. This will reduce the resistance of the alternator and, in turn, the resistance seen by the rider. The current going into the battery will only be limited when the battery voltage is lower than a certain threshold, or when the current going into the battery is too high. In other situations, the charging circuit built into the alternator will adequately control the current going into the battery. This behavior can be tested by connecting a discharged battery, selecting a low-resistance riding course, monitoring the microcontroller software signals, and verifying the expected behavior of the MOSFET.

5.3 Microcontroller and Software

The microcontroller we will use will be in the PIC18F722 family. It was chosen because of its unique capture pulse width modulation module and it fulfills the I/O and memory requirements of the project. The capture PWM module will be used to interpret the signal coming in from the CompuTrainer. It also has a second PWM module to output the signal to the braking device after it has been altered by our software. It also has an analog to digital converter. One pin of the converter will be used to interpret the input from the current sensor and the other to interpret the voltage sensor so we can calculate the power coming from the alternator. An LCD screen can be connected to it through an I²C interface. Memory is not a major issue with our microcontroller, as we are not running many different complicated programs on it. It will be run on a regulated 5 V supply that will come from the battery, which ensures that the battery can always be

monitored. The microcontroller will be connected to a current sensor, an LCD, a power MOSFET, as well as having an input which polls the voltage on the battery. The microcontroller is shown in Figure 9.

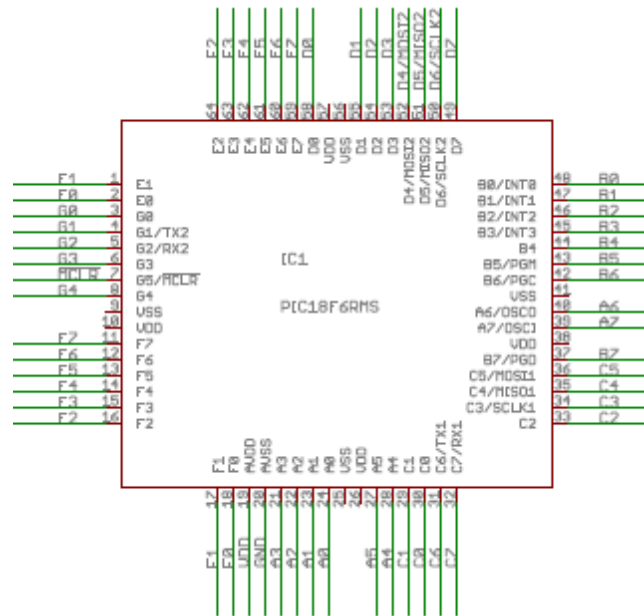


Figure 9. Microcontroller schematic.

The microcontroller will take regular readings of the voltage and current produced by the alternator and calculate the electric power being generated. This information will be displayed on the LCD and used to determine how much resistance the alternator is producing. Additionally, the voltage readings of the battery will be used to determine its charge level. The charge level will be displayed on the LCD and monitored to ensure that the battery is neither overcharged nor depleted. If the battery is being overcharged, the microcontroller will electrically disconnect the battery from the alternator, and if the battery is too depleted, the microcontroller will disconnect the battery from the inverter.

The CompuTrainer will be connected to the microcontroller, and the RPM and resistance signals will be constantly read. The RPM information will be used in calculating the mechanical resistance of the alternator. The resistance signal specifies the total amount of mechanical resistance needed, so once the alternator's resistance is subtracted from this value, a new resistance signal will be generated to operate the CompuTrainer's brakes. Total resistance will be produced by a combination of brakes and alternator. The resistance signal is a PWM signal at 3.58 kHz. Our microcontroller has a hardware PWM module that will be used to both read the CompuTrainer's signal and then generate our own with which we will control the brake.

```
while(true)
{
    // Get Voltage and Current Readings
    voltage = readVoltage();
```



```

current = readCurrent();
power = voltage*current;

LCDDisplay("Instantaneous Power: "+power+" W);

// Determine Battery Charge Level
charge = getBatteryCharge(voltage);
LCDDisplay("Battery Power: " + charge + "%");

// Manage Battery Power
if(charge > 99)
{
    disconnectAlternator();
    connectInverter();
} else if(charge < 30) {
    connectAlternator();
    disconnectInverter();
} else {
    connectAlternator();
    connectInverter();
}

// Set Resistance
speed = readSpeed();
load = readResistanceSignal();

altResist = getAltResistance(speed,power);

changeBrakeSignal(load-altResist);
}

// User Functions:

// Writes Text to the LCD
void LCDDisplay(string text);

// Reads battery voltage from the A/D converter
double readVoltage();

// Reads current into battery from the A/D converter
double readCurrent();

// Computes battery charge based on the its voltage
int getBatteryCharge(double voltage);

```

```

// Sets output pin low to turn off power MOSFET on alternator
void disconnectAlternator();

// Sets output pin high to turn on power MOSFET on alternator
void connectAlternator();

// Sets output pin low to turn off power MOSFET on inverter
void disconnectInverter();

// Sets output pin high to turn on power MOSFET on inverter
void connectInverter();

// Computes speed based on time between pulses of the CompuTrainer
// speed signal
double readSpeed();

// Uses the PWM module to find the duty cycle of resistance signal
int readResistanceSignal();

// Calculate mechanical resistance of alternator based on its speed
// and electrical power output
int getAltResistance(double speed, double power);

// Generate a PWM signal to control the brake, with the specified
// duty cycle
void changeBrakeSignal(int resistance);

```

To test the microcontroller and board functions we will use our current kit board. We will need to reprogram it so the LCD and PWM modules can be used at the same time and may need to create a separate analog to digital converter. We will test how much power the board takes from its source so we know how much will be needed to turn it on when the battery is low on charge. We will use the current LEDs on the board and current LCD screen to test the user interface.

We will need to verify that all our sensors are taking proper readings of voltage and current. To test this, we will be running the alternator with a motor and charging the battery. The readings of microcontroller will then be compared to the readings of a DMM to ensure they are accurate.

We also need to verify that the system is properly reading and transmitting the signals that interface with the CompuTrainer. The CompuTrainer allows the user to set the load directly, so we will be able to generate resistance signals and then check to see if the microcontroller is reading the value correctly. The resistance signal that we generate will be captured with the USBee logic analyzer to ensure it is correct. We will test the RPM

signal by comparing our microcontroller reading to the actual spin rate of the CompuTrainer using a tachometer.

Board Design

The board will be powered by the battery and will be regulated to 5 V by the LT1763CS8-5#PBF voltage regulator made by Linear Technology. It can take an input voltage of up to 20 V and will output a fixed 5 Volts at 500 mA. The voltage regulator is shown in Figure 10. There will also be an additional connection to power the board from a wall socket and a switch that can switch the power supply from the wall or the battery. This can be used for testing, and to power the system if the battery charge level were to get too low. Ideally, however, using a power MOSFET, this should never happen once the system has been implemented fully. The programmer illustrated in Figure 11 (created by Mike Schafer) will have a reset button for any issues that arise during use.

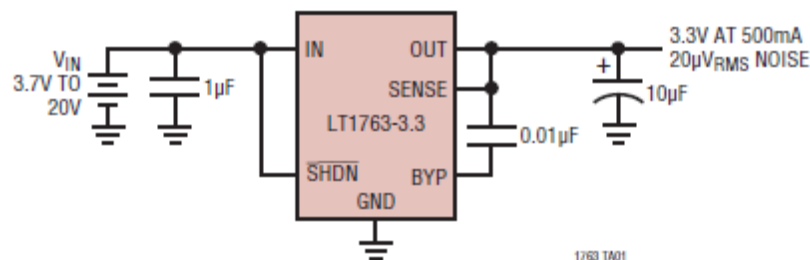


Figure 10. Voltage regulator schematic.

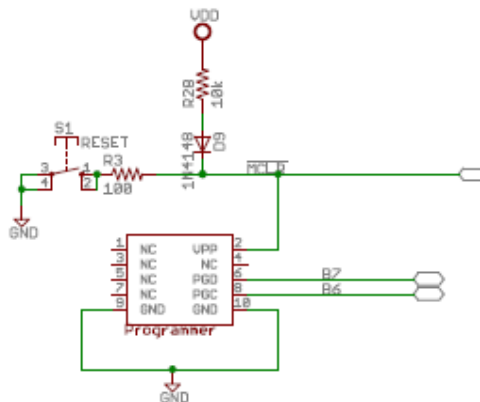


Figure 11. Programmer Schematic.

LCD

A 4 line, 20 character per line, LCD will be used. It is manufactured by Newhaven Display International and uses an I²C interface. The LCD will also run at 5 V. The battery charge level will always be displayed when the battery is in use. When the entire system is on, the total energy generated will be displayed as well. The instantaneous,

maximum power, and average power will cycle through on the remaining lines of the screen. Figure 12 shows the LCD schematic of the ports on the board.

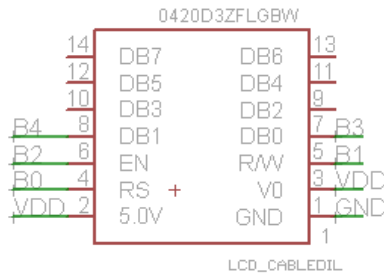


Figure 12. LCD pin diagram.

Pins will be connected to one port on the microcontroller to connect the auxiliary board. A ribbon cable will interface between the two and will have a signal for each of the gates on each of the power MOSFETS, a 5 V supply, ground, and the voltage signal from the current sensor that will be an input to the analog to digital converter. On the same port, we will take a voltage reading on the connection near the current sensor to also be an input to one of the channels on the A/D converter.

The complete board schematic is attached as the final page of this document.

6 Major Component Costs

Tables 2, 3, and 4 below give an estimated cost breakdown of our project. Our most expensive items are the alternator, battery, and printed circuit board. Other components will be required to build the charging circuit and to populate the boards, but these costs are minimal.

Table 2. Power Generation Subsystem Costs

Component	Cost	Source
CompuTrainer system with software	\$0.00	Outpost Sports
Car Alternator	\$58.94	www.Car-Stuff.com
Current sensor on generator output (620-1226-1-ND)	\$4.73	DigiKey (Allegro Microsystems)
Subsystem Total:	\$63.67	

Table 3. Battery Subsystem Costs

Component	Cost	Source
12V Deep Cycle Lead Acid Battery	\$107.63	www.Defender.com
Inverter	\$99.00	www.TheInverterStore.com
Power MOSFETs (2)	\$1.35	DigiKey (Infineon)

		Technologies)
Subsystem Total:	\$209.33	

Table 4. Circuit Board Subsystem Costs

Component	Cost	Source
Printed Circuit Board	\$50.00	Advanced Circuits
Microcontroller	\$12.54	DigiKey
LCD, NHD-0420D3Z-FL-GBW	\$24.90	DigiKey
Voltage Regulator, LT1763CS8-5#PBF	\$4.13	DigiKey
Ribbon Cable	\$0	Supply closet
Subsystem Total:	\$79.03	
Project Total:	\$352.03	

7 Conclusions

ElecTrek’s design for Outpost Sports’ bicycle training facility will allow the owner to capture the energy created by the riders into a useful form. The design will generate electricity without altering the training experience and will not alter the braking system or the existing software. The primary components will be an alternator, battery, inverter, sensors, and a microcontroller.

The primary issues with the design involve dynamically changing the resistance on the braking system to make up for the resistance added by the alternator. These will be resolved throughout the design process. We anticipate that working out each question with the design will be challenging, but that we will be able to complete the project successfully.

8 References

Current sensor datasheet:

http://www.allegromicro.com/en/Products/Part_Numbers/0714/0714.pdf

Voltage Regulator datasheet:

<http://cds.linear.com/docs/Datasheet/1763fg.pdf>

LCD datasheet:

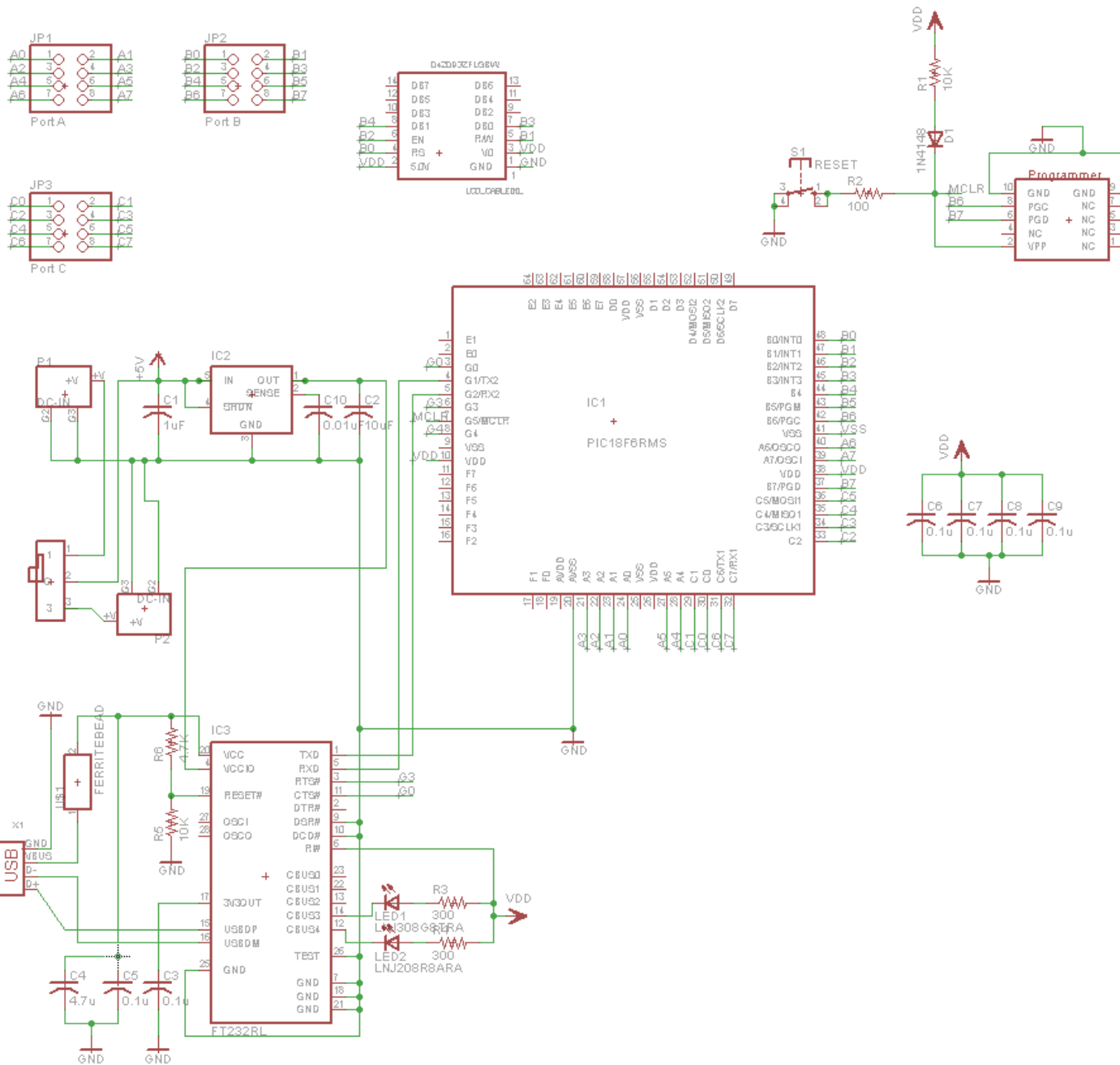
<http://www.newhavendisplay.com/specs/NHD-0420D3Z-FL-GBW.pdf>

Power MOSFET datasheet:

http://www.infineon.com/dgdl/BSZ067N06LS3_rev2.0.pdf?folderId=db3a30431ddc9372011ebafa04517f8b&fileId=db3a30431ddc9372011ebb0f41537ff6

Microcontroller datasheet:

<https://wiki.nd.edu/download/attachments/24546033/18F8722.pdf?version=1&modificationDate=1285085744000>



Complete Board Schematic