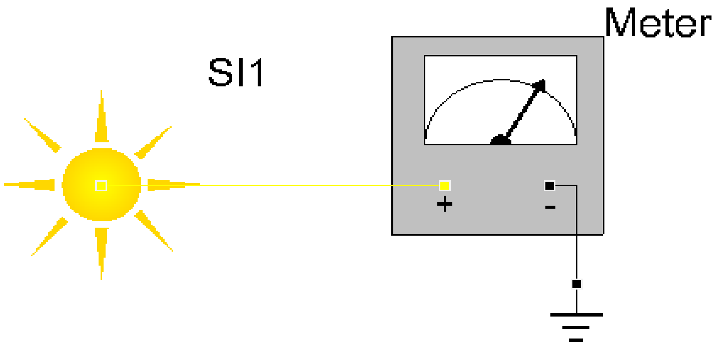
System Simulations

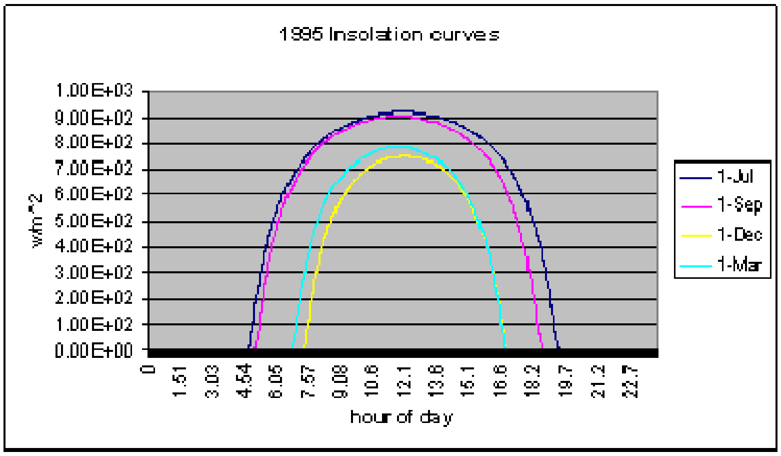
Following the PDR presentation, it became obvious we needed away to better assess our design decisions and test whether they were feasible. In the following system simulations the key components of the project design are addressed. The software used is Virtual Test Bed (VTB), developed by the University of South Carolina. This software offers a wide range of applicable models including solar panels, batteries, buck converters, and insolation.

1. Insolation

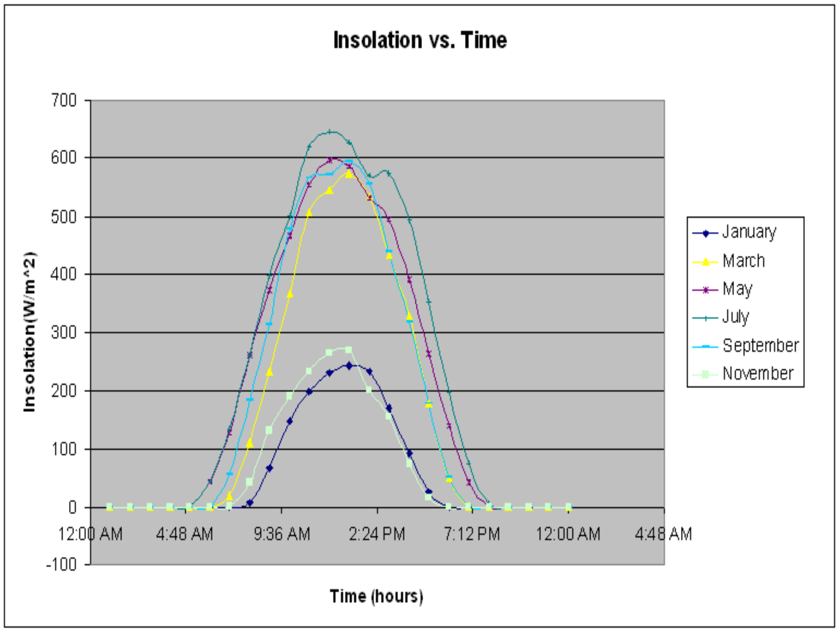
Understanding the input to our system was a key part in creating the design solution. VTB contains Insolation. The model allows inputs for the location in the northern hemisphere and specified time and date. The model is shown below.



This model was used to see how much sunlight would hit our arrays throughout the year. It was determined that the model month index was incorrect and off by half a year. After the model was correct we generated these curves which should the insolation for March, july, September, and December 1st.



As expected there is more sunlight in July and September than March and December. Furthermore, it shows how the insolation varies throughout the day. Our system can expect to see roughly 700-900 W/m^2 during mid day, but less than 1 W/ m^2 during the night and early morning. The VTB data was verified with actual insolation data from the National Solar Radiation Database (NSRD). In the graph below, the insolation curves are given averaging the insolation at that time of day for the entire month as recorded by NSRD.



The NSRD data appears to be much lower than the VTB data, but that is because the NSRD averages all the data at a certain time of day for the month. Since, the peak of the insolation curve is shifting over the month, the average is lower than looking at a single day as was done with VTB.

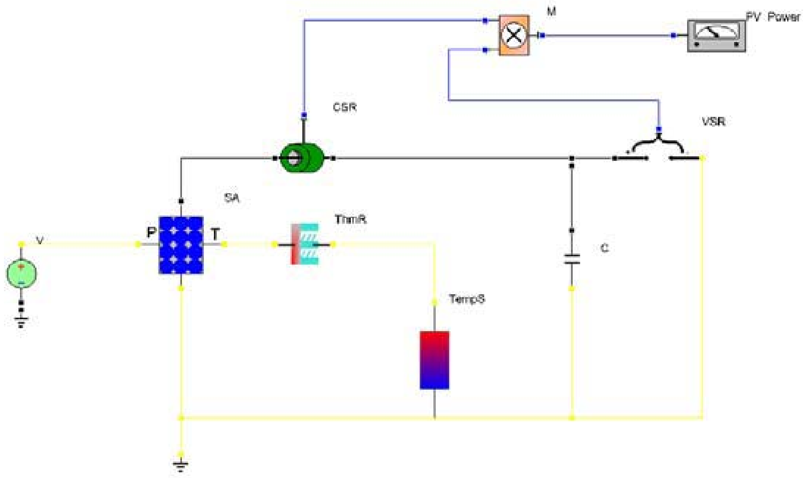
The table below gives some summary facts to the two set of data and show that they correlate to a high degree of the course of a year.

Data Source	Yearly Insolation Average(W/m^2)	Maximum Insolation (W/m^2)
VTB	325	919
NSRD	326	978

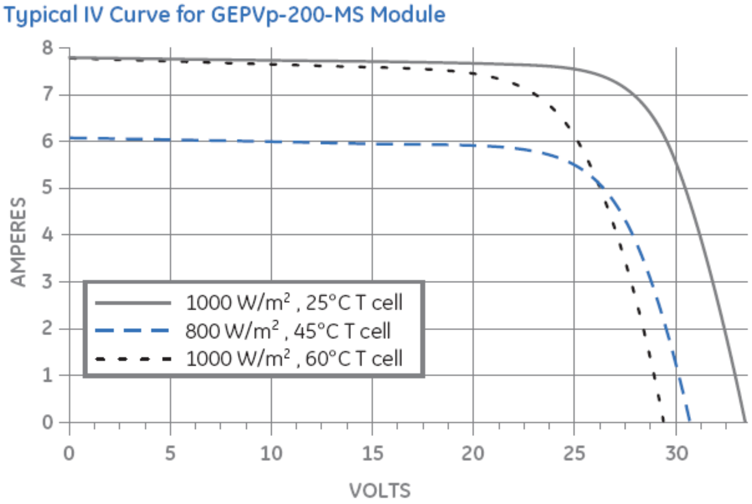
This study provided confidence that VTB and the insolation model could be used for further modeling. It also better defined the input to our system.

2. Panel Model

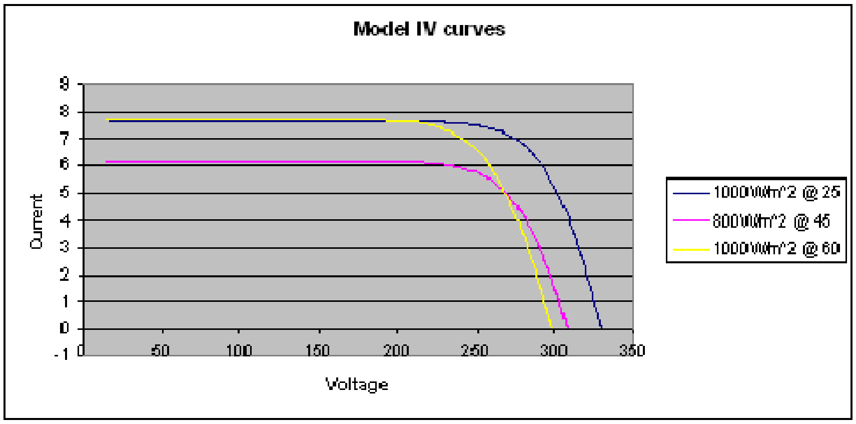
The next step was to determine how much power would come out of our solar arrays given the insolation found in earlier simulations. The output voltage and current vary greatly based on the temperature and insolation conditions. To better understand this interaction, the VTB model of the solar panels shown below was created.



In this model, the capacitor starts at zero volts, but then the charge created by the solar panels begins to build up and the voltage grows on the capacitor, which also changes the current coming out of the array. In this way the time-based VTB simulation could generate I-V curves. The goal of this model was to match the I-V specifications given by General Electric for the GEPVp-200-MS solar panels shown below.



Through trial and error an accurate model of the solar panels was developed. The I-V characteristics for the VTB model are shown below. The same conditions are tested in VTB as presented in the GE specifications. The voltages in the VTB model are 10 times the GE specification to simulate the 10 of the GE modules in series on the roof.



Once the VTB solar panel model was verified analysis, important data was extracted about the operating point of the solar panels. The system can fix either a voltage or current of the solar panel. Then based on the temperature and insolation a resulting current or voltage comes out of the array based on the I-V curves. It is desired to operate in the knee of the I-V curves to maximize power. Coming out of PDR the group was still unsure if the system could run on a fixed voltage operating or whether Maximum Power Point Tracking (MPPT) was needed.

VTB was used to simulate many different insolation values at a variety of temperatures. These simulations showed that under normal environmental conditions for Easton, PA, the maximum power point operating voltage could change anywhere in the range of 230-305v. A sample table is shown below for 32^{0}C.

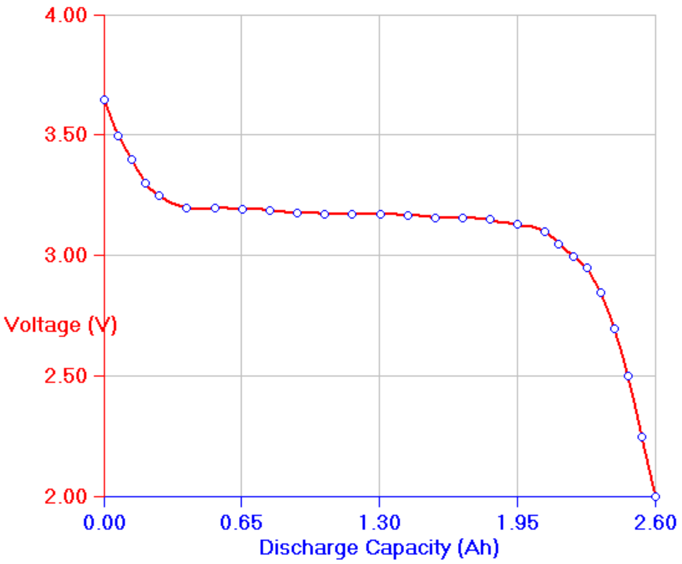
Insolation (W/m^2)	Temperature (K)	Current (A)	Voltage (V)	Power (W)
1000	305.4	7.65	255	1950.75
950	305.4	7.25	258	1870.5
900	305.4	6.87	256	1758.72
850	305.4	6.51	254	1653.54
800	305.4	6.11	257	1570.27
750	305.4	5.73	257	1472.61
700	305.4	5.36	257	1377.52
650	305.4	4.98	257	1279.86
600	305.4	4.6	256	1177.6
550	305.4	4.21	256	1077.76
500	305.4	3.82	257	981.74
450	305.4	3.43	255	874.65
400	305.4	3.05	254	774.7
350	305.4	2.68	253	678.04
300	305.4	2.29	253	579.37
250	305.4	1.91	251	479.41
200	305.4	1.53	248	379.44
150	305.4	1.15	244	280.6
100	305.4	0.76	240	182.4
50	305.4	0.38	231	87.78
0	305.4	0	0	0

In addition, a comparison was made between a system which used a MPPT algorithm against a system that used a fixed operating voltage. It was determined that the ideal fixed operating voltage system could lose up to 12% of the possible power just based off its operating point. This would be a 12% loss before the power even reached the electronics of the system. The project team decided that it was worth adding the complexity of an MPPT algorithm to ensure our solar panels were producing as much power as possible.

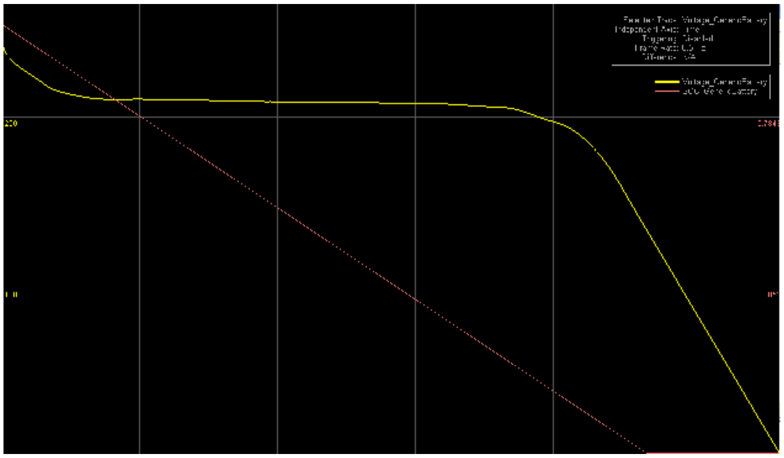
The solar panel also provided a rough estimate for how much power we could expect to see coming out of the solar panels throughout the year. By using the average temperature in Easton for a specific month and the expected insolation, we could determine how much power would be produced by a MPPT system throughout the day. By integrating this value over the course of the day, the average power produce per day was calculated for each month. This calculation found the least amount of energy produced is in December, where about 12 kWh are produced daily. The month of July produced the most energy with an average of 21.7 kWh per day.

3. Battery Model

In an effort to start combining pieces of the system design in a larger more complex and accurate model of the system, a model for the batteries was created. There battery supplier didn’t not have a thorough data sheet that could be used to create a model. In fact the charging and discharging chrematistics of the batteries will most likely have to be generated experimentally. For a system level simulation however, a generic LiFePO4 battery model was created. The graph below shows the discharge characteristics of one cell of the LiFePO4 batteries.



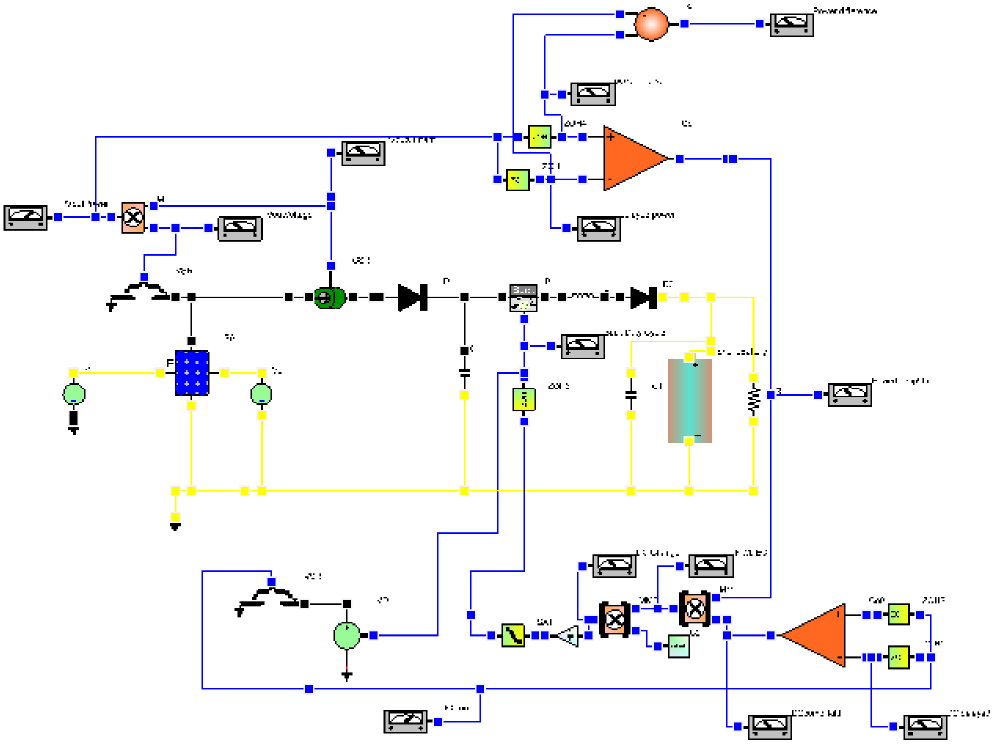
Our system will contain 4 parallel strings of 64 of these cells in series. The simulation below shows the entire battery pack discharging into a load drawing a constant power. The yellow line is the battery voltage and the red line is the state of charge of the batteries (0-1).



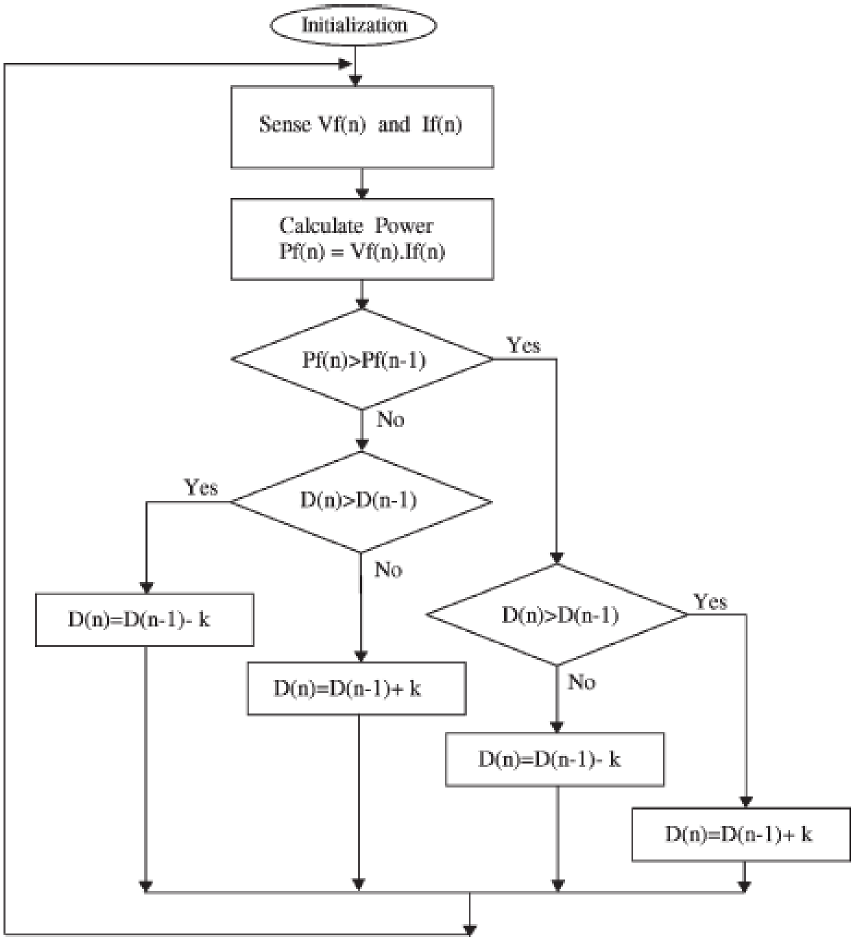
Developing a model for the battery was important in combing components into a system level design and seeing how the components interact.

4. MPPT

After PDR the MPPT algorithm was a big uncertainty. How would such an algorithm work? How complex would the algorithm be to implement? How much confidence is there that our group could implement such an algorithm? These were questions that had to be answered before our design was finalized. Through VTB, a model was created to include a control algorithm to make the solar panels operate at the maximum operating voltage.



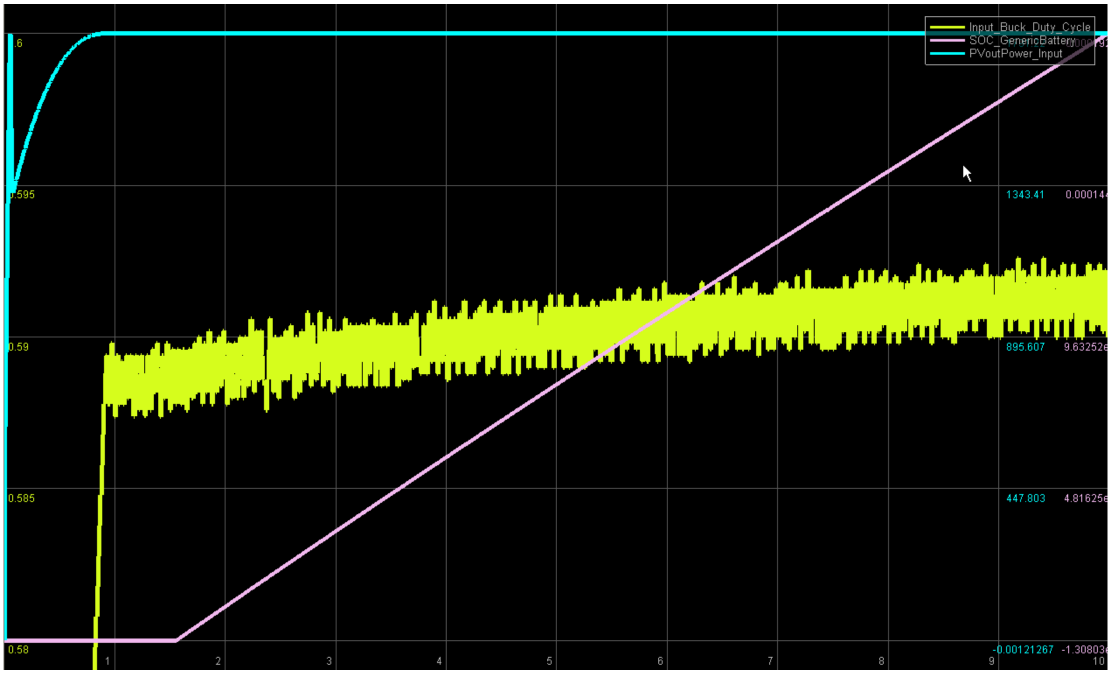
This model included the insolation model, solar panel model, battery model, and a buck converter. The buck converter controls the ratio of its input and output voltage by changing the duty cycle of a switch. In this way the buck converter could determine the operating voltage of the solar panels. In the model shown above, all blue signals are control signal which are being used to implement a maximum power point tracking system. This model uses a simple perturbation and observation algorithm, which is described as a flow diagram below.



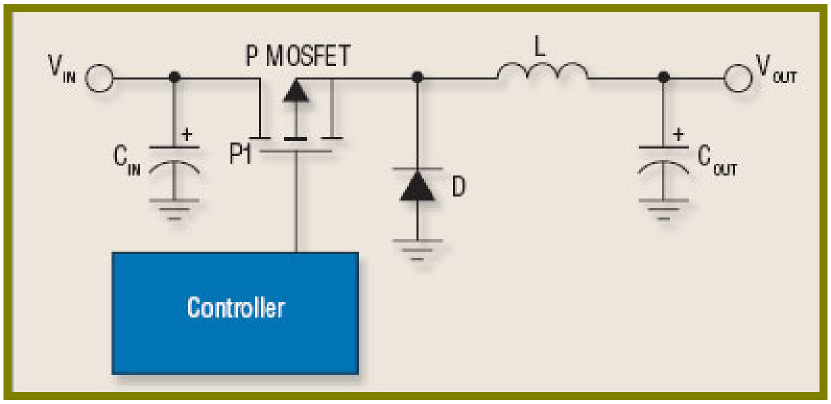
“A Maximum Power Point Tracking System With Parallel Connection for PV Stand-Alone Applications” **IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 55, NO. 7, JULY 2008 page 2674**

The perturbation and observation algorithm basically measures the power coming out of the solar arrays. If the buck converter duty cycle is increasing and power is increasing the algorithm will keep increasing the duty cycle. If the buck converter duty cycle is decreasing and power is increasing the algorithm will keep decreasing the duty cycle. If the power starts to decrease, the algorithm is going to change the direction of the duty cycle (decreasing or increasing). In this way, the circuit should produce the maximum power for a wide range of environmental conditions.

The model was developed for the perturbation and observation algorithm. The model maintains a maximum power for a variety of temperatures and insolation values tested. In addition, when the maximum power of the arrays is reached, the arrays are operating at the voltages and currents, which were found to produce the maximum power in the I-V curves generated by the solar panel model (section 2).



In the simulation above, we can see the proper operation of the MPPT algorithm. The blue line is the output power of the arrays. There are some transients initially to charge up capacitors, but the algorithm quick increases the power to operate at a maximum point and then maintains this output power. The pink line is the battery state of charge. The current coming out of the solar arrays is slowly charging the batteries. As a result the voltage of the batteries is slowly increasing. The input to the buck converter wants to stay at a fixed voltage to maximize power coming out of the array, but the voltage at the output of the buck converter is increasing slowly with the battery voltage. In order to accommodate for this changing ratio between the buck converter's output and input voltage, the duty cycle of the buck converter's switch must change. We can see this in the simulation. The yellow line is the buck converter duty cycle and it is slowly increasing to accommodate the increasing battery voltage.



Buck Converter

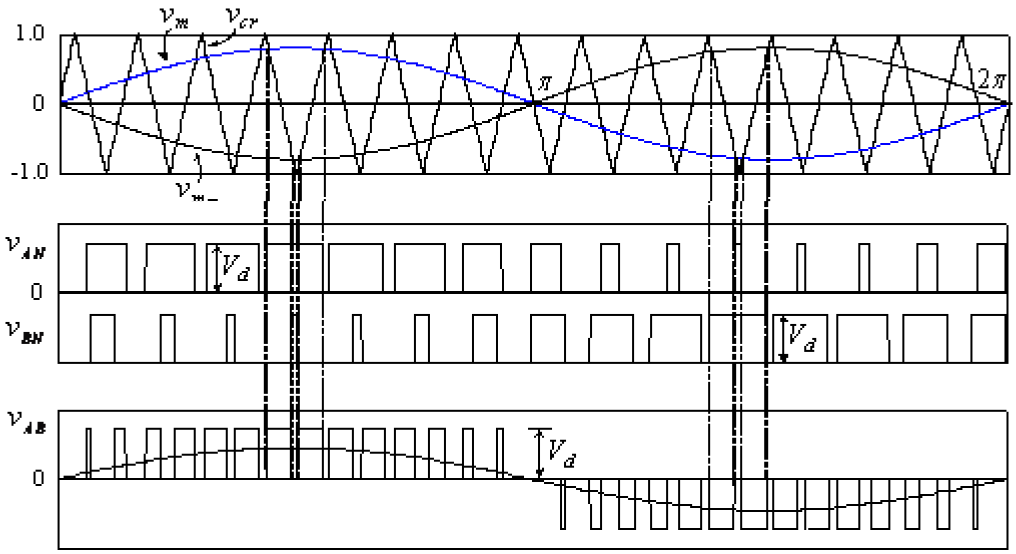
This model provided great confidence that an MPPT algorithm was feasible and actually not much extra work to implement. Using a microcontroller we can use the perturbation and observation algorithm to control the buck converter duty cycle and achieve MPPT.

Unfortunately there were some limitations to the model. The battery did not always seem to charge properly. When the battery started out with a charge greater than about .015% of its capacity, it would start to discharge. This occurred even when the input voltage to the buck converter was far greater than the battery voltage. It was difficult to analyze this problem because the time step for the simulation must be small to carry out MPPT, but the batteries charge very slowly. This problem needs to be resolved, but the model showed a simple perturbation and observation MPPT algorithm can maximize power.

5. Inverter Simulation

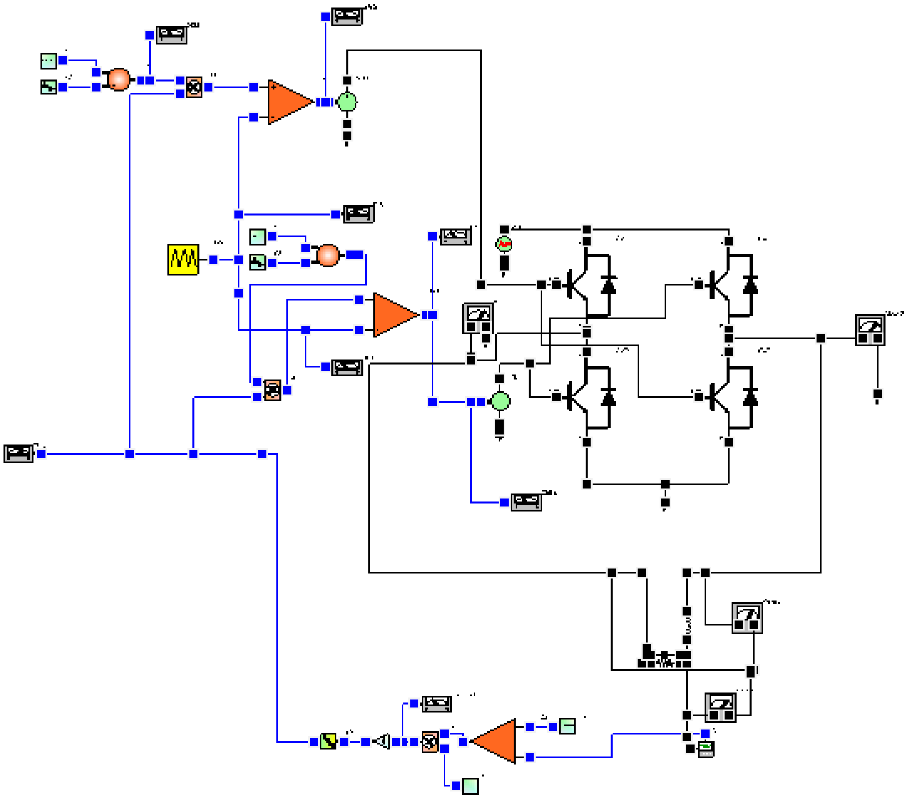
The second big control algorithm in the system was an algorithm to regulate the output voltage. The voltage coming from the battery or solar panel array feeds right into the inverter, which produces 120VAC. Thus the inverter needs to be able convert the input voltage of anything between 160VDC (from the battery) to 305VDC to 120VAC. To do this we needed a control algorithm for our PWM scheme driving the inverter switches.

Based on our Matlab simulations of different PWM schemes, we found the Unipolar PWM had the lowest harmonic content after filtering. Keeping harmonic content low was important to meet specification on THD of 3%. The unipolar scheme works by comparing two 60Hz sine waves to a faster triangle wave. The sine waves are 180^0 out of phase as shown below.



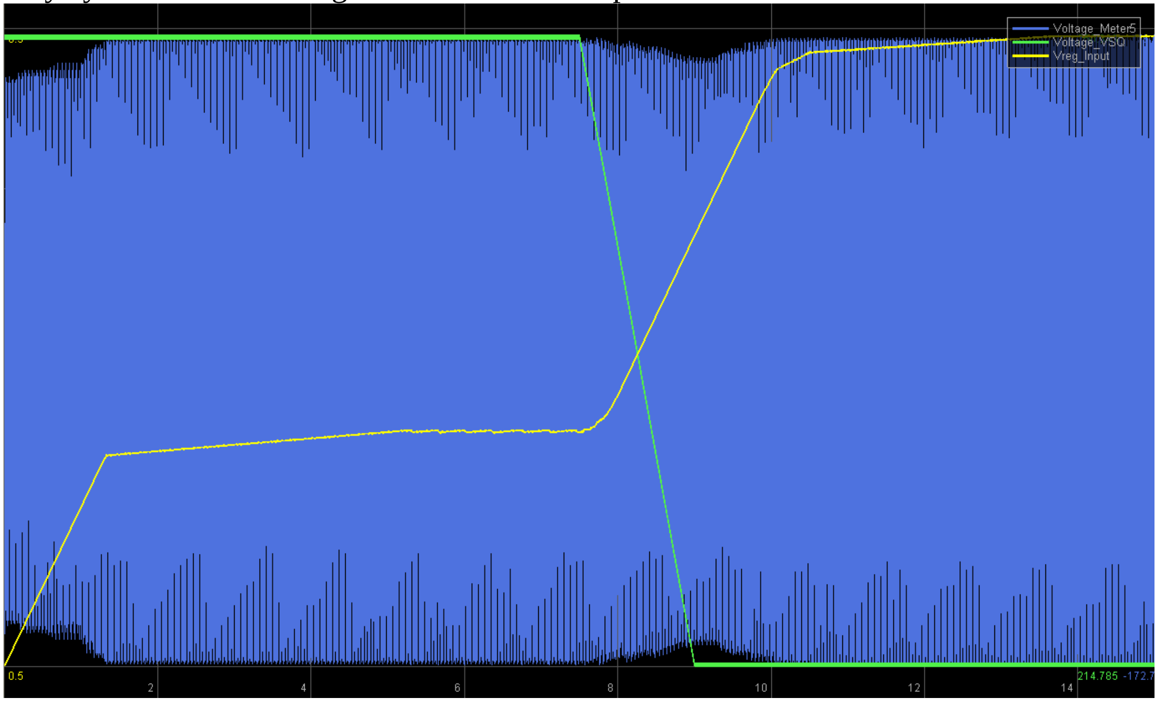
Whenever the blue wave is greater than the triangle wave, in one leg of the inverter we turn on the top IGBT and turn of the bottom IGBT. Otherwise we turn on the bottom IGBT and turn off the top IGBT in the first leg. Whenever the black sine is greater than the triangle wave we turn on the top IGBT and turn off the bottom IGBT of the second leg of the inverter. Otherwise, we turn off the top IGBT and turn on the bottom IGBT of the second leg of the inverter. This creates the voltages V_{AN} and V_{BN} .The load is attached between creating the differential voltage V_{AB}. This voltage filters out nicely to a 60HzAC signal.

This type of PWM scheme was implemented in the VTB model below. In addition, a control algorithm was added to regulate the output voltage. By decreasing the amplitude of the reference sine wave (blue and black), the pulses decrease in length. As a result the voltage at the output decreases. Increasing the sine amplitude does the reverse and increases the output voltage. In this way the model measured the output AC voltage if it was less than 120, the sine amplitudes were increased. If the output voltage was too high the sine amplitudes were decreased.



In the simulation below the high voltage input to the Inverter changes to show how the system responds to a changing input voltage. This is important since the voltage coming from the PV array and battery is dynamic. The green line is the voltage coming into the H-Bridge, the blue line is the AC signal coming out at the output, and the yellow line is the duty cycle of the PWM being fed in to the H-Bridge switches. The simulation shows that initially the output voltage is about 100VAC. The duty cycle of the PWM control increases until it reaches a point where the output settles around 120VAC. We can see that the amplitude of the sine wave does converge to a fixed value by 7.5 seconds. After 7.5 second the input voltage to the H-Bridge changes from its initial value of 250v to

215v. This causes the output voltage to decrease, and the feedback loop must readjust the duty cycle of the PWM signal to correct the output.



This model gave confidence to the PWM scheme incorporated into our design. The model proved that it is possible to have a varying voltage coming into the inverter and regulated the output voltage to a specified value.

There were several short comings of the model. First, VTB could not accurately model the IGBTs. Switch cell models were used but they caused converges errors for the filter values calculated at the output. These filter values did work in PSpice models. Initially it appeared that changing one switch cell to be on at the same time that another switch cell was turned off could cause the converge problems. However, a model was created to specifically prevent this from happening by inserting small delays into the driver signals, but convergence errors still persisted. In addition because the switch cells did not accurately model the voltage drop of the IGBTs a range of input voltages to the inverter that would produce 120VAC could not be defined. This VTB model has proved the proof of concept and that this PWM scheme and control algorithm can regulate an output voltage for varying input voltage. Due to several shortcomings of VTB it appears more detailed simulations will need to be done with other software.