Variable Step Maximum Power Point Tracking for Photovoltaic Modules with Embedded Temperature Sensing

John Sealy

A thesis

submitted in partial fulfillment of the

requirements for the degree of

Master of Science in Electrical Engineering

University of Washington

2016

Committee:

Richard Shi

Daniel Kirschen

Program Authorized to Offer Degree:

Electrical Engineering

University of Washington

Abstract

Temperature-Based Variable Step Maximum Power Point Tracking for Photovoltaics

John Sealy

Chair of the Supervisory Committee:
Professor Richard Shi
Electrical Engineering

The industry standard method for maximum power point tracking of photovoltaics is still one of the simplest methods, in spite of more efficient methods being available. More efficient methods are often more expensive or difficult to implement. To bridge the gap between simplicity and efficiency, the author developed a new maximum power point tracking method. This method incorporates a variable step size and embedded temperature sensing. By interpolating amongst several pre-calibrated points, the maximum power point is estimated and used as the first step before switching to a simpler, more common algorithm. The proposed method is simple to implement, does not require expensive additional components, does not depend on knowledge of datasheet parameters, and incorporates feedback to ensure the actual maximum power point is reached, and not simply estimated.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

First and foremost, I'd like to thank my wonderful partner, Kelsey, for always supporting me. I'd also like to thank Dr. Richard Shi and Dr. Daniel Kirschen for their guidance. This project also wouldn't have been possible without help from my colleague, Ahmad. Finally, I'd like to acknowledge my dog, Doris, and my cat, Stringer Bell, for their invaluable support.

I. Introduction

Section 1.1: Motivation and Objectives

In spite of the growing popularity of solar energy, there are still issues that need to be overcome. Lack of large scale energy storage is a barrier for intermittent and unpredictable energy sources such as wind and solar. In addition, solar energy is often seen as inefficient.

Photovoltaic module efficiency is reduced when not operating at the maximum power point. For a module with a constant set of conditions, there are numerous possible operating points or pairs of voltage and current values. The maximum power point is the operating point at which the most power is delivered to the load.

Since the operating point for a solar module with a constant set of conditions depends on the impedance of the load applied, many attempts have been made to "adjust" the impedance seen by a module to force it to operate at the maximum power point. This is done by inserting a power converter between the module and the load. The duty ratio of the switch in the converter is regulated to change the input impedance.

There are many algorithms used to force a photovoltaic module to operate at the maximum power point. They vary in complexity and effectiveness. However, in spite of the amount of research in this area, the industry standard is still one of the simplest methods [1]. This is because the simplest methods are easy to implement, do not require additional expensive components, do not require knowledge of solar panel datasheet parameters, and still increase the efficiency of the solar panel.

The purpose of this project is to develop a maximum power point tracking method that incorporates the simplicity of the most common methods but delivers better performance. The proposed method accomplishes this without adding mathematical complexity or additional expensive components, and it does not require use of datasheet parameters.

Section 1.2: Solar Energy Background

In recent years, renewable energy has experienced a large growth in the United States. In 2013, renewable energy grew to nearly 15% of installed US capacity and 13% of electricity generation and accounted for 61% of new capacity installation in the US [2]. This trend continued in 2014, with renewable energy growing to 15.5% of installed capacity and 13.5% of generation while again accounting for more than 50% of new capacity additions [3].

While solar energy makes up a comparatively small portion of the total electricity produced in the United States (only 0.5% of generation in 2013 [2] and 0.8% in 2014 [3]), it is also growing very quickly. The installed solar capacity increased by more than 54% in 2014 and accounted for more than 48% of new renewable capacity installed in 2014 [3]. Between 2000 and 2013, solar electricity generation increased worldwide by a factor of 68 [2]. The amount of solar electricity capacity and generation is even higher than these numbers suggest, as they do not include smaller grid-connected capacity and generation, a rapidly growing market [3]. Early numbers from 2015 suggest similar growth, with solar energy account for 40% of new installed capacity in the first half of 2015, more than any other energy type [4].

Indeed, residential solar capacity increased 51% from 2013 to 2014 [4]. This is likely because the residential costs of installing solar capacity have dropped 45% since 2010; even utility costs have dropped as low as $0.05/kWh [4]. With the dropping costs and added interest in solar energy, the value of US solar installations in 2014 was over $17.8 billion, an increase of 9% over 2013 [4].

I. Background Information

In this section, we provide a background on the important concepts surrounding photovoltaics, including discussions of the effects of temperature, irradiance, and loading.

Unlike many sources of electrical energy, the power input-output relationship of a solar panel is complicated. Whereas with many rotating generators the output electrical power is equal to the input mechanical power minus some losses, the output power of a solar panel can vary based on the type of load applied, even holding the temperature and irradiance constant.

Consider the example shown in Figure I.1. Figure I.1 shows a family of voltage and current curves for a solar panel. Each curve represents the characteristic for a panel with a different level of irradiance. The straight line represents a purely resistive load; the intersection of the load line and the characteristic is the operating point of the panel at a given irradiance. The dot on each characteristic represents the maximum power point, or the location on the characteristic at which the maximum power is being delivered to the load.

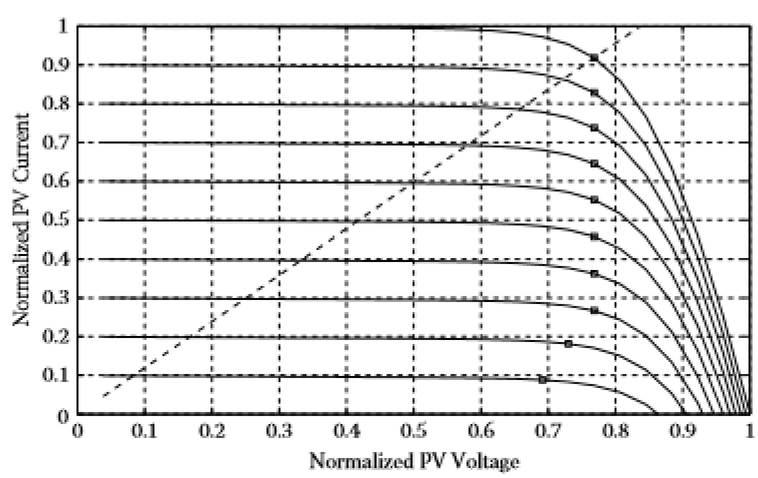

Figure I.1. A family of voltage-current characteristics for a solar panel with varying levels of irradiance. The straight line represents a resistive load [1].

Consider the intersection of the load line with the highest curve. At this level of irradiance, the panel is delivering maximum power to the load. However, should the panel receive a lower level of irradiance, the characteristic will change to one of the others, but the load remains the same. Thus, the panel will be operating at a point that no longer delivers the maximum power for the current characteristic.

This can also be seen with varying temperatures. Figure I.2 shows the effect of increasing temperatures. As the temperature increases, the maximum power point moves to the left, at a lower voltage.

This results in a dynamic optimization problem. To operate most efficiently, the power from the solar panel should be maximized. However, the value and location of the maximum changes with different temperature and irradiance conditions.

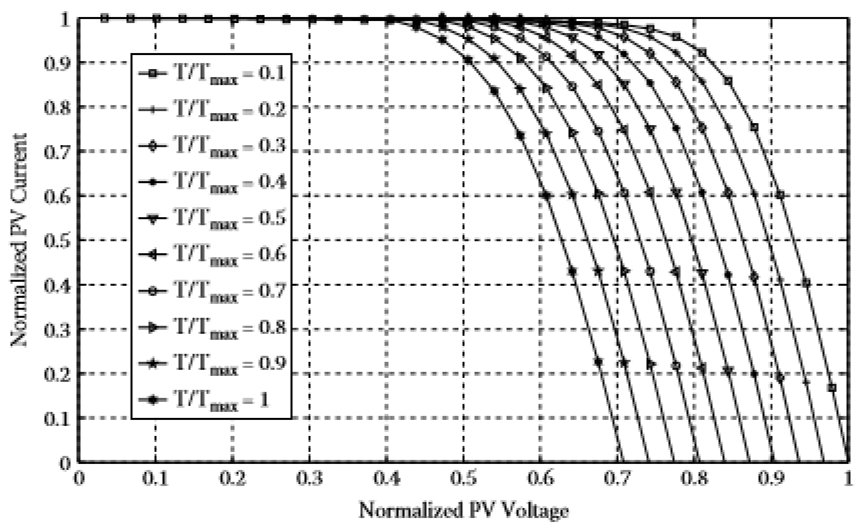

Figure I.2: This shows the effect of changing temperatures on the voltage and current characteristics of a solar panel [1].

In order to successfully operate at the maximum power point, a tracker must be implemented that can dynamically find the maximum power point over time with these changing irradiance and temperature conditions.

Section 1.3: Models for solar panels

A common electrical model for a photovoltaic module is shown in Figure I.3.

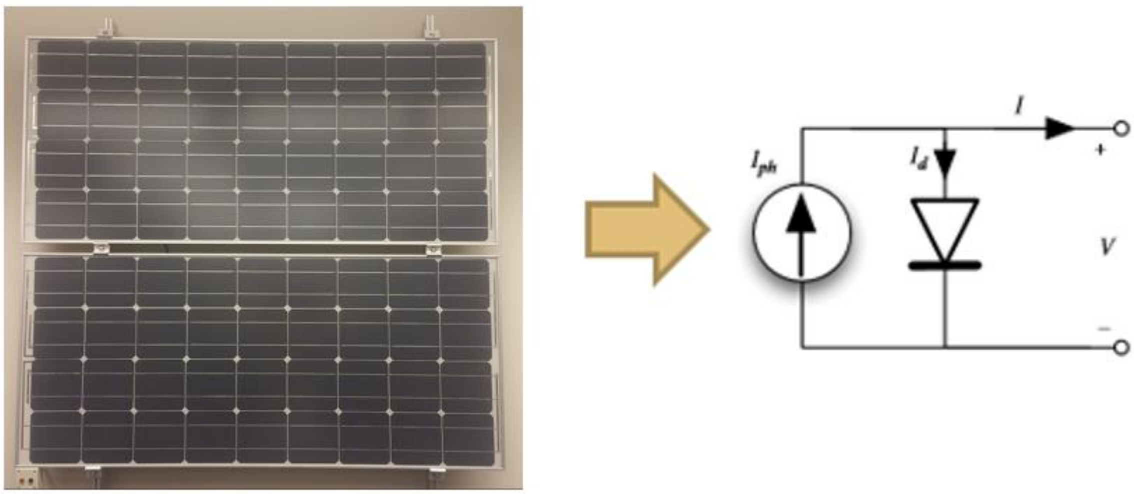

Figure I.3: The ideal model of a solar panel [1].

In the ideal model, I_{ph} is defined as shown in (1*)*.

$$I_{ph} = I_{ph,STC} \frac{G}{G_{STC}} [1 + a_I(T - T_{STC})] \qquad (1)$$

The standard testing conditions (STC) are 25 degrees C and 1000 W/m^2. Thus, the value of the solar current is dependent upon the level of irradiance (G), the temperature (T) and the temperature sensitivity (a_I) [1]. The current leaving the solar panel is the solar current minus the diode current. The current through a diode has the familiar form of (2), where I_0 is the reverse saturation current (typically in the range of nA) and V_t is the thermal voltage.

$$I_d = I_0(e^{\frac{V}{Vt}} - 1) \quad (2)$$

Thus, the equation for the current leaving the cell is shown in (3).

$$I = I_{ph} - I_0(e^{\frac{V}{Vt}} - 1) \quad (3)$$

This equation results in a current-voltage characteristic similar to that in Figure I.2. However, this model is ideal and does not include losses from conducting metallic ribbons or other non-idealities [1]. There are other models that better capture these effects, including the addition of parallel and series resistances, models with additional diodes, and models with controlled current sources. However, these models create additional mathematic complexity, and the differences between the models over areas of interest are small [5]. Thus, for this work, the simplest ideal model will be used.

The power from the solar panel is shown in (4). An example of a power and voltage characteristic is shown in Figure I.4.

$$P = V[I_{ph} - I_0(e^{\frac{V}{Vt}} - 1)] \quad (4)$$

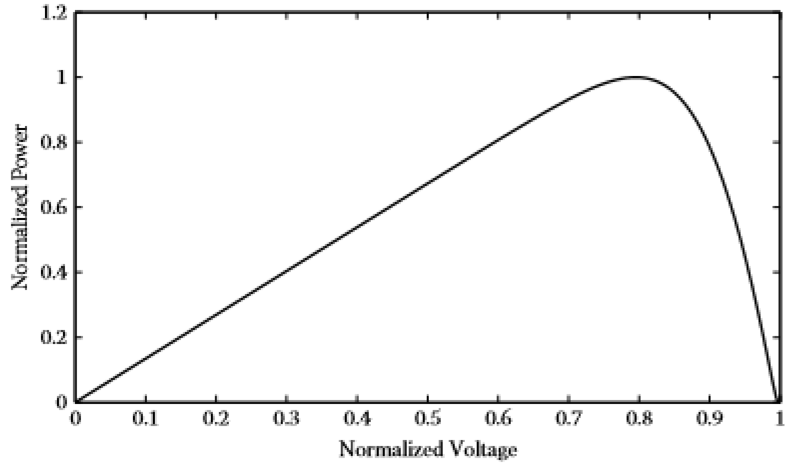

Figure I.4: The power and voltage of a solar panel under ideal conditions [1].

Section 1.4: *Tracking Types*

Because the shape of the characteristic depends on temperature, irradiance, and physical parameters of the solar panel, it is difficult to change the shape of the characteristic to meet the load line. Instead, maximum power point trackers attempt to change the impedance seen by the terminals of the solar panel. This has the effect of changing the load line seen by the solar panel; thus, the load line can be adjusted to always fall on the maximum power point for any characteristic.

To change the impedance seen by the solar panel, a switching power converter is placed between the solar panel and the load. Whereas switching converters are usually used to control voltage levels, in maximum power point tracking, regulation of the switch has the effect of changing the impedance and thus the operating point. This is shown in Figure I.5.

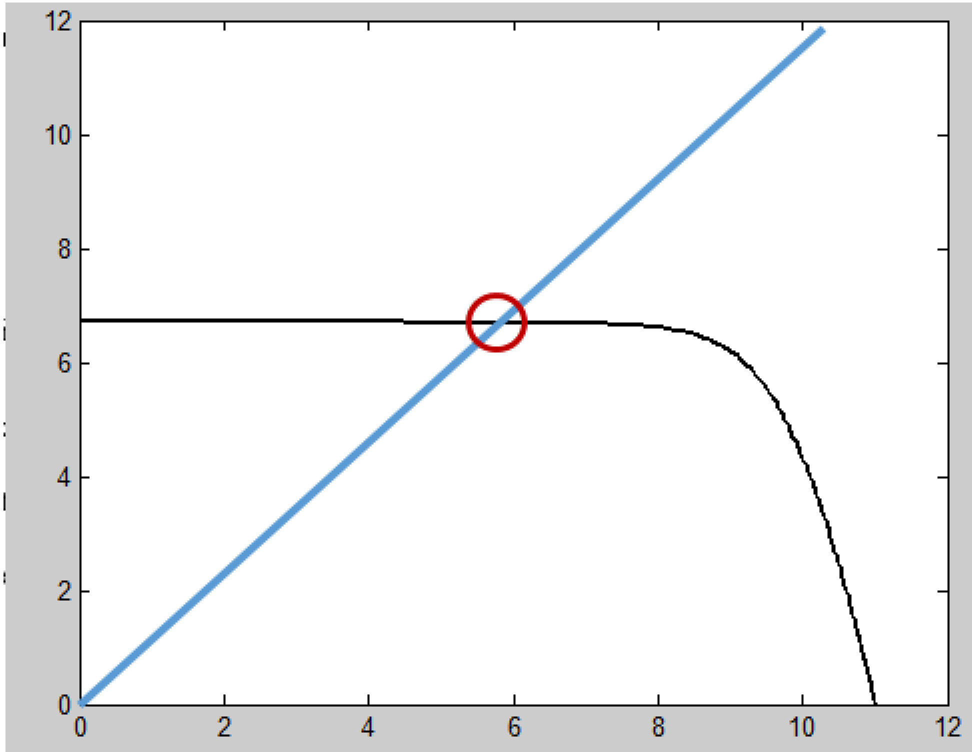

Figure I.5: The operating point (red) of the solar panel is the intersection of the load line (blue) and the panel characteristic (black). [1].

The slope of the load line in Figure I.5 is 1/R, where R is the impedance seen by the panel. As the impedance changes due to the regulation of the switch, the slope of the line changes, and thus the operating point changes. By adjusting the impedance, the operating point can be set to the maximum power point.

There are many different algorithms and methods used to dynamically track the maximum power point. Common methods are perturb and observe, incremental conductance, and constant voltage [6]. The following sections provide a brief description of each of these methods.

1.4.a: Perturb and Observe

The perturb and observe (P&O) algorithm typically makes use of a voltage and current sensor. These sensors periodically take readings of the panel or load voltage and current. Based on these readings, previous readings, and previous tracker decisions, the duty cycle of the tracker is adjusted to attempt to reach the maximum power point. A more detailed description follows.

Consider a tracker that has been operating for some time. The controller takes measurements of voltage and current. Using these measurements, it calculates the power being delivered. Then, it compares the power at the current time to the power from the previous measurements. There are two possibilities: the power is greater now, or the power is less now. If the power is greater, the controller makes the same decision in this cycle as it did in the previous cycle. If it increased the duty cycle and the power went up, it continues to increase the duty cycle. If it decreased the duty cycle and the power went down, it continues to decrease the duty cycle. If, on the other hand, the current power delivered is less than the previous power delivered, then it makes the opposite decision. If the duty cycle was increased and the power decreased, then the wrong decision was made and the duty cycle is now decreased. If the duty cycle was decreased and the power decreased, then the duty cycle is increased.

Thus, the controller periodically adjusts the operating point and observes the effects. Based on this information, the operating point is adjusted again. This is often called a “hill climbing” method, because it results in a tracker that climbs to the maximum power point, and then oscillates around this point. A flow chart for the P&O algorithm is presented in Figure I.6.

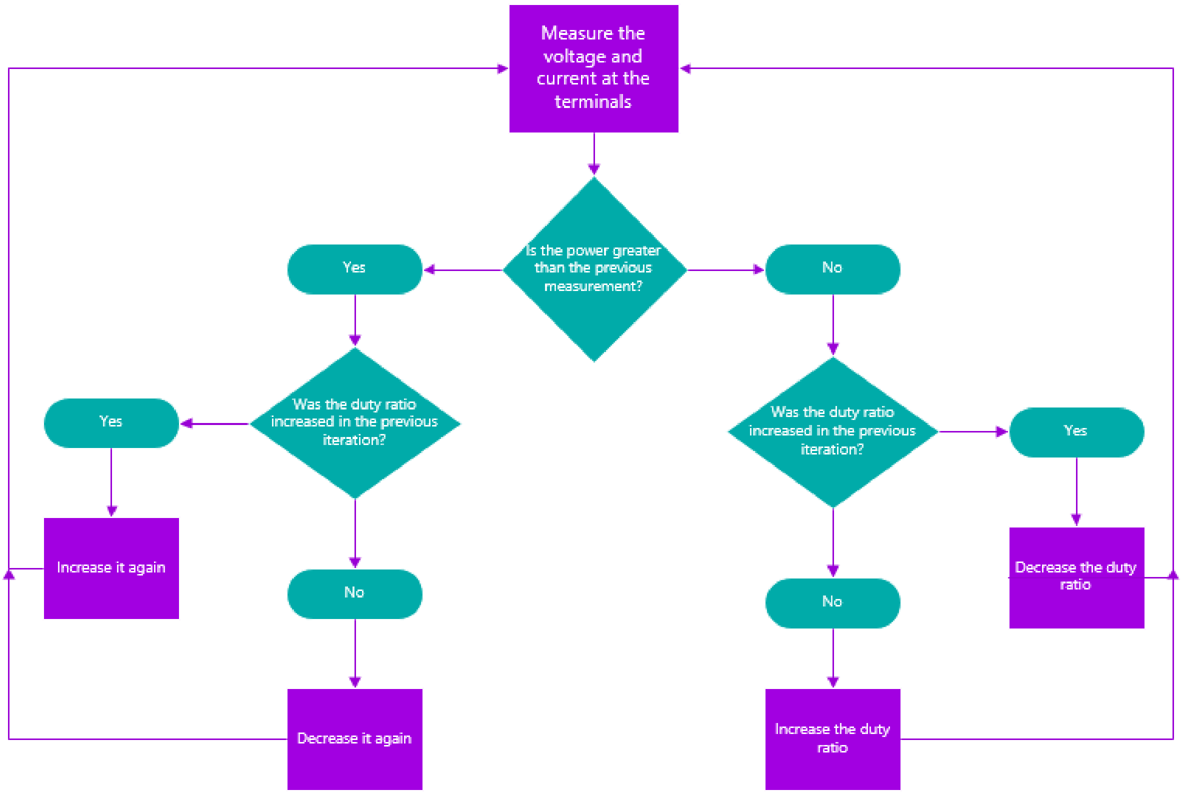

Figure I.6: This shows the algorithm for the perturb and observe method.

1.4.b: Incremental Conductance

A similar method is the method of incremental conductance (INC). Like the P&O method, the incremental conductance method uses a voltage and current sensor to provide information and periodically adjust the duty cycle. However, the INC method differs in the criteria it uses to make a decision.

Consider the power-voltage characteristic shown in Figure I.7.

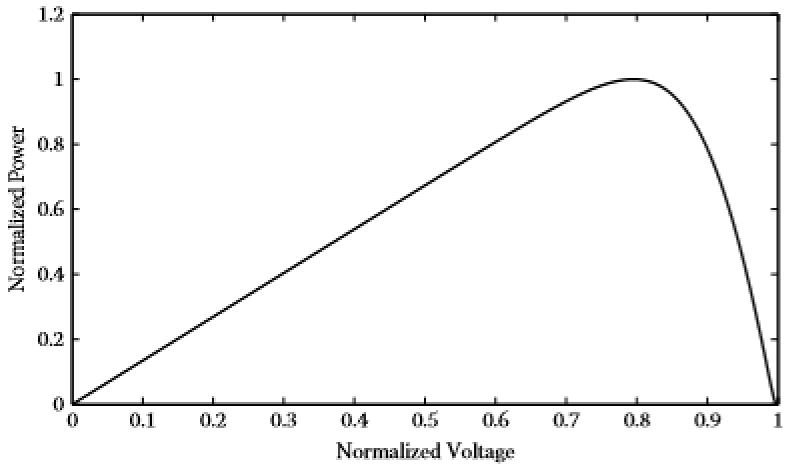

Figure I.7: This shows a voltage and power characteristic for a solar panel.

At the maximum power point, (*5*) holds true.

$$\frac{dP}{dV} = 0 \tag{5}$$

Substituting for power and yields (6).

$$\frac{d(V * I)}{dV} = 0 \tag{6}$$

Differentiating yields (7).

$$I + V * \frac{dI}{dV} = 0 \tag{7}$$

Rearranging gives (8).

$$\frac{-I}{V} = \frac{dI}{dV} \tag{8}$$

From (*8*), it is easy to see from where the incremental conductance algorithm derives its name. At the maximum power point, the negative of the current over the voltage is equal to the rate of change of the current over the voltage. Thus, the controller makes a decision about the duty cycle based on the relative values of the current, voltage, and rate of change of the current.

For the purposes of the controller, at time *t* the rate of change of current with respect to voltage is based on the previous measurements, as shown in (*9*).

$$\frac{-\mathrm{I(t)}}{\mathrm{V(t)}} = \frac{\mathrm{I(t)} - \mathrm{I(t-1)}}{\mathrm{V(t)} - \mathrm{V(t-1)}} \tag{9}$$

If $dP/dV > 0$, then the operating point is on the left side of a maximum. If $dP/dV < 0$, then the operating point is on the right side of a maximum. If $dP/dV = 0$, then the operating point is on the maximum. These cases correspond to the present current over the present voltage being greater than the negative rate of change, less than the negative rate of change, and equal to the negative rate of change respectively [7]. If the first case is true, then the duty cycle should be increased, moving the operating point to the right. If the second case is true, the duty cycle should be decreased, moving the operating point to the left. If the third case is true, the operating point is the maximum power point and no adjustment is necessary. Thus, using the value of the voltage and current at the present measurement and the previous measurement, the controller can make a decision about the duty cycle adjustment. A flow chart for a possible INC algorithm is shown in Figure I.8 [7]. Note that the voltage difference must be checked as the first step to avoid a divide-by-zero error.

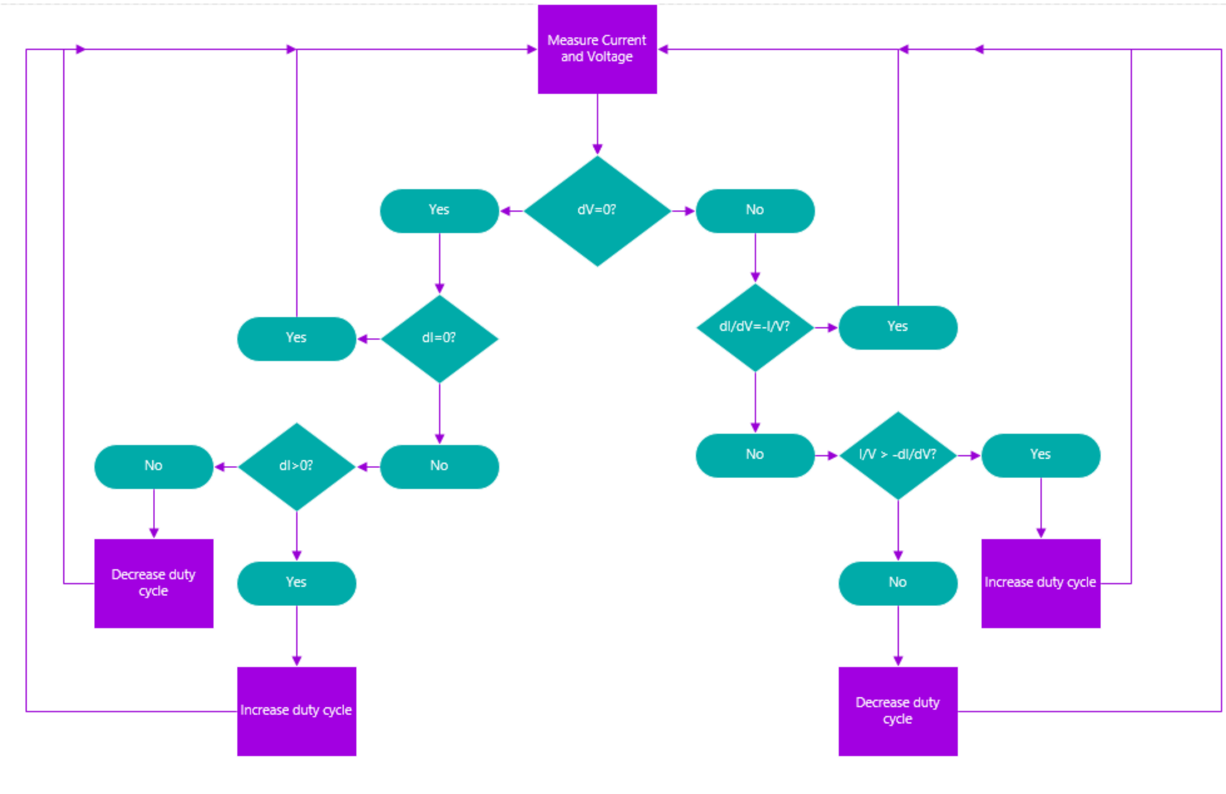

Figure I.8: A flow chart for the INC algorithm.

1.4.c: Constant Voltage

The constant voltage method is one of the simplest maximum power point tracking methods. It is based on the observation that the voltage at the maximum power point is often near a constant fraction of the open circuit voltage. This can be observed in Figure I.9.

Since the voltage at the maximum power point is approximately 70-80% of the open circuit voltage for most of the curves, a reasonable approximation for the maximum power point can be made by simply finding the open circuit voltage and setting the voltage of the solar panel to some

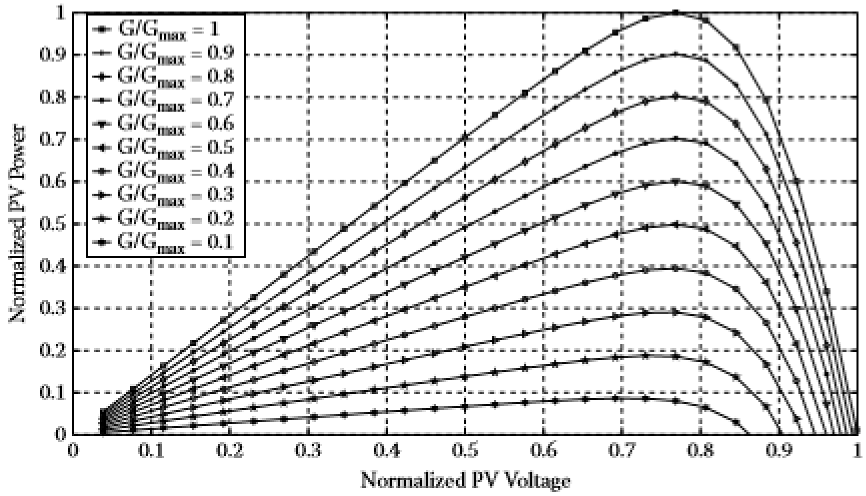

Figure I.9: A family of power vs. voltage curves. The maximum power point is approximately 70-80% of the open circuit voltage for all curves *[1]*.

constant fraction K of the open circuit voltage. Typical values of K range from 78-92% [8]

Section 1.5: Buck Converter

Typically, power electronics switching converters are used to change the level of some voltage input to a different level at the output of the converter. There are many different types of converters, such as buck (output voltage is lower than input voltage), boost (output voltage is higher than input voltage), or buck-boost (output voltage is higher or lower than input voltage). Each of these converters has different topologies. One of the simplest topologies is the buck converter. An example of a buck converter is shown in Figure I.10.

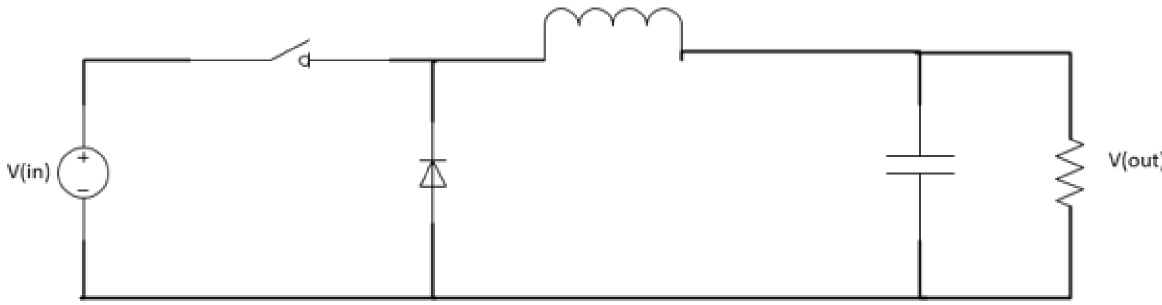

Figure I.10: A basic buck converter.

The operation of the buck converter is as follows: when the switch is closed, current flows from the DC source through the inductor. This creates a voltage drop across the inductor, so the voltage V(out) is less than the voltage V(in). When the switch is open, the energy stored in the inductor's magnetic field will create a current flow, with the inductor acting as a voltage source. The diode creates a return path for the current. Provided the switch is closed again before the energy stored in the magnetic field of the inductor dissipates, V(out) is always non-zero. The capacitor serves to reduce the voltage ripple across the load. The relationship between the output voltage and the input voltage is given in (10).

$$\mathrm{V(out)} = \mathrm{V(in)} * \mathrm{D} \qquad (10)$$

In (10), D is referred to as the duty ratio. The duty ratio is the time that the switch spends closed over the total switching period. This is illustrated in Figure I.11.

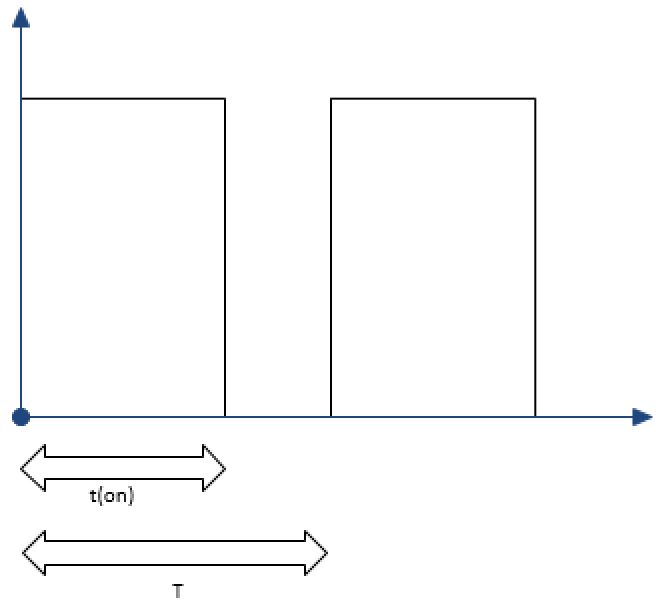

Figure I.11: An illustration of the duty ratio.

Switching converters like the buck converter are essential for maximum power point tracking. The regulation of the switch in the converter is what allows a tracker to change the operating point of a solar panel. The controller used in a maximum power point tracking scheme samples the inputs (usually panel voltage and current or load voltage and current) and makes a decision about the duty ratio of the switch. A block diagram of a typical scheme is shown in Figure I.12.

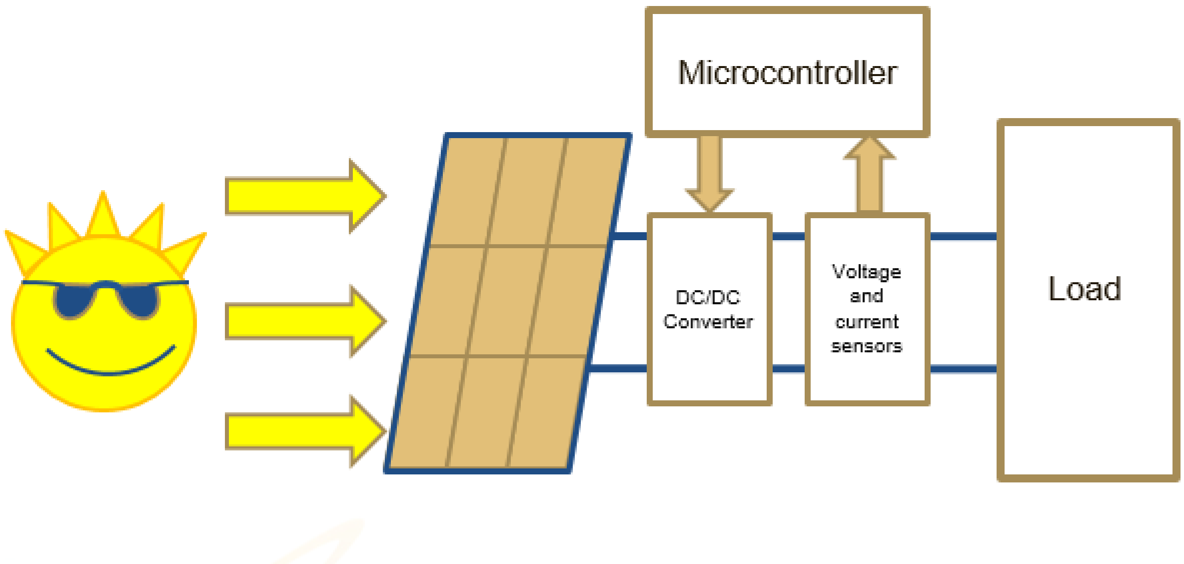

Figure I.12: A block diagram for a typical maximum power point tracking scheme.

II. Proposed Method: Variable Step with Embedded Temperature Sensing

Section II.1: Introduction and Motivation

In spite of the general effectiveness of the previously discussed methods, there are some issues. For example, the P&O method is commonly used. However, consider the case in Figure II.1:

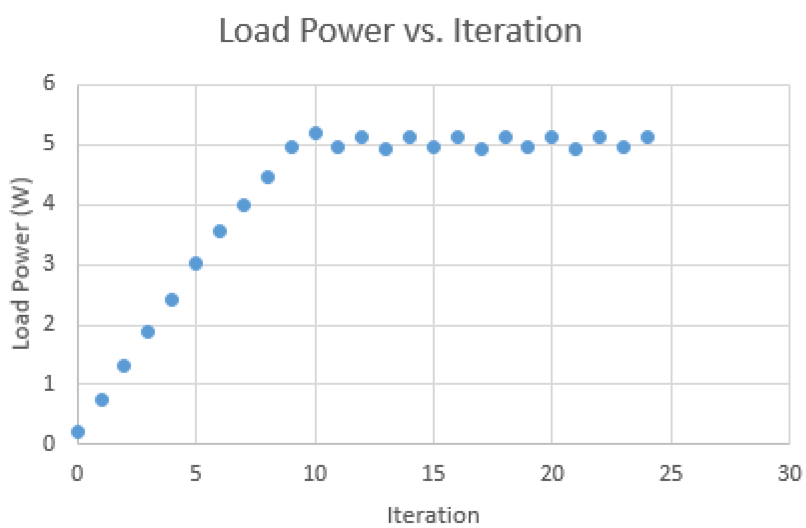

Figure II.1: The load power vs. microcontroller iterations of a solar panel using perturb and observe maximum power point tracking.

In Figure *II.1*, a solar panel is subjected to a perturb and observe tracking scheme. The term "iteration" on the x-axis refers to any time the duty ratio is changed and new data is collected. The tracker is changing the duty ratio based on the observed load power. When started, the duty ratio is zero and no power is being delivered. The ratio is then incremented once, the power increases, and the duty ratio is incremented again. This results in a "climb" to the maximum power point. Once the maximum power point has been reached, the operating point will oscillate around it. This occurs at around iteration 10. For most microcontrollers, each decision is made very quickly (fast enough to be negligible). However, a decision about duty ratio adjustment can only be made after the system has had time to settle. That is, the voltage and current of the panel must have time to return to a new steady value after an adjustment has been made before a new adjustment can be made. The time between perturbations can thus cause a loss of power as the system climbs to the maximum power point.

This can be partially remedied by making the steps larger; if the duty ratio is increased or decreased in bigger steps, it will climb to the maximum power point more quickly. However, upon reaching steady state, this means the oscillations will take the operating point further from the true maximum power point [6].

To combat this, some authors have proposed variable step sizes [9]. The types are numerous, but one of the simpler methods is to combine the constant voltage and perturb and observe methods. With this method, the first step made is to set the panel voltage to approximately 70% of the open circuit voltage. Then, the P&O algorithm is run as normal. This has the effect of reducing the initial climb to the maximum power point on startup or rapidly changing conditions.

However, as mentioned in the previous section, the maximum power point voltage is not always at approximately 70% of the open circuit voltage. Thus, if a better estimation of the maximum power point voltage can be used, then the initial step will be much closer, resulting in an even smaller climb and saving more power.

There is also an issue with rapidly changing conditions [10]. For example, if the irradiance is increasing, the power delivered by the panel will increase, provided all other conditions remain constant. However, the controller will believe that the power increased due to the perturbation of the operating point, even if the perturbation caused the operating point to move away from the true maximum. Thus, the controller will perform the same action, moving the operating point further from the maximum again. This will continue as the irradiance increases, until the conditions stabilize and the controller has a chance to right itself.

One method of estimating the maximum power point is by sensing the temperature [11]. This method is based on datasheet parameters. The maximum power point voltage at a reference temperature is given, as is a temperature coefficient of the maximum power point voltage. As the temperature changes, the voltage at the maximum power point changes by a scalar of the temperature difference. Thus, if the temperature is measured, the maximum power point voltage can be estimated. However, this requires the temperature coefficient of the maximum power point voltage to be provided on the datasheet, which is often not the case. Additionally, the temperature dependence may vary from the manufacturer provided value [12].

Section II.2: Proposed Method

A proposed method for overcoming these limitations follows. As shown previously, the current and voltage of a panel depend on the amount of irradiance and the temperature. Thus, the maximum power point depends on the amount of irradiance and the temperature. The temperature can be directly measured easily, but the irradiance is more difficult to measure. However, the open circuit voltage also depends on the irradiance and temperature. This is seen in (11).

$$V_{oc} = V_T * \ln(\frac{I_{ph}}{I_0} + 1) \qquad (11)$$

Since the open circuit voltage is easy to measure, the irradiance can be measured indirectly by measuring the open circuit voltage and temperature. Thus, a relationship exists between the open circuit voltage and temperature and the maximum power point voltage. In lieu of deriving a mathematical relationship, a set of data points can be collected. These points consist of the open circuit voltage, the temperature, and the voltage at the maximum power point at various environmental conditions. Then, during the operation of the maximum power point tracker, the open-circuit voltage and temperature are measured. A plane is created from nearby pre-collected points and the maximum power point voltage is estimated. The operating point is set to this estimated point, and then the P&O algorithm is run as normal. For example, given 3 points $p_1=<V_{oc1},T_1,V_{MPP1}>$, $p_2=<V_{oc2},T_2,V_{MPP2}>$, $p_3=<V_{oc3},T_3,V_{MPP3}>$, three equations can be written as in (12).

$$\begin{aligned} aV_{oc1} + bT_1 + cV_{MPP1} + d &= 0 \\ aV_{oc2} + bT_2 + cV_{MPP2} + d &= 0 \\ aV_{oc3} + bT_3 + cV_{MPP3} + d &= 0 \end{aligned} \tag{12}$$

In (12) d is a non-zero number and a, b, and c are defined as follows:

$$\begin{aligned} D &= \begin{vmatrix} V_{oc1} & T_1 & V_{MPP1} \\ V_{oc2} & T_2 & V_{MPP2} \\ V_{oc3} & T_3 & V_{MPP3} \end{vmatrix} \\ a &= \frac{-d}{D}\begin{vmatrix} 1 & T_1 & V_{MPP1} \\ 1 & T_2 & V_{MPP2} \\ 1 & T_3 & V_{MPP3} \end{vmatrix} \\ b &= \frac{-d}{D}\begin{vmatrix} V_{oc1} & 1 & V_{MPP1} \\ V_{oc2} & 1 & V_{MPP2} \\ V_{oc3} & 1 & V_{MPP3} \end{vmatrix} \end{aligned} \tag{13}$$

$$c = \frac{-d}{D}\begin{vmatrix} V_{oc1} & T_1 & V_{MPP1} \\ V_{oc2} & T_2 & V_{MPP2} \\ V_{oc3} & T_3 & V_{MPP3} \end{vmatrix}$$

Then, an equation for a plane can be created with (14).

$$aV_{oc} + bT + cV_{MPP} + d = 0 \tag{14}$$

Then, the maximum power point voltage can be estimated with (15).

$$aV_{oc,measured} + bT_{measured} + cV_{MPP} + d = 0 \tag{15}$$

Once the maximum power point voltage is estimated, the operating point must be set to this voltage. A typical method of doing this in the constant voltage method is by measuring the panel voltage and comparing it to the estimated voltage, then adjusting the duty ratio accordingly. This is shown in Figure II.2.

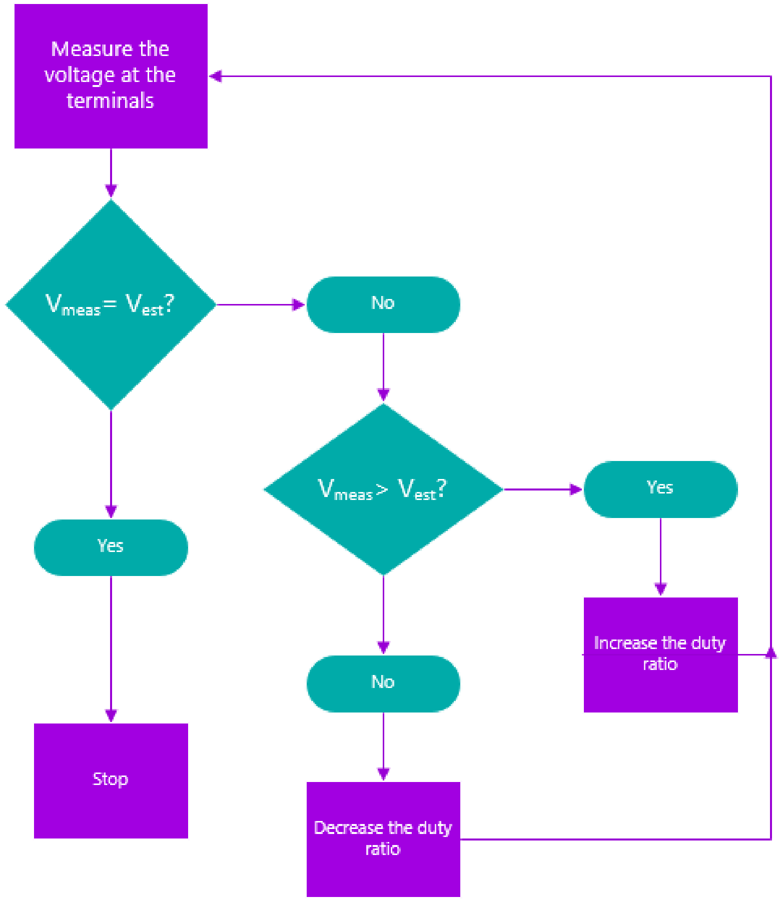

Figure II.2*:* A flow chart for a common method of achieving a specific voltage.

One downside to this method is that it increases the number of increments required to reach the maximum power point. Instead, the author examined the relationship between the duty ratio and the panel voltage by increasing the duty ratio from 0 to 100% and measuring the voltage. An example of the relationship is shown in Figure II.3.

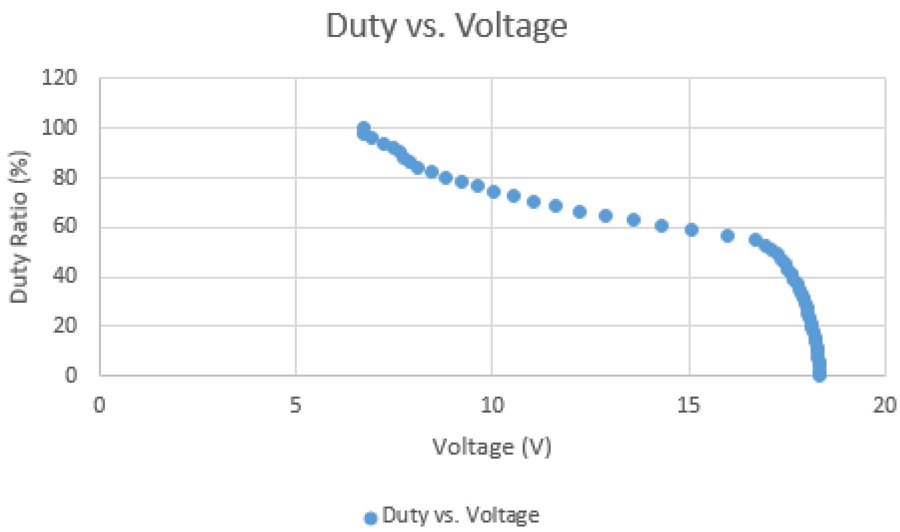

Figure II.3: An example of the desired duty ratio to achieve a specific voltage.

While this curve is non-linear, it can be approximated as piecewise linear. This is shown in Figure II.4 and Figure II.5. Voltages below 7 V are ignored, since the maximum power point voltage is generally not in this range.

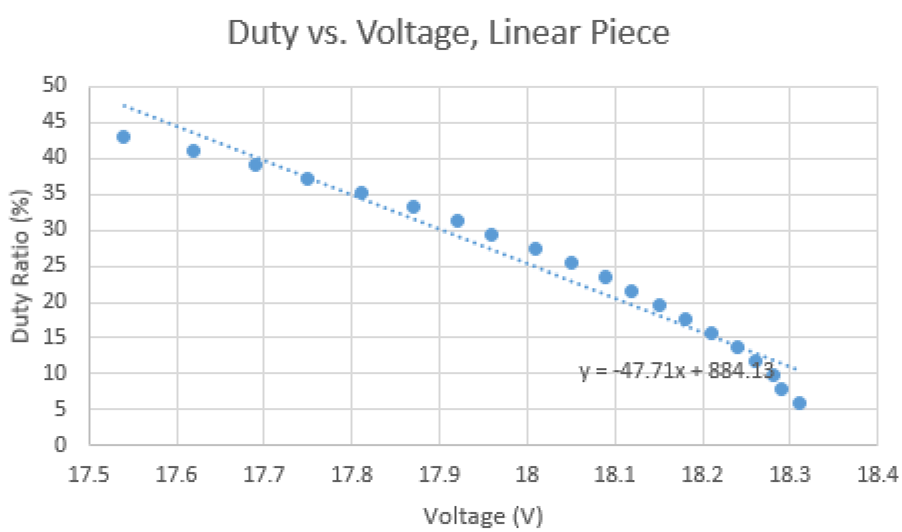

Figure II.4: A linear approximation of part of the duty ratio vs. voltage curve with a high light intensity.

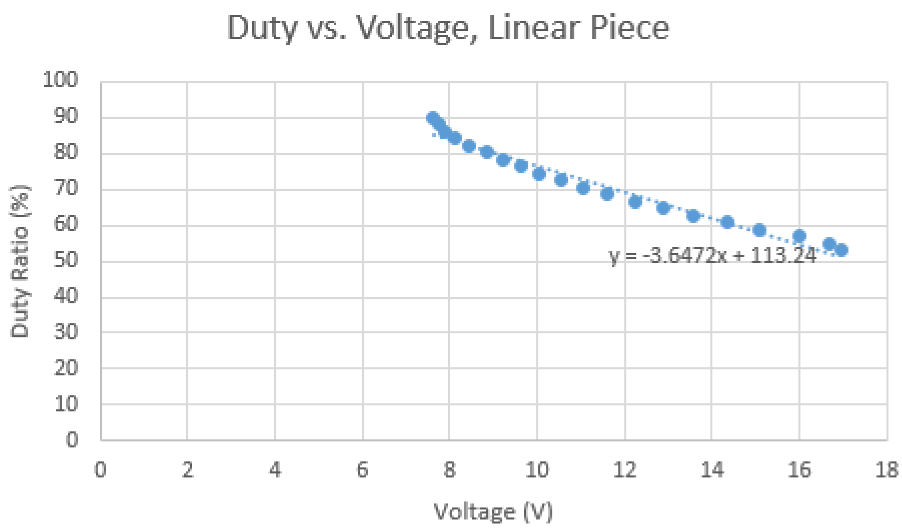

Figure II.5: A linear approximation of part of the duty ratio vs. voltage curve with a high light intensity.

As can be seen in Figure II.4 and Figure II.5, the relationship between the duty ratio and voltage is approximately linear over certain ranges. However, the same relationship does not hold over all conditions. Thus, the duty ratio vs. voltage characteristic must be created every time conditions change significantly. However, provided that conditions do not change often, this still results in fewer duty ratio decisions being made than the perturb and observe algorithm (since it requires climbing to the maximum power point) and the constant voltage method (since it requires increments to set the operating point to the maximum power point). Thus, the proposed algorithm is shown in Figure II.6. Figure II.7 shows a block diagram of the proposed setup.

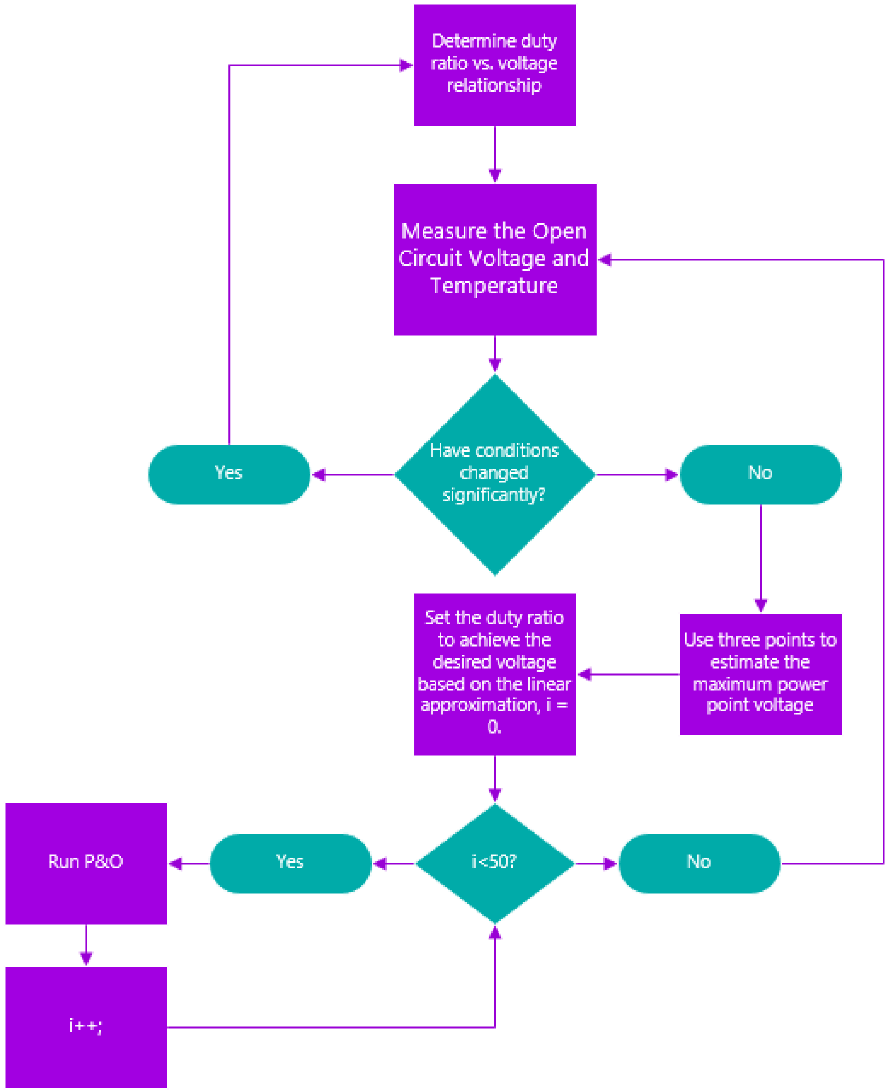

Figure II.6: The proposed algorithm flow chart.

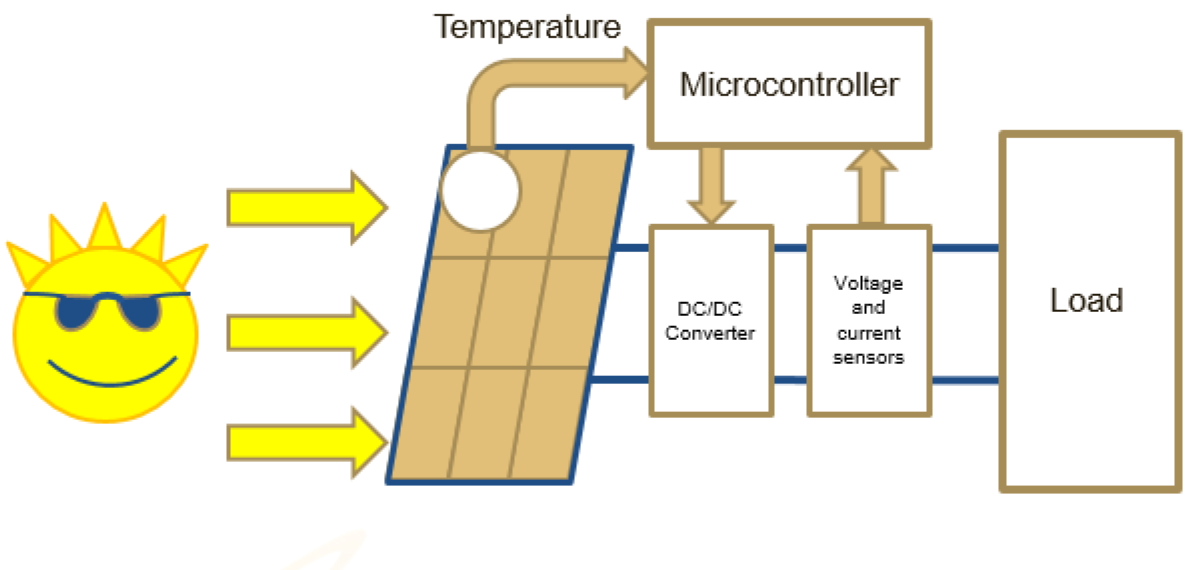

Figure II.7: The block diagram for the proposed method.

III. Simulation Results

To determine the viability of the new method, first the author constructed a MATLAB program to determine whether a viable relationship could be created amongst the open circuit voltage, the temperature, and the maximum power point voltage. The parameters used for this simulation are shown in Table *III.1*. $I_{PH,STC}$ refers to the photon current at standard testing conditions, α is the temperature coefficient of current, n is the number of series connected cells, E_{gap} is the band gap of the semiconductor material and C is the temperature coefficient of the semiconductor material.

Table III.1. Parameters Used

$I_{PH,STC}$	7.7 A
α	.07
n	20
C	373.0739
E_{gap}	1.8*10^-19 J

The temperature was varied from 303 to 352 K and the irradiance was varied from 500 to 1500 W/m^2. Figure III.1 shows the resulting relationship. From the plot, it is clear that the surface created by the open circuit voltage, temperature, and maximum power point voltage relationship lends itself well to the proposed method; it is roughly planar. Interpolating amongst several points should give a good approximation of the maximum power point voltage.

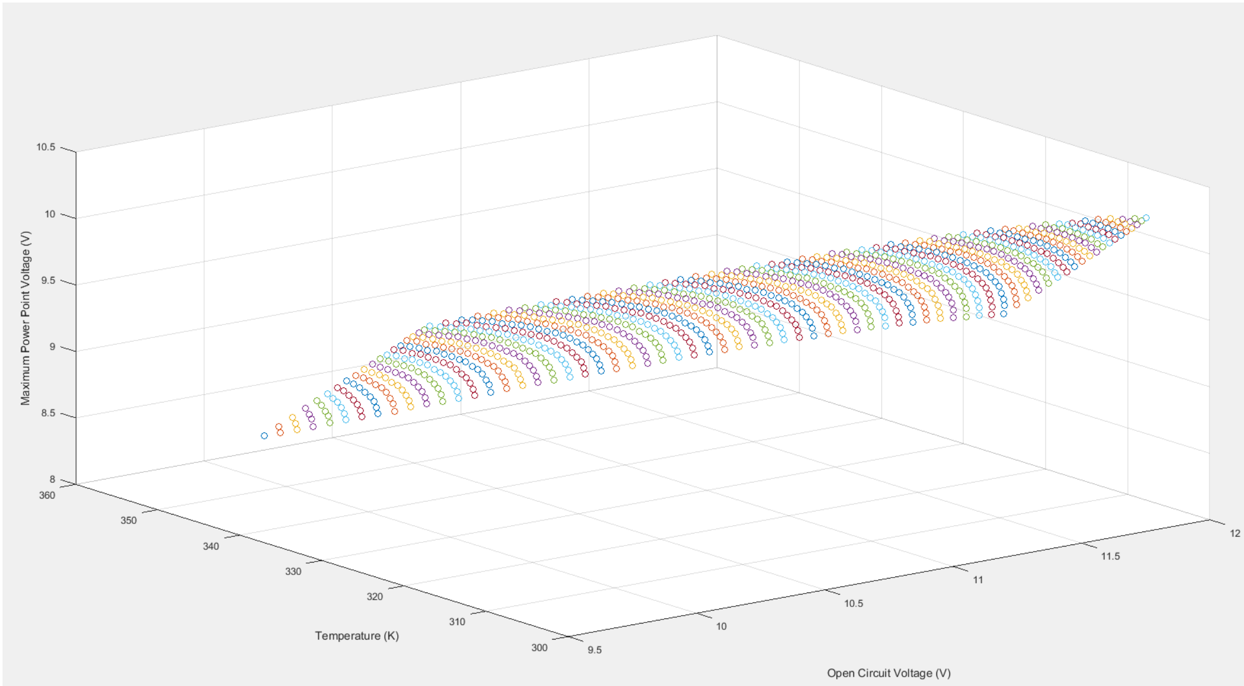

Figure III.1: The relationship amongst the maximum power point voltage, the temperature, and the open circuit voltage.

Next, the author created a Simulink model to test the perturb and observe algorithm. The Simulink model for the perturb and observe algorithm is shown in Figure III.2.

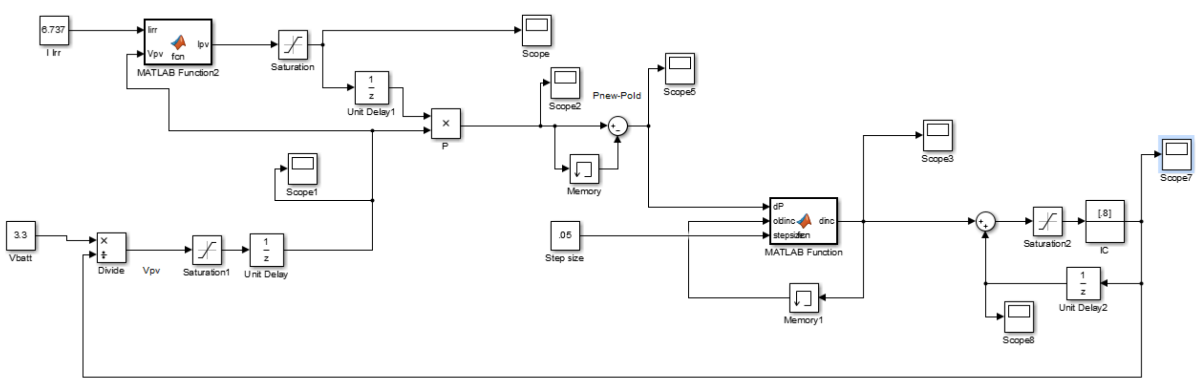

Figure III.2: The Simulink model of the perturb and observe algorithm.

This model assumes a solar current of 6.737 A and a temperature of 30 degrees Celsius. The voltage of the load is assumed to be constant; this simplifies the model construction. Figure

III.3 shows the change in duty ratio over time starting from a duty ratio of .8. Figure III.4 shows the change in duty ratio over time starting from a duty ratio of .05.

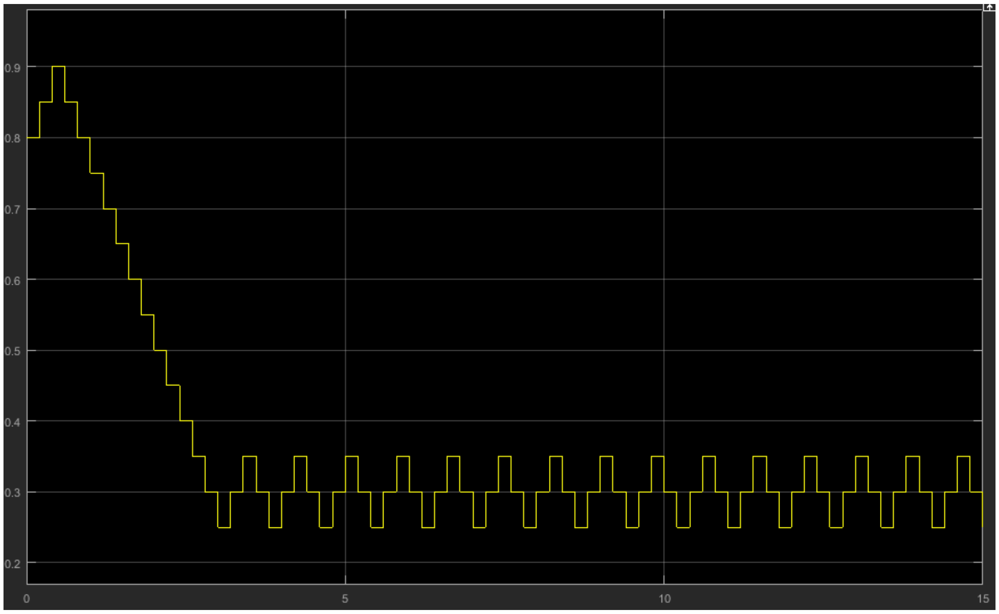

Figure III.3: The change in duty ratio (y-axis) over time (x-axis) when the duty ratio starts at a high value.

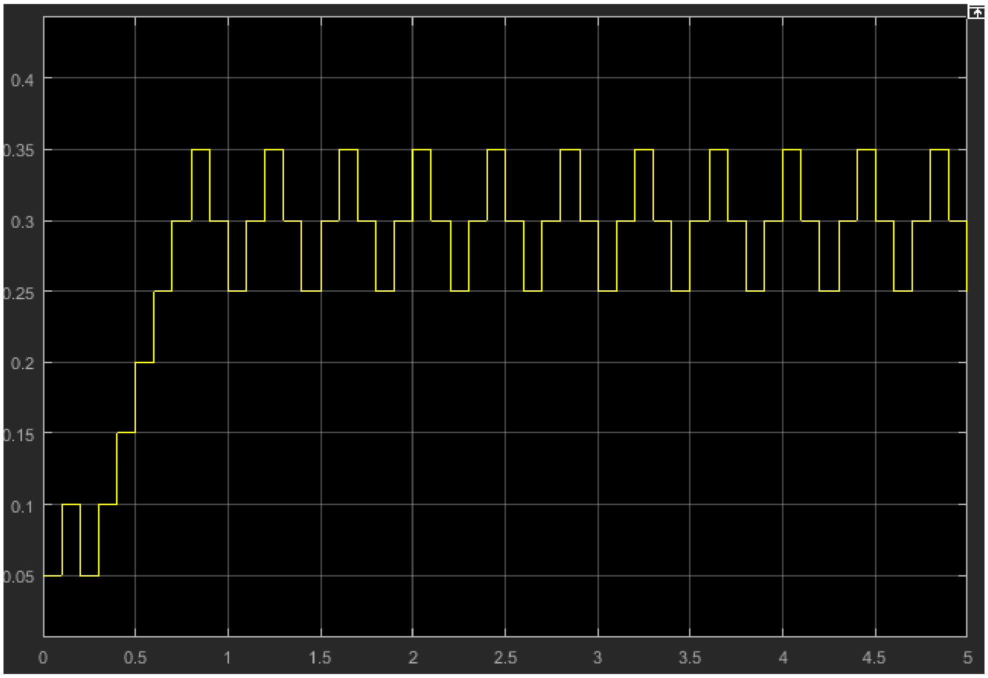

Figure III.4: The change in duty ratio (y-axis) over time (x-axis) when the duty ratio starts at a low value.

In both cases, the duty ratio increases or decreases to the optimal point and begins oscillating. The initial errors (increasing and decreasing before climbing) are caused by the delay blocks in the Simulink model. These blocks help the simulation converge, but result in the first several increments being incorrect.

Next, the author determined what effect being able to reliably estimate the correct duty ratio would have on the number of increments required. To do this, first the author ran several trials to determine the optimal duty ratio for several sets of conditions. Table *III.2* shows the optimal duty ratio for a total of four combinations of conditions (two sets of solar current, two sets of temperature.

Table III.2. Optimal Duty Ratio for Four Sets of Conditions

	Solar current = 10 A	Solar current = 5 A
T=40 degrees C	.15	.25
T=20 degrees C	.4	.3

Next, the Simulink model in Figure III.5 determined the correct duty ratio for a given solar current and temperature. The expanded function block is shown in Figure III.6.

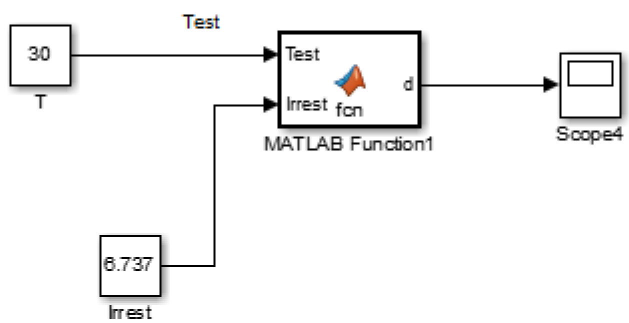

Figure III.5: The Simulink model used to determine the correct duty ratio for a given irradiance and temperature.

```
function d = fcn(Test,Irrest)
%#codegen
Tref1=40;
Tref2=20;
Iref1=10;
Iref2=5;
drefT1I1=.15;
drefT2I1=.2;
drefT1I2=.25;
drefT2I2=.3;

dTestIref1=(Tref2-Test)/(Tref2-Tref1)*drefT1I1+(Test-Tref1)/(Tref2-Tref1)*drefT2I1;
dTestIref2=(Tref2-Test)/(Tref2-Tref1)*drefT1I2+(Test-Tref1)/(Tref2-Tref1)*drefT2I2;

d=(Iref2-Irrest)/(Iref2-Iref1)*dTestIref1+(Irrest-Iref1)/(Iref2-Iref1)*dTestIref2;
```

Figure III.6: The expanded Matlab function used to determine the correct optimal duty ratio for a given irradiance and temperature.

The function in Figure III.6 determines the duty ratio via bilinear interpolation. Setting the solar current to 6.737 A and the temperature to 30 degrees C yielded an optimal duty ratio of .27. The perturb and observe algorithm from Figure I.6 was then run again, this time with the initial duty ratio set to .27. The results are shown in Figure III.7.

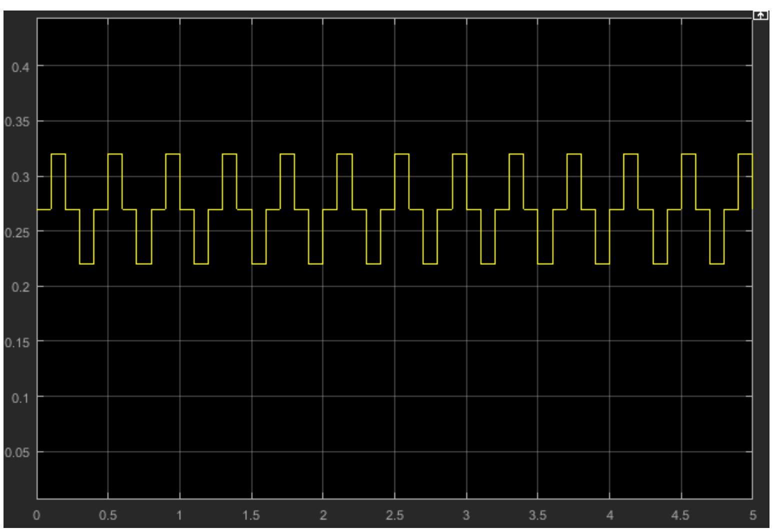

Figure III.7: The duty ratio (y-axis) over time (x-axis) when the initial duty ratio is set to .27.

From Figure III.7, it is clear that the duty ratio estimation was able to eliminate the initial increments that either climb down or up from the starting duty ratio. This implies that correct estimation of the duty ratio may increase the power output of the panel by eliminating the transitory time before it begins oscillating around the maximum power point.

IV. Experimental Set-Up

A schematic of the complete experimental setup is shown in Figure IV.1.

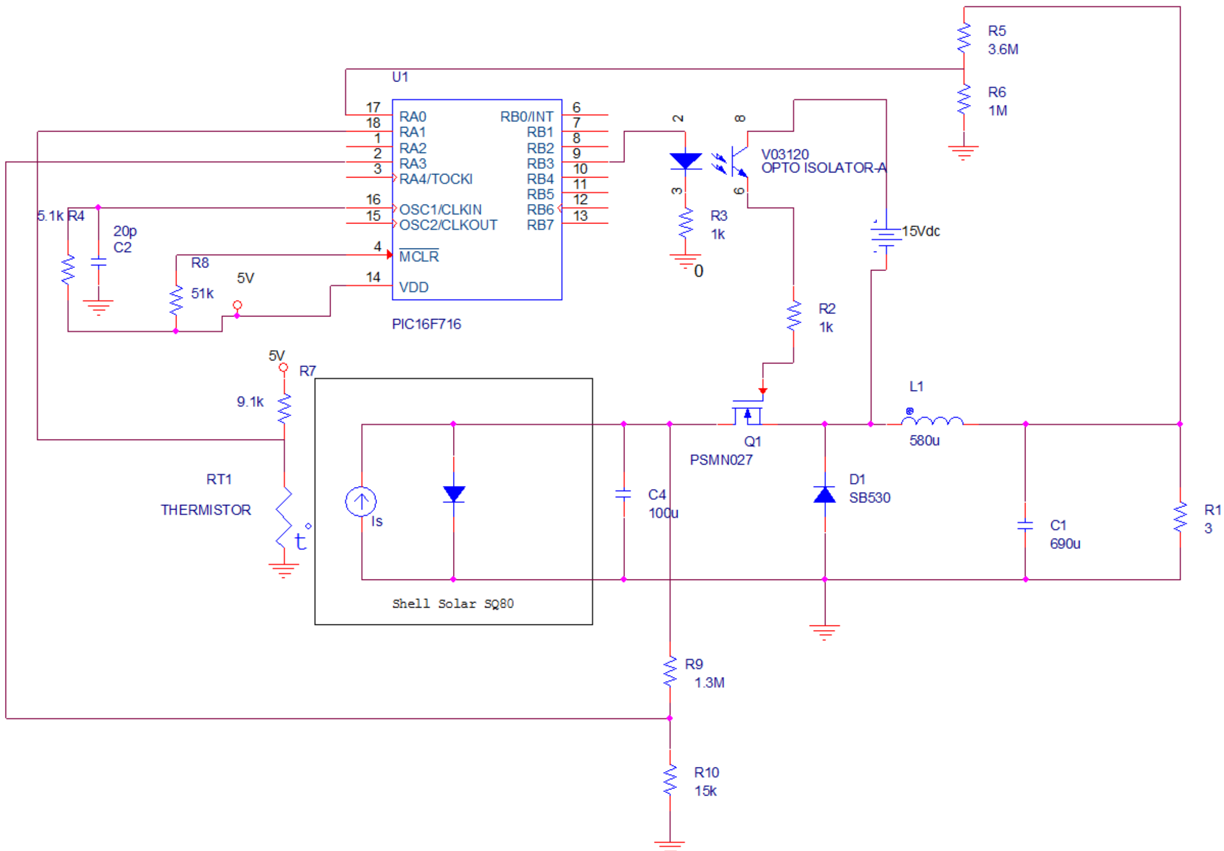

Figure IV.1: The schematic of the experimental setup.

The solar panel used was an 80-W Shell Solar SQ80. Light was provided to the panel using an Arri Arrilite 2000 W halogen lamp with a dimmer switch to control the intensity. The thermistor, a Vishay NTCALUG03A103G, was affixed to the surface of the panel. The microcontroller was a PIC16F716, and was isolated from the panel and converter with a Vishay VO3120 optoisolator. The setup of this optocoupler ensures the gate to source voltage of the transistor is sufficiently high when switching on. The algorithms were written in C-language in

MPLAB IDE X. The microcontroller was programmed using a PICkit 3 Programmer/Debugger. This code is presented in Appendix I.

First, the solar panel was characterized by taking a set of data points. The data points consisted of the panel's open circuit voltage, temperature, and voltage at the maximum power point for varying levels of intensity. To characterize the panel, it was attached to the buck converter and the duty ratio was increased from zero to 100%. This changed the impedance seen by the panel and thus created the voltage-current characteristic. Table *IV.3* shows a result of the data collected; Figure IV.2 through Figure IV.4 show the graphical results of the characterization.

Table IV.3. Data Collected to Calibrate the Controller

Point	Voc (V)	Temp (C)	Vmpp (V)
A	18.4	42	11.9
B	17.9	40	11.5
C	17.4	37	10.7

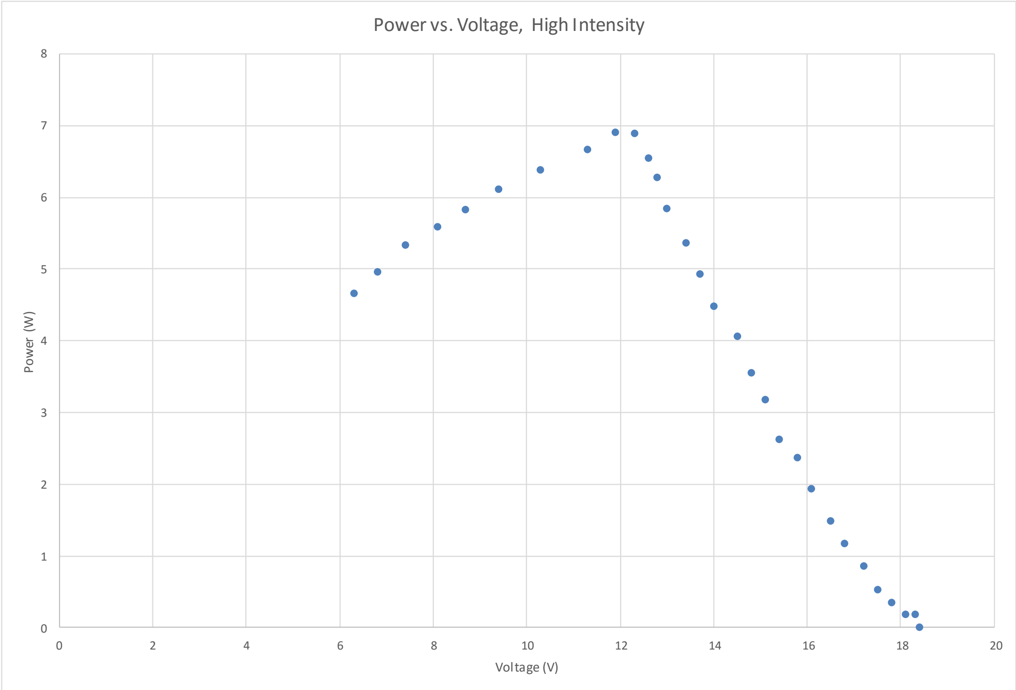

Figure IV.2: The power vs. voltage characteristic for a high intensity of light.

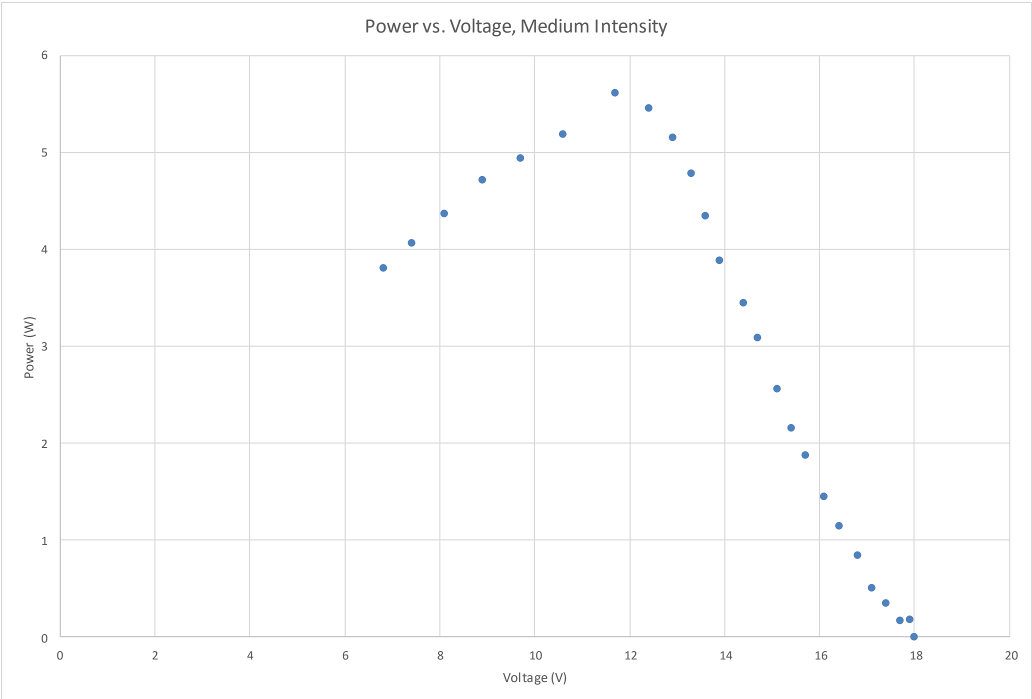

Figure IV.3: The power vs. voltage characteristic for a medium intensity of light.

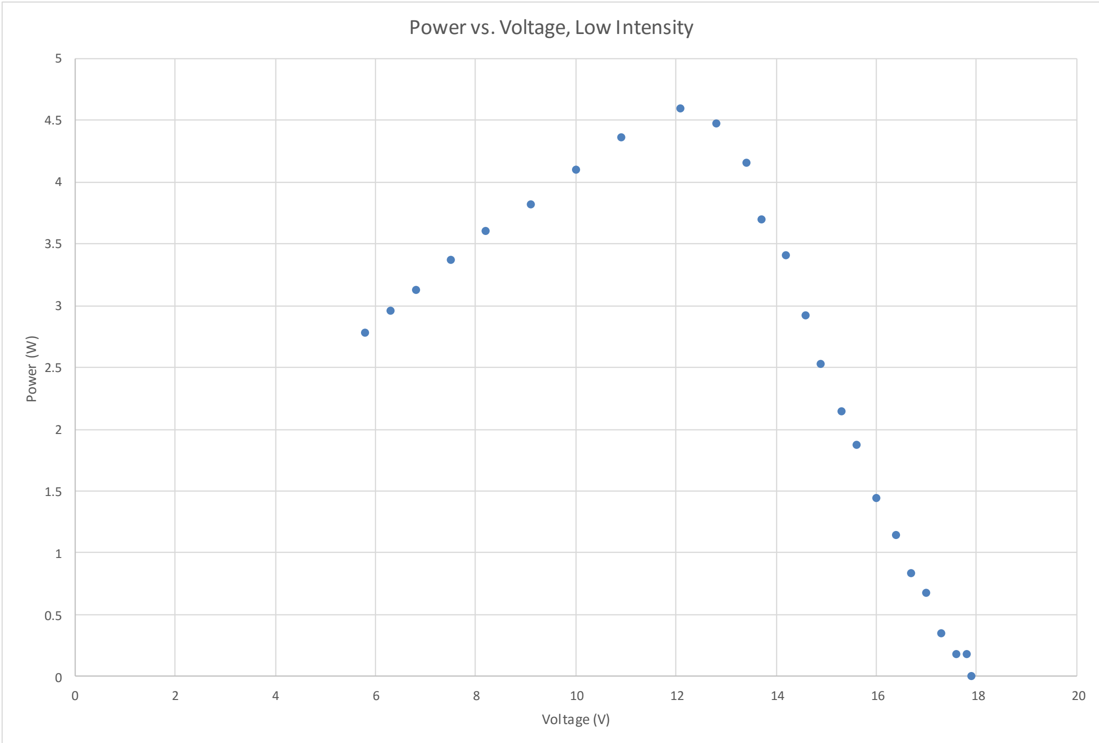

Figure IV.4: The power vs. voltage characteristic for a low intensity of light.

From these data points, a plane can be constructed. Then, measuring the open circuit voltage and the temperature can allow the voltage at the maximum power point to be interpolated.

Next, the panel voltage was measured at incrementing duty ratios. The voltage at a specific duty ratio was recorded. This allowed a voltage vs. duty ratio characteristic to be created and a linear trend to be approximated. This allowed for setting a specific duty ratio in order to achieve a specific voltage at a given set of conditions. These are shown in Figure IV.5 and Figure IV.6.

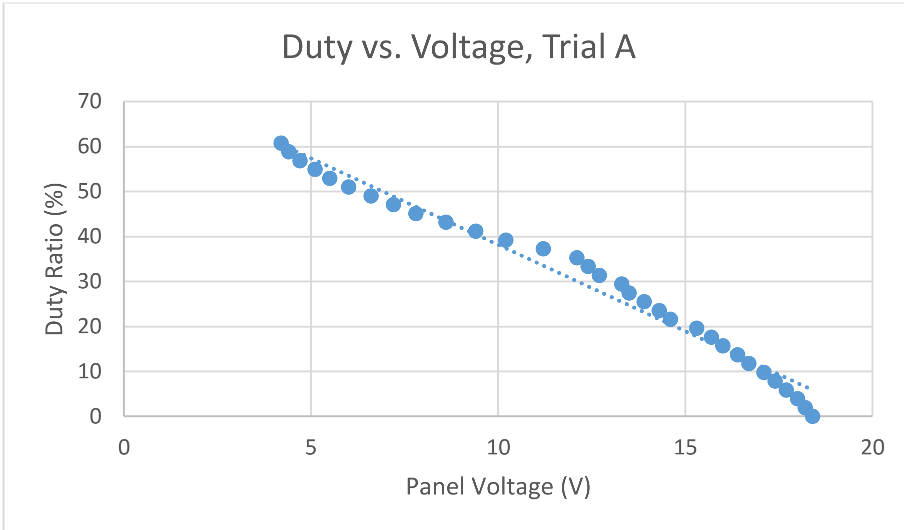

Figure IV.5: The duty ratio required to achieve a desired voltage in trial A.

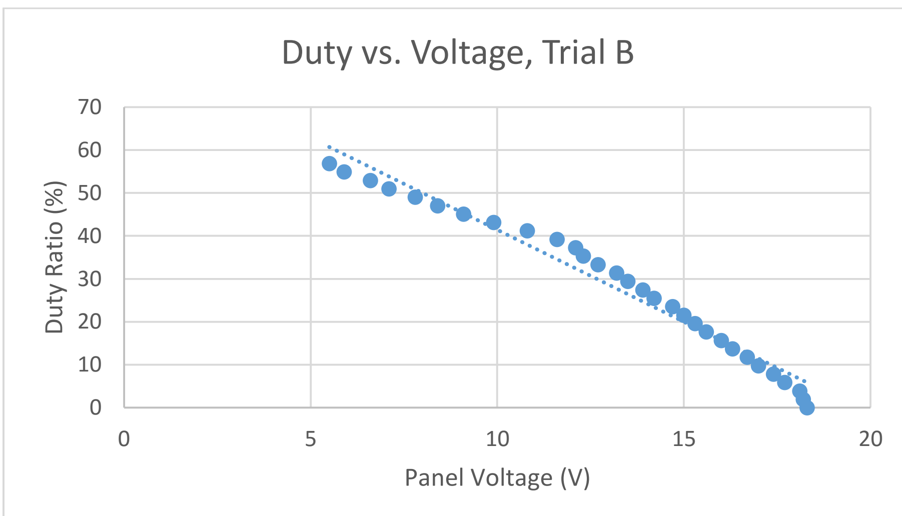

Figure IV.6: The duty ratio required to achieve a desired voltage in trial B.

At the same time, the power and voltage characteristic of the panel was created. This would allow the author to determine if the algorithm correctly tracked to the maximum power point voltage. This is shown in Figure IV.7.

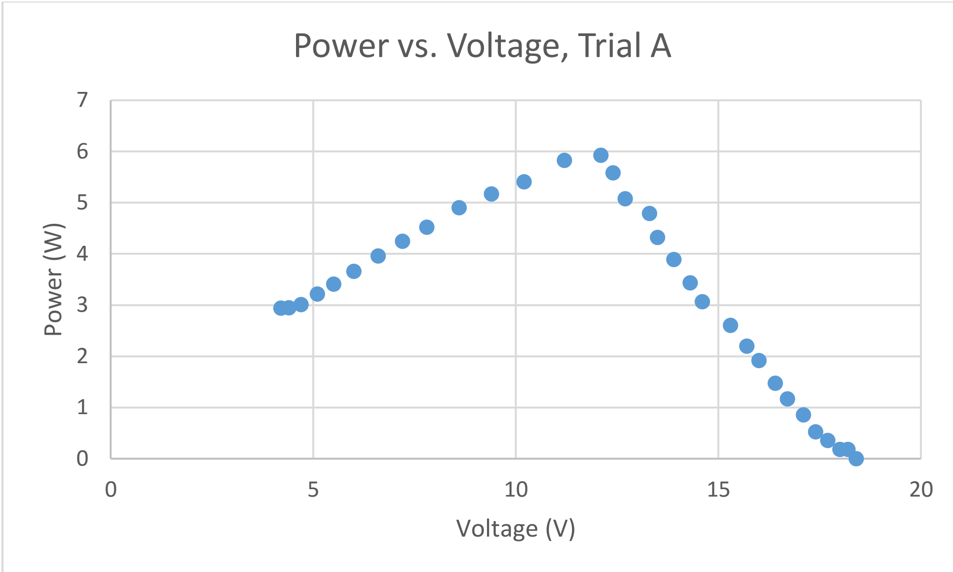

Figure IV.7: The power and voltage characteristic for trial A.

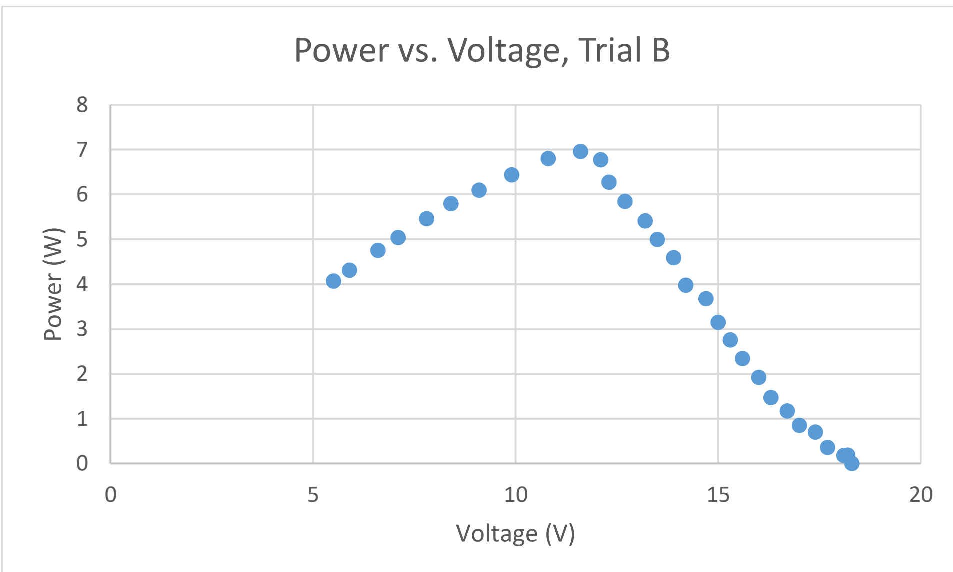

Figure IV.8: The power and voltage characteristic for trial B.

V. Experimental Results

In Trial A, the author experimentally determined the maximum power point voltage to be approximately 12.1 V. After the author ran the calibration to determine the appropriate duty ratio to achieve a specific voltage, the microcontroller stepped through the program and the author recorded the iteration, load voltage, panel voltage, and duty ratio. Here, "iteration" refers to any time the duty ratio was adjusted; this corresponds to each time the microcontroller must take measurements, make a decision, and change the duty ratio. The load voltage was recorded because, with a simple resistive load, the load voltage is sufficient to determine the load power. The panel voltage vs. the iteration number is shown in Figure V.1.

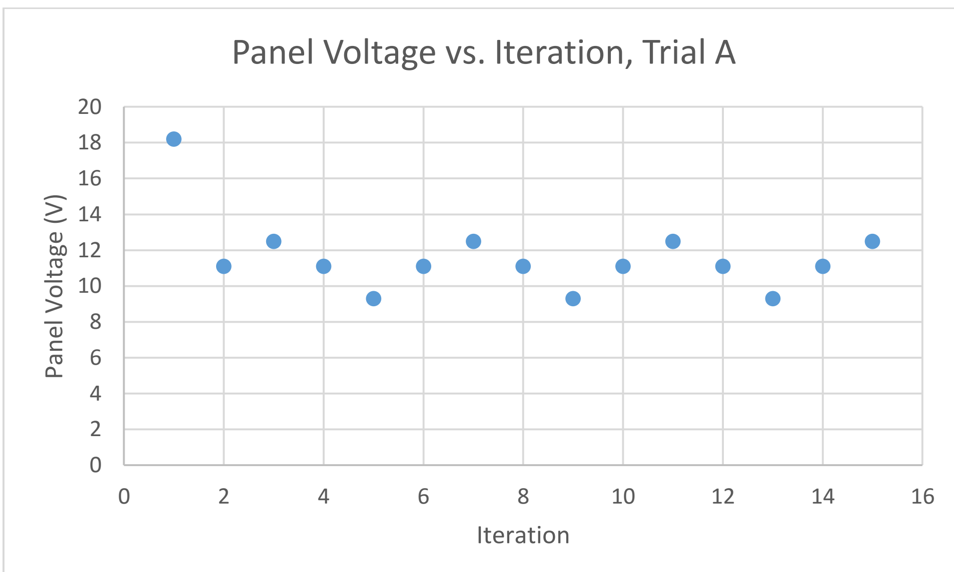

Figure V.1: The panel voltage vs. the iteration for trial A.

As can be seen in Figure V.2, the tracker immediately sets the panel voltage to approximately the maximum power point voltage and begins oscillating immediately. The beginning steps of attempting to climb to the maximum power point are skipped entirely. Further, the oscillations occur around a panel voltage between 11.1 and 12.5, correctly estimating the measured maximum power point. Each decision made by the microcontroller can be seen in Figure V.2.

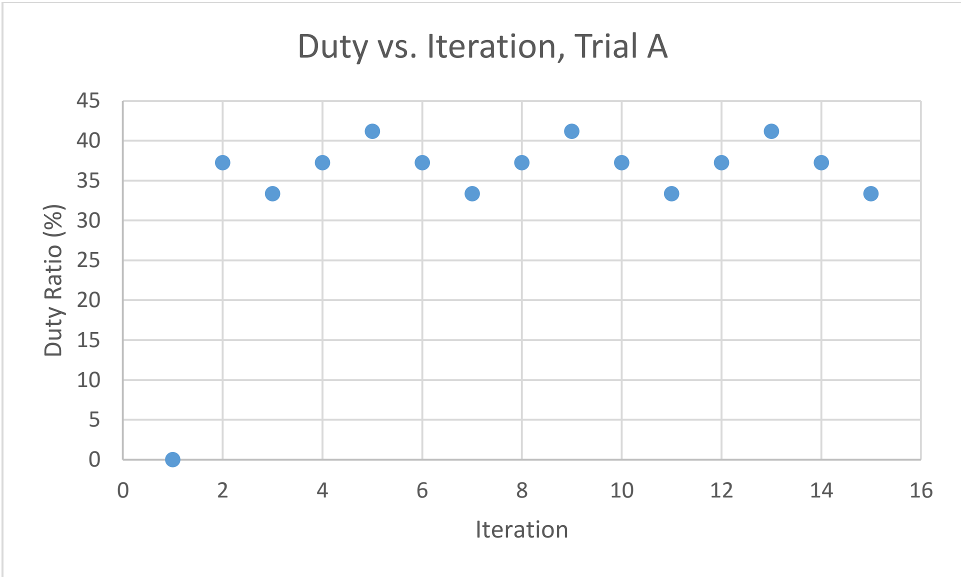

Figure V.2: The duty ratio vs. the iteration for trial A.

Further, it can be seen that the controller makes the correct decisions by observing Figure V.2 and Figure V.3. When increasing the duty ratio causes the load voltage to go down, the next decision made is to decrease the load voltage, and vice versa.

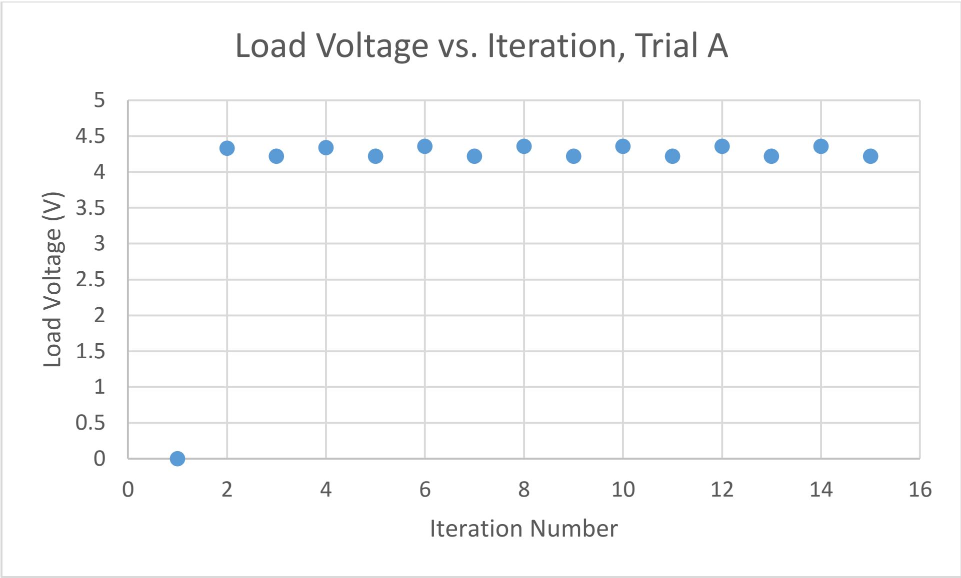

Figure V.3: The load voltage vs. the iteration for Trial A.

After the intensity of light was changed and the panel was allowed to settle to a different temperature, the author re-characterized the duty ratio vs. voltage of the panel and conducted Trial B. from the characterization, the author determined the maximum power point voltage was approximately 11.6 V. The panel voltage vs. the iteration is shown in Figure V.4.

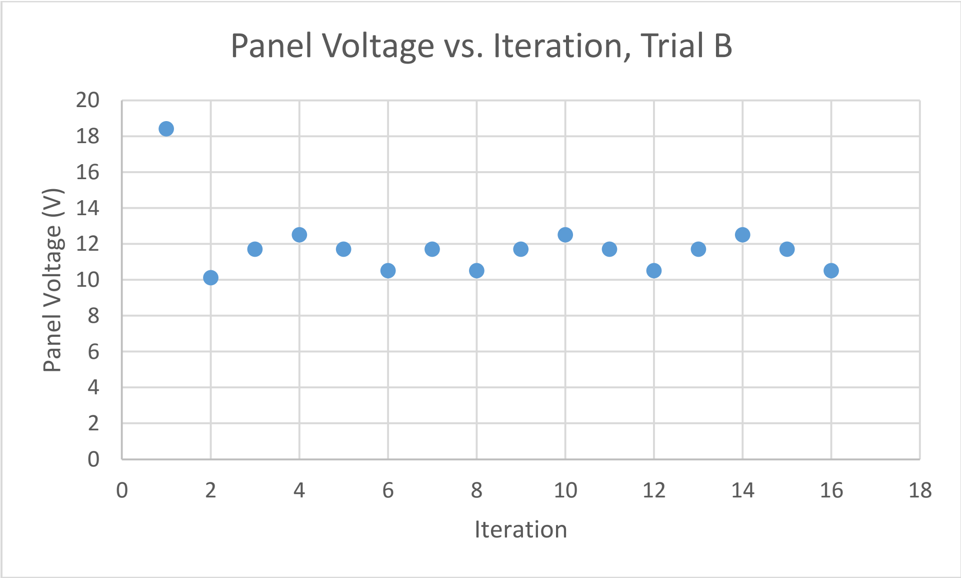

Figure V.4: The panel voltage vs. the iteration for Trial B.

Once again, the tracker immediately sets the panel voltage to a point that is approximately the maximum power point and begins oscillating immediately around the 11.7 V point, very close to the experimentally determined 11.6 V point. Each decision made by the controller can be seen in Figure *V.5*.

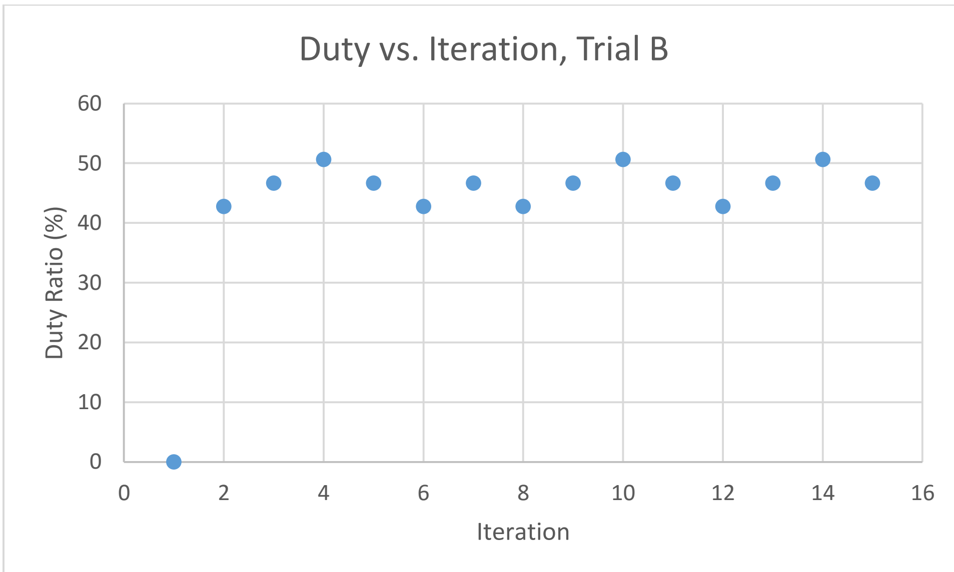

Figure V.5: The duty ratio vs. the iteration for Trial B.

Coupled with Figure V.6, it can be seen that the controller mostly makes the correct decisions. However, the difference in the load voltage at a duty ratio between 42% and 46% is so small, the controller occasionally makes a step in the wrong direction that has a relatively minor effect on the load power. This likely means that the maximum power point is between 10.5 V and 11.7 V, and these two points correspond to a roughly equal load power.

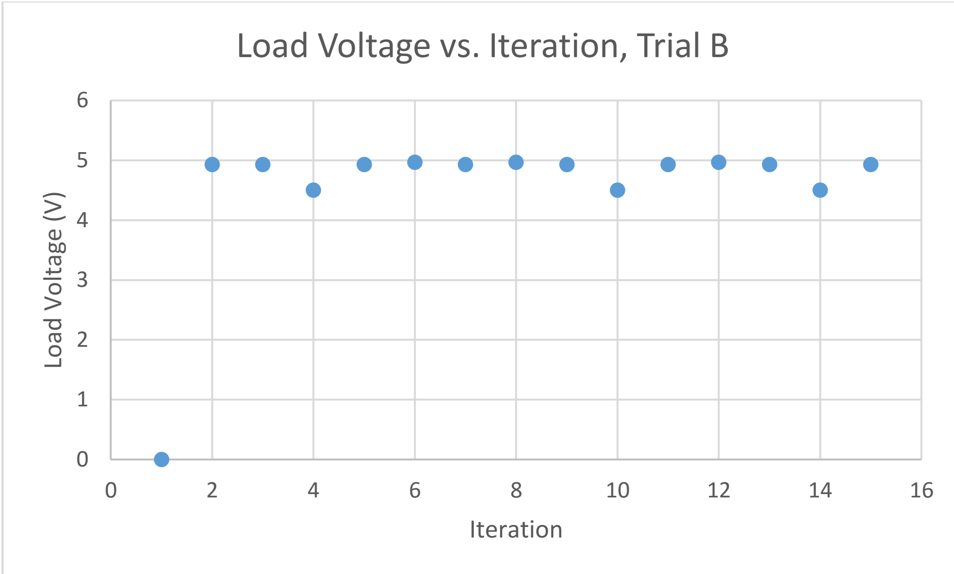

Figure V.6: The load voltage vs. the iteration for Trial B.

VI. Conclusions and Future Work

The proposed method successfully tracked to the maximum power point of the solar panel under several conditions. Further, it was able to reduce the number of steps required to reach steady state oscillations; in the conditions tested, the operating point was set and the steady state oscillations began immediately, requiring no climbing. Under similar conditions and with the same step size, the perturb and observe algorithm alone required 10-12 increments before beginning steady state oscillations. Thus, the new algorithm shows a marked improvement.

Further, the new algorithm does not require complex mathematical concepts, relying only on simple interpolation. Additionally, no knowledge of the panel datasheet values is required. While some pre-calibration is necessary, this can be done quickly for any panel or module. The algorithm also employs feedback to ensure the maximum power point is actually reached, and not just estimated. All of this was accomplished with only the addition of a temperature sensor; the sensor is a simple thermistor which is cheap in both power consumption and cost.

There is ample room for future work on this project. Among the simplest future work that can be done is to repeat the experiment over a wider range of conditions. The experimental set up the author used was sufficient to show proof of the effectiveness of the algorithm over a small range. However, it was insufficient to provide a wider range of conditions to prove greater generality. Performing further experiments would likely require many more calibrating data points, and would require more complex code to determine which data points between which to interpolate.

Further, the experimental process could be improved with a microcontroller with greater resolution. Because the ADC channels are only 8-bit, the resolution of the load voltage measurement is relatively low. This means that small changes in the load power are difficult to detect. This presented two problems. First, the microcontroller would occasionally make an incorrect incrementing decision when the load power changed by only a very small amount. Second, the duty ratio incrementation had a lower limit. Anything below several percent at a time would result in load power changes that were often too small for the microcontroller to reliably measure. In order to truly determine if the proposed algorithm is more effect than the constant voltage and perturb and observe combination, small duty ratio increments are needed to more finely track to and oscillate around the maximum power point.

An additional benefit to buying a more sophisticated microcontroller would be the ability to implement more sophisticated automated features. The program as it stands was large enough to fill the PIC16F716 memory. Because of this, much of the calibration was done manually. The duty ratio vs. voltage linearization was done manually, with the resulting linear relationship added to the program as needed. Each calibration data point was also collected and entered manually. With a more sophisticated controller, the duty ratio estimation could be done automatically when conditions change significantly. Also, additional calibration data points could be collected during normal operation.

Additional future work may be done changing the algorithm itself. In this experiment, the open circuit voltage and temperature were measured. Since the temperature and the solar current are the two factors that influence the location of the maximum power point, they must both be accounted for in a calculation that estimates the maximum power point. However, the open circuit voltage depends linearly upon temperature and only logarithmically on the solar current. Thus, while the effect of temperature could be easily accounted for by direct measurement, the effect of the change in insolation was harder to measure. Large changes in the insolation only had a small effect on the open circuit voltage. This could be remedied by performing experiments where the two parameters measured are the temperature and the short circuit current. The short circuit current is approximately equal to the solar current, which is linearly proportional to the irradiance. It is the author's belief that measuring short circuit current and temperature will allow for better estimation of the maximum power point, and thus greater efficiency in the tracker.

An additional area of research would be on the issue of partial shading. When partially shaded, there may be several relative maxima and one absolute maximum on the power vs. voltage curve. Since the absolute maximum may appear at a location not predicted by the tracker, this may result in tracking to an incorrect position. One possibility for future research is in applying this tracking scheme to a sub-panel tracker. That is, if this tracker is applied to each cell individually as opposed to the panel as a whole, the effects of partial shading will be less severe. It is less likely that each cell will be partially shaded, and the loss in power from tracking to a local maximum will be less severe on an individual cell than on a whole panel.

References

[1] N. Femia, Power Electronics and Control Techniques for Maximum Energy Harvesting in Photovoltaic Systems, Boca Raton: CRC Press, Taylor & Francis Group, 2013.

[2] S. Esterly, 2013 Renewable Energy Data Book, Golden, CO: National Renewable Energy Laboratory (NREL), 2014.

[3] P. Beiter, 2014 Renewable Energy Data Book, Golden, CO: National Renewable Energy Laboratory (NREL), 2015.

[4] Solar Energy Industries Association, "Solar Energy Industries Association," 17 12 2014. [Online]. Available: http://www.seia.org/research-resources/solar-industry-data. [Accessed 9 3 2016].

[5] T. Bennet and A. Zilouchian, "Developing a Photovoltaic MPPT System," Proquest Dissertations and Theses, 2012.

[6] B. Subudhi and R. Pradhan, "A Comparative Study on Maximum Power Point Tracking Techniques for Photovoltaic Power Systems," *IEEE Transactions on Sustainable Energy,* vol. 4, no. 1, pp. 89-98, 2013.

[7] G. Cipriani, V. Di Dio, F. Genduso and R. Miceli, "A New Modified Inc-Cond MPPT Technique and its Testing an a Whole PV Simulator Under PSC," *IEEE Applied Power Electronics Conference and Exposition,* pp. 3060-066, 2015.

[8] B. Subudhi and R. Pradhan, "Characteristics Evaluation and Parameter Extraction of a Solar Array Based on Experimental Analysis," *2011 IEEE Ninth International Conference on Power Electronics and Drive Systems,* pp. 340-344, 2011.

[9] W. Xiao and W. Dunford, "A Modified Adaptive Hill Climbing MPPT Method for Photovoltaic Power Systems," *IEEE 35th Annual Power Electronics Specialists Conference,* vol. 3, pp. 1957-1963, 2004.

[10] K. Hussein, I. Muta, T. Hoshino and M. Osakada, "Maximum Photovoltaic Power Tracking: an Algorithm for Rapidly Changing Atmospheric Conditions," *IEEE Proceedings: Generation, Transmission, and Distribution,* vol. 142, no. 1, pp. 59-64, 1995.

[11] R. Coelho, F. Concer and D. Martins, "A MPPT Approach Based on Temperature Measurements Applied in PV Systems," *2010 9th IEEE/IAS International Conference on Industry Applications,* pp. 1-6, 2010.

[12] P. Kamkird, N. R. W. Ketjoy and S. Sukchai, "Investigation on Temperature Coefficients of Three Types of Photovoltaic Module Technologies Under Thailand Operating Condition," *Procedia Engineering,* vol. 32, pp. 276-383, 2012.

Appendix I: C Code for New Algorithm

```
/*
 * File:   PandOmain1.c
 * Author: John
 *
 * Created on January 7, 2016, 1:30 PM
 */

#include <stdio.h>
#include <stdlib.h>
#include <xc.h>
#define _XTAL_FREQ 8000000
#pragma config FOSC = RC        // Oscillator Selection bits (RC oscillator)
#pragma config WDTE = OFF        // Watchdog Timer Enable bit (WDT disabled and can be enabled by SWDTEN bit of the WDTCON register)
#pragma config PWRTE = OFF      // Power-up Timer Enable bit (PWRT disabled)
#pragma config BOREN = OFF      // Brown-out Reset Enable bit (BOR disabled)
#pragma config BODENV = 25      // Brown-out Reset Voltage bit (VBOR set to 2.5V)
#pragma config CP = OFF        // Code Protect (Program memory code protection is disabled)
//#pragma  CONFIG1L=0b00110011;   //CONFIG1L is an 8 bit register that controls
                        //things.  Bit 1 is the brown-out reset
                        //voltage bit (2.5 at 0), Bit 6 is the
                        //brown-out reset selection bit (0=disabled)
                        //bits 5-4 are unimplemented. Bit 3 is the power
                        //up timer enable bit (0=disabled). Bit 2 is the
                        //watchdog timer enable bit (0 is disabled)
                        //Bit 1-0 is the oscillator selection bit. This
                        //controls the type of external oscillator.
                        //11 = RC oscillator
                        //10 = HS oscillator
                        //01 = XT oscillator
                        //00 = LP oscillator
                        //For the time being, we set clock to RC osc
#define DUTYCYCLE (CCPR1L)  //makes setting the duty cycle easier
//const int DUTYINCREMENT = 10;
/*PIN MAP:
 * 1: RA2/AN2                18: RA1/AN1
 * 2: RA3/AN3/VREF               17: RA0/AN0
 * 3: RA4/TOCKI                16: OSC1/CLKIN
 * 4: MCLR/VPP                 15: OSC2/CLKOUT
 * 5: VSS                    14: VDD
 * 6: RB0/INT/ECCPAS2             13: RB7/P1D
```

```
 * 7: RB1/T1OSO/T1CKI            12: RB6/P1C
 * 8: RB2/T1OSI                  11: RB5/P1B
 * 9: RB3/CCP1/P1A               10: RB4/ECCPAS0
 */
int getVoltage();
int getCurrent();
int tempMeas();
int getVoltage1();
int adjustDuty(int newv, int oldv, int oldincrement);
int main(int argc, char** argv) {
   TRISB = 0x00;   //The TRISA and TRISB registers determine if a pin is an
               //output or input. 0 indicates output, 1 indicates input; set all portB pins to output
   ADCON1 = 0b00000000;     //ADCON1 is the analog/digital control register. bits
                     //7-3 are unused; bits 2-0 configure the a/d port
                     //000=RA3-RA0 are analog pins. VDD is the reference
                     //001=RA3 is the reference, RA2-RA0 are analog pins
                     //100=RA2=Digital, others=analog, VDD is reference
                     //101 RA3 is reference, RA2 is digital, others analog
                     //11x RA3-RA0 are digital
                     //so i've set pins RA3-RA0 as analog pins
   ADCON0 = 0b01000000;     //ADCON0 is also a A/D configuration register
                     //bits 7-6 control the conversion clock
                     //00 = Fosc/2
                     //01 = Fosc/8
                     //10 = Fosc/32
                     //11 = Frc (internal ADC RC oscillator)
                     //so I've set A/D clock to Fosc/8
                     //Bits 5-3 are the analog channel select bits
                     //000 = AN0
                     //001 = AN1
                     //010 = AN2
                     //011 = AN3
                     //do not use other combinations
                     //So I've set it to use analog channel AN0
                     //Bit 2 is the Go/Done bit. Setting it to 1 starts a
                     //conversion. This bit is cleared to 0 when the
                     //conversion has completed.
                     //Bit 1 is unimplemented.
                     //Bit 0 enables or disables the ADC. Setting it to 1
                     //enables the ADC. Zero disables it. So I've enabled
                     //it.
   T2CON = 0b01111100;    //T2CON is a timer control register. it determines
                     //things like PWM frequency.
                     //Bit 7 is unused.
                     //Bit 6-3 are the postscaler select bits, which scale
                     //the overall oscillator.
```

```
                    //0000 = 1:1 postscaler
                    //0001 = 1:2
                    //0010 = 1:3
                    //0011 = 1:4
                    //0100 = 1:5
                    //etc
                    //1111 = 1:16 postscaler. so I've set it to 1:16
                    //Bit 2 is the timer2 on bit. 1 = Timer 2 is on
                    //Bit 1-0 are the clock prescaler bits
                    //00 = prescaler is 1
                    //01 = prescaler is 4
                    //11 = prescaler is 16
PR2 = 0b11111000;   //PR2 controls the PWM period. The period is
                    //[(PR2)+1]*4*Tosc*(Timer2 prescale)
                    //it also helps determine the duty cycle. The
                    //denominator of the duty cycle is 4(PR2+1)
                    //for our purposes, PR2 is currently a "relative"
                    //way to set the period; since we are using an RC
                    //oscillator for the external clock, the value of
                    //Tosc is not known a priori; it must be measured.
CCP1CON = 0b00001100;  //The CCP1CON register controls the Compare/capture/
                    //pwm module.
                    //Bits 7-6 are the PWM output configuration bits.
                    //00 = single output, P1A modulated, P1B,C,D are
                    //normal port pins
                    //01 = P1D modulated, P1A active, P1B&C inactive
                    //10 = P1A,B modulated with dead band control, P1C,D
                    //assigned as port pins
                    //11 = P1B modulated, P1C active, P1A,D inactive
                    //so I've chosen P1A as our output
                    //Bits 5-4 are the least significant bits of the PWM
                    //duty cycle. They are concatenated on to CCPR1L, all
                    //8 bits of which are the 8 most significant bits of
                    //the duty cycle.
RA0 = 1;                //set Pin A1 to input (pin 18 on the pic)
RA1 = 1;                //set pin A0 to input (pin 17 on the pic)
RA3 = 1;                //set pin A2 to input (pin 1 on the pic)
int newv;
newv = 0;
int oldv;
oldv = 0;
int duty;
duty=10;
int oldduty;
oldduty = 0;
int dutyinc;
```

```
    dutyinc = 0;
    int oldincrement;
    int getv;
    oldincrement = -10;
 //   int Voc;
 //   Voc = 17;
    long NewVoc;
    NewVoc = 0;
//    int avalue = -46;
//    int cvalue = -11;
//    long Aterm;
    long vholder = 0;
    long Vpv;
    int VocCalc;
    VocCalc = 0;
    Vpv = 0;
    while (1){
         DUTYCYCLE =200;  //change!!!!
         int temp = tempMeas();   //measure the temperature
//        RB1 = 1;             //close the FET to get the short circuit current
//        currentmeasure = getCurrent();  //get the current
//        Isc = (currentmeasure-128)*.156;
        NewVoc = getVoltage1();
        VocCalc = NewVoc*5*9/255;  //9 is the resistor ratio
//        RB1 = 0;             //open the FET
//        Aterm = avalue*VocCalc;
//        int VmppEst  = (-1000+VocCalc*46-temp/10)/-11; oldest one
        int VmppEst = (-100+VocCalc*8-2*temp)/-3;  //newer one
  //      int VmppEst = (-100+VocCalc*9-2*temp)/-1;   //newest one
        if (VmppEst > VocCalc){
            VmppEst = VocCalc;
        }
        if (VmppEst < 0){
            VmppEst = 0;
        }
 //       duty = -7*VmppEst+230;  //full light
//        duty = -6*VmppEst+188;   //.7Isc
 //       duty = -10*VmppEst+195; //Trial A
        duty = -12*VmppEst+229;
        DUTYCYCLE = duty;
//        duty = 60;
//        Vpv = getVoltage1();
//       Vpv = Vpv*5*9/255;
//        int counter = 0;
 /*       while (abs(Vpv-VmppEst)>1 && counter <5){
            if (Vpv-VmppEst < -3){
```

```
        DUTYCYCLE = DUTYCYCLE - 20;
        duty = duty -20;
    }
    else if (Vpv-VmppEst > 3){
        DUTYCYCLE = DUTYCYCLE + 20;
        duty = duty +20;
    }
    else if (1<=(Vpv-VmppEst)&&(Vpv-VmppEst)<=3){
        DUTYCYCLE = DUTYCYCLE + 5;
        duty = duty +5;
    }
    else if (-1>=(Vpv-VmppEst)&&(Vpv-VmppEst)>=-3)
    {
        DUTYCYCLE = DUTYCYCLE - 5;
        duty = duty -5;
    }
    Vpv = getVoltage1();
    Vpv = Vpv*5*9/255;
    counter ++;
}*/

for(int i =0;i<25;i++){
    vholder = 0;
    for(int k=0;k<10;k++){
        getv = getVoltage();
        vholder+=getv;
        _delay(100);
    }
    newv=vholder/10;
    _delay(100);
    dutyinc = adjustDuty(newv,oldv,oldincrement);
    duty+=dutyinc;
    if (duty>0 && duty < 256){
        DUTYCYCLE = duty;   //Set duty cycle
    }
    else if (duty<=0){
        dutyinc = 10; //dutyincrement;
        DUTYCYCLE = 10;
        duty = 10;
    }
    else if (duty >= 256)
    {
        duty -= 10;
        dutyinc = -10;
        DUTYCYCLE = duty;
    }
```

```
            oldduty = duty;
            oldv = newv;
            oldincrement = dutyinc;
//          _delay(200000);
        }
    }
    return (EXIT_SUCCESS);
}


int getVoltage(){
//   ADCON0[5:2] = 0b0001;
    ADCON0 = 0b00001101;//set the analog channel to A0, start the conversion
    int newvoltage = ADRES;
    return newvoltage;
}


int getVoltage1(){
 //   ADCON0[5:2] = 0b0011; //set the analog channel to A3, start the conversion
    ADCON0 = 0b00011101;
    int newvoltage1 = ADRES;
    return newvoltage1;
}


/*int adjustDuty(int newv,int newi,int oldv,int oldi, int oldduty,int newduty){
    int newpower = newv*newi;
    int oldpower = oldv*oldi;
    int increment;
    int pdifference = newpower-oldpower;
    if (pdifference>0 && newduty>oldduty)
        increment = DUTYINCREMENT;
    else if (pdifference<0 && newduty>oldduty)
        increment = -DUTYINCREMENT;
    else if (pdifference>0 && newduty < oldduty)
        increment = -DUTYINCREMENT;
    else if (pdifference<0 && newduty > oldduty)
        increment = DUTYINCREMENT;
    else
        increment = DUTYINCREMENT;
    return increment;
}
*/
int adjustDuty(int newv,int oldv,int oldincrement){
    long newpower;
```

```
    newpower = newv*newv;
    long oldpower;
    oldpower = oldv*oldv;
    int increment;
    int pdifference;
    pdifference = newpower-oldpower;
    if (pdifference>1 && oldincrement >0)
        increment = 10;   //DUTYINCREMENT
    else if (pdifference<=-1 && oldincrement > 0)
        increment = -10;
    else if (pdifference>1 && oldincrement < 0)
        increment = -10;
    else if (pdifference<= -1 && oldincrement < 0)
        increment = 10;
    else
        increment = oldincrement;
    return increment;
}
int tempMeas(){
        ADCON0 = 0b00000101; //AN0
        int tempmeas = ADRES;
        int tempvolt = 50*tempmeas*4/256;
        int resistance = tempvolt*180/(200-tempvolt)*50;
        int approxtemp = (8700-resistance)/182+25;
        return approxtemp;
}
```