48550 RES - Development of an improved MPPT algorithm

Joel Goodwin 98055953
Robert Carey 99139382
William Rooke 12051342

October 20, 2019

UTS

UNIVERSITY OF TECHNOLOGY SYDNEY

Contents

List of Figures

List of Tables

1 Purpose

The purpose of this project is to utilise the technqiues learnt in the previous labs and this subject and produce an improved maximum power point tracking system. In the labs 3 & 4 we seen some of the most common techniques to create a MPPT system for solar power. The purpose of the improved MPPT system is to do the following.

1. To stay at the actual maximum power point regardless on the level of insolationstays at true MPPs regardless of the level of solar intensity (insolation)
2. It needs to respond effectively to changes in insolation and react quick to the new MPP

2 Introduction

The methods that we utilised in the labs were the constant voltage method and peturb and observe method. Although both of the methods are effective they do have some disadvantages which is the reason we need to propose an improved solution. Starting with the peturb and observe method which is probably the most common method used in MPPT systems. It is really hard to find the balance where the system reacts fast enough to changes in insolation but making sure that the step sizes are small enough so that the oscillations around the MPP are kept to a minimum reducing losses. Therefore we can't be great at both minimising response time but also minimising steady state oscillations. This is something that we propose to address with our proposed solution, looking at the Constant Voltage method this is possibly the simplest solution that involves comparing the measured voltage of the system compared to a reference voltage. The problem of this is that it doesn't consider any variations for things like temperature etc, this means calculations need to be calculated at various different locations in order to get a true MPP. It can be a very fast method to get near the MPP, however it is generally not a very accurate method should conditions change.

Therefore we need to design a solution that reacts faster like the constant voltage system however continually updates the MPP with the accuracy of P&O. Our method that we came up with be discussed later on in the report. As this was a group report it was important to work together as a team and delegate roles effectively, in this project we worked more in a collaborative approach where we worked together in labs. Although we work collaboratively, the main roles of each person can be seen below in the table.

Table 1: Project Roles

Name	**Role**
Robert Carey	**Initial research, Coding and general lab operations**
William Rooke	**Initial research, Coding and general lab operations**
Joel Goodwin	**Initial research, Project Report and general lab operations**

The dynamics of the group worked well and there were no issues throughout the lab. This is because good communication and respect among each other was practiced throughout the project to ensure a successful project.

3 Literature Review

3.1 Fuzzy Logic MPPT

3.1.1 What particular issue of MPPT has this paper [2] addressed?

Fuzzy logic MPPT addresses the power fluctuation issue which is a large downside to MPPT methods such as Perturb and Observe (P&O) and Incremental Conductance (IC) which stems from their oscillation about the maximum power point

(MPP) of the system. It also addresses the inability to compensate for sudden insolation change which is an issue with methods such as Short Circuit Pulse or Open-Circuit Voltage. Due to the large number of non-linearities and changing dynamics that exist as part of a solar PV system, more complex control systems, such as using Fuzzy Logic Control, have been developed to overcome these issues.

3.1.2 What is the algorithm of the proposed MPPT?

The algorithm changes the inputs of the system, change in PV voltage and change in PV output power, and the output of the system, the DC-DC converter's duty cycle, into fuzzy variables. Membership functions are also developed using a number of methods, including trial and error. Based on the fuzzy values and the membership functions of each value, a fuzzy rule is applied and the output is inferred. The value of the output is generated by aggregating any rules which have been used, and their inferred outputs, using either a centre of area method or by using a weighted average.

3.1.3 How does this MPPT algorithm address the issue?

By making the inputs and outputs of the system fuzzy variables, the control system is better equipped to handle sudden changes in insolation and the inherent non-linearities in a solar PV system. It also does not require an in-depth knowledge of the system model, meaning it is more robust.

3.1.4 What are the improvement shown from this MPPT?

Fuzzy logic control algorithms require no previous knowledge of the system and are more robust than more conventional MPPT algorithms, while also decreasing oscillation about the operating point of the system and responding faster to changing inputs and atmospheric conditions. Due to these improvements, the efficiency of the system is also increased.

3.1.5 Are there any potential issues with this proposed MPPT?

Although it is considered easier to implement than some other control systems, such as using particle swarm optimisation or a genetic algorithm, its comparative complexity compared to more conventional MPPT methods such as P&O are a barrier to its widespread implementation, as it requires more computational power and performance, as well as being more complex to implement.

3.2 MPPT Based on Sliding Mode Control

3.2.1 What particular issue of MPPT has this paper [3] addressed?

The sliding mode control model is to enhance the typical Perturb and Observe (P&O) and other simple algorithms which has issues with steady state error and speed of reaching the MPP. The sliding mode control is a non-linear controller that proposes to control non-linear systems which non-linearities and changing dynamics are presest in the PV system.

3.2.2 What is the algorithm of the proposed MPPT?

The algorithm is to use the control theory of sliding surfaces. For this system the purpose is to measure the change in the reference voltage and PV voltage and then use the control to change the duty cycle ratio. This surface is to project the system so that it will slide down the surface to the required output.

3.2.3 How does this MPPT algorithm address the issue?

This algorithm addresses the issue using the basics of sliding mode control. The reason for this it can handle nonlinear systems, also the system is quire robust so therefore it can deal with uncertainties and changes throughout the system. Overall the system converges much fast which is an issue of P&O and the system controls well with minimal steady state error which is an issue for both of the techniques earlier used.

3.2.4 What are the improvement shown from this MPPT?

This method accelerates the movement to the MPP when comparing with a traditional P&O method. This is dependant of the surfaces that are selected, if the surface chosen considers voltage, current and temp the results are much better. The respnose of the system is much faster and the steady state error is significantly reduced when comparing with traditional techniques.

3.2.5 Are there any potential issues with this proposed MPPT?

One of the main issues with a lot of technologies is the difficulty and time to implement the solution vs the actual performance enhancement. This technique enhances much simpler solutions like the P&O method which reinforces that the complexity in comparison to these types of methods are much more difficult. With more complex solutions comes more cost and time and this one of the main issues with this design.

3.3 Adaptive MPPT

3.3.1 What particular issue of MPPT has this paper [1] addressed?

This paper is attempting to address the two main issues of MPPT, those being the ability to respond quickly to varying irradiance as well as their ability to track the MPP without excessive oscillation. The ability of the algorithm to optimise these as much as possible results in the maximum achievable power being extracted from the attached PV panel.

3.3.2 What is the algorithm of the proposed MPPT?

The proposed MPPT algorithm consists of three sub algorithms. The first being the Current Perturbation Algorithm (CPA). It uses the concepts from the conventional P&0 algorithm but uses current instead of voltage. The second is an Adaptive Control Algorithm (ACA). This is used to determine an operating point much closer to the MPP when the system is exposed to sudden changes in irradiance. The final algorithm is the Variable Perturbation Algorithm (VPA). This algorithm is used to reduce the size of the perturbation steps every time the MPP is crossed.

3.3.3 How does this MPPT algorithm address the issue?

The CPA portion of the MPPT algorithm is taking care of keeping the PV panel operating around the MPP, it is the addition of the ACA and VPA that deal with main issues of oscillation and responsiveness. The ACA moves the operating point of the system closer to the MPP using coarse perturbation size when it sees a large change in irradiance. This increases the responsiveness of the system. The VPA dynamically reduces the perturbation steps when it sees a change in the polarity of the power. This effectively reduces the oscillations of the system around the MPP.

3.3.4 What are the improvements shown from this MPPT?

The main improvements shown from this MPPT are its reduction in the amount of oscillations once the region of the MPP is found. It also improves the responsiveness of the MPPT by increasing the size of the perturbation steps with large changes in irradiance.

3.3.5 Are there any potential issues with this proposed MPPT?

The potential issues of this MPPT are that it will still see some level of delay when exposed to changes in irradiance and some level of oscillation. While this is minimal compared to some other algorithms the potential loses could become much larger when scaled up to large solar systems.

4 Design of an Improved MPPT algorithm

After reviewing all of the initial research we decided to stick with more of an adaptive Peturb and Observe method we found this easiest to implement and the results of the algorithm sounded desirable. The main reason we went down this path is that two main issues of both the techniques done in previous labs is the fact of speed of the MPPT system and accuracy. So we thought that maybe we could try and combine the strengths of both of the P&O system and Constant Voltage. The strengths of each system suited the weaknesses of each other and this is what was desirable. This method certainly was made simpler as we knew what to expect with both systems as we had carried out them in previous labs.

When beginning to right the code it was always easier to begin with a diagram about what results we want to improve from our design. Reiterating what was discussed earlier the system parameters that we were after were the following

1. Improved response time to get to the MPP (Transient Response)
2. Reduced oscillations around the MPP (Steady state Response)

Our method that we came up with to use the Constant Voltage Reference for the value outside of the limits. Then inside of the limits it switched the P&O with a much smaller step size to ensure that steady state error was kept to a minimum with minimal power losses. This conceptually met our initial requirements as it allowed for faster speed to the reference utilising the strengths of the constant voltage method. This is done by using large increases of the duty cycle by 10% when outside the limits. Once the voltage reaches those limits it switches over to the normal P&O method with 1% duty cycle changes to reach the MPP. We know from lab 4 that this method uses the output power for this operation, it will then check if the new power value has increased or decreased its value and then alters the duty cycle accordingly this can be seen in the flow chart below.

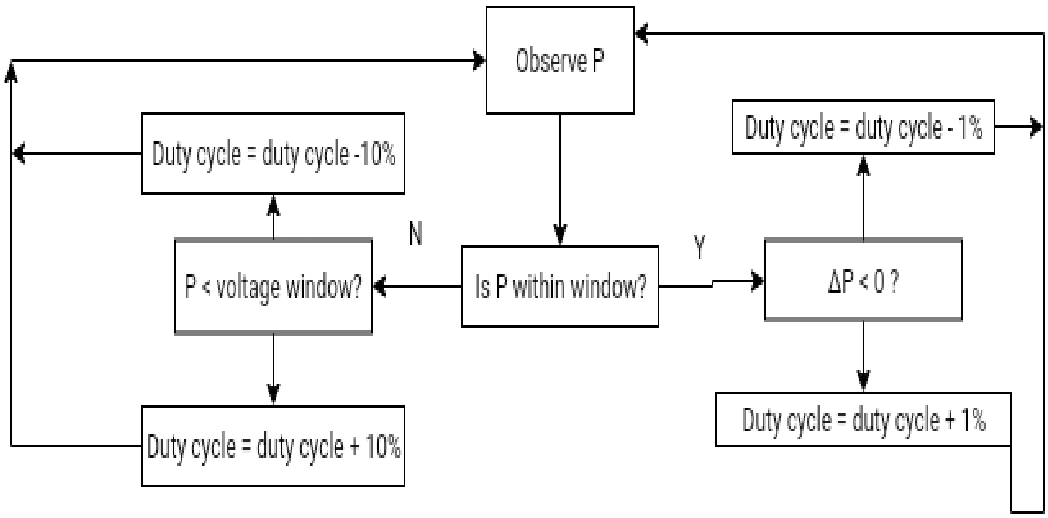

Figure 1: Algorithm Flowchart

After we came up with a flowchart for the system we needed to then write the code, the code that was implemented as can be seen below. The combines the codes that we have previously used in to one super code that can do both, it reads the ADC values of the output voltage and PV emulator voltage. After it reads the values it converts them to the corresponding voltage value, and checks to see what condition that value meets. If it is inside the constant voltage statements it increases/decreases the duty cycle by 10%. Once the voltage gets within the limits it will then calculate the new power and compare the old value and alter the duty cycle according. The stored value of the power is constantly updated to the new value of the power.

One thing that was quite important in our code was deciding on the limits for the MPPT algorithm, the limits that changed from the Constant Voltage section to the P&O section. To decide on the appropriate limits we looked at the results from lab 4 which is the power vs the voltage graph. Looking at the image we can see that graph can be seperated in terms of the response, looking at the power response from voltages 0 to 17V it increases linearly. Also from this point above to about 21V it becomes non linear and the changes are a lot smaller and therefore in this region we wanted to use the more accurate P&O section. Where as, in the initial stages the constant voltage technique will respond fast enough for the system. Therefore the limits that we chose to make the limits was 16V and 21V.

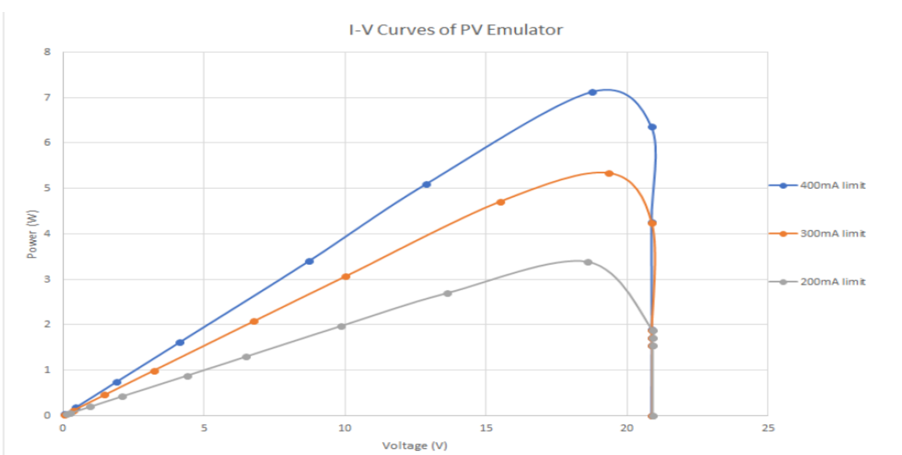

Figure 2: Algorithm Flowchart

```
    Vout_CV = analogRead(ADC_Vout) * 0.0049;
    Vout_adjV = Vout_CV / 0.18;
    Vpv_CV = analogRead(ADC_Vpv) * 0.0049;
    Vpv_adjV = Vpv_CV / 0.18;
    Serial.print("Out:");
    Serial.println(Vout_adjV);
    Serial.print("PV:");
    Serial.println(Vpv_adjV);

    if (Vpv_adjV < V1)
    {
      Duty = Duty - largeStep;
    }
    else if (Vpv_adjV > V2)


    {
      Duty = Duty + largeStep;
    }
    else //peturb
    {
      newP = Vout_adjV * Vpv_adjV;

      if (oldP > newP)
        Duty = Duty - smallStep;
      else
        Duty = Duty + smallStep;

      oldP = newP;
    }

    if (Duty > 90)
      Duty = 90;
    else if (Duty < 10)
      Duty = 10;

    setDutyCycle(Duty);

    waitTime = millis() + period;
  }
}
```

Figure 3: Algorithm for Improved MPPT

5 Results

To test the performance of the algorithm, the system was tested the same as Constant Voltage MPPT and the P&O method. Looking at the results below at different currents we can see that the duty cycle of the buck converter is yellow, the output voltage green and the input current is blue.

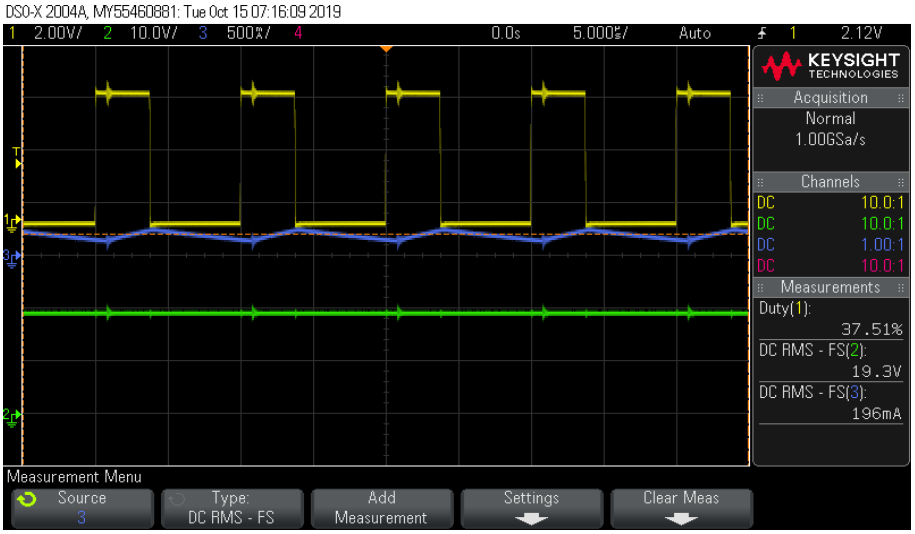

Figure 4: Duty Cycle At 200mA

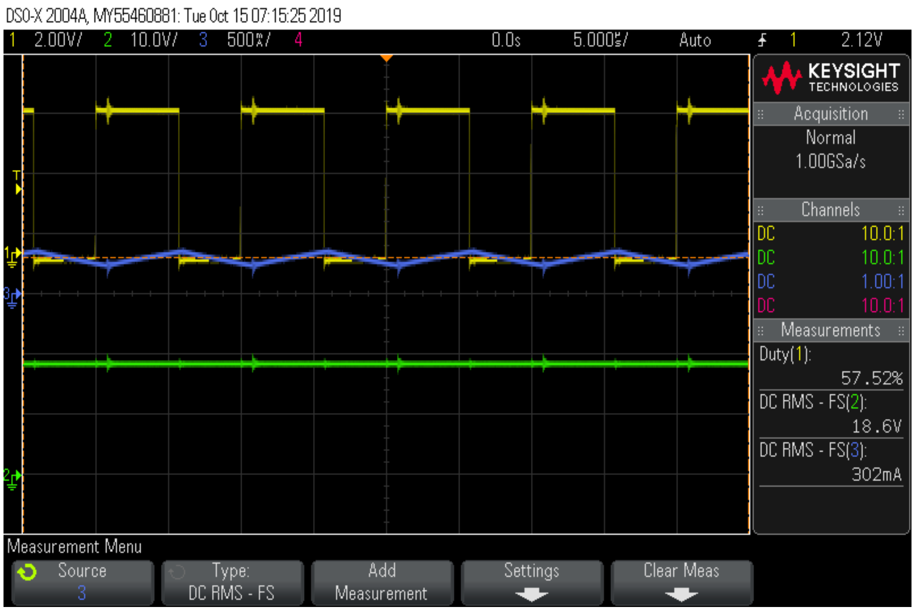

Figure 5: Duty Cycle At 300mA

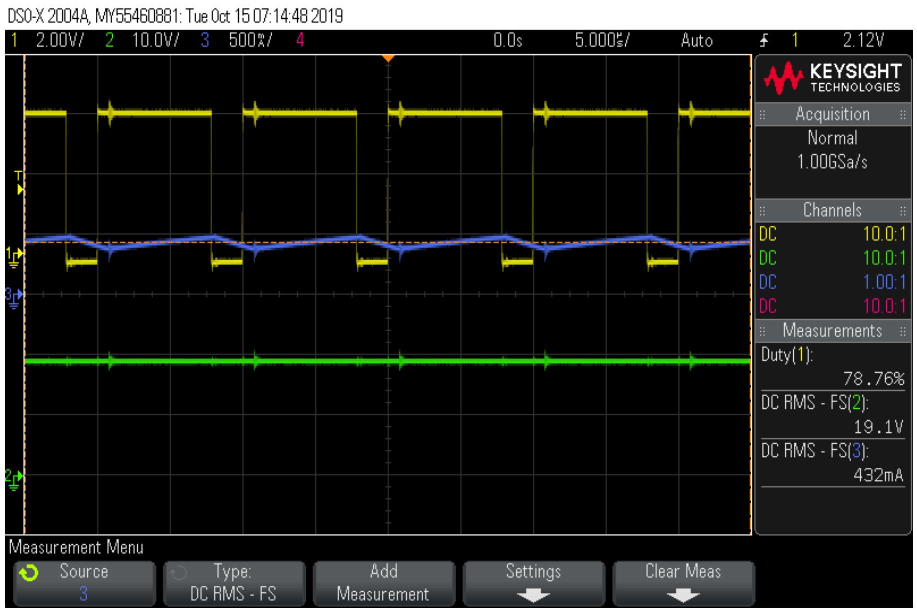

Figure 6: Duty Cycle At 400mA

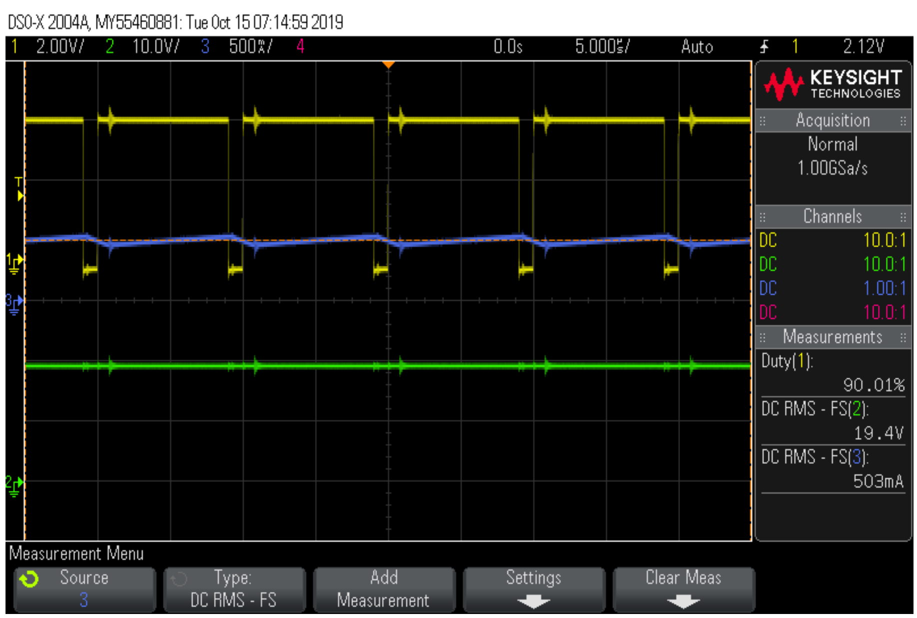

Figure 7: Duty Cycle At 500mA

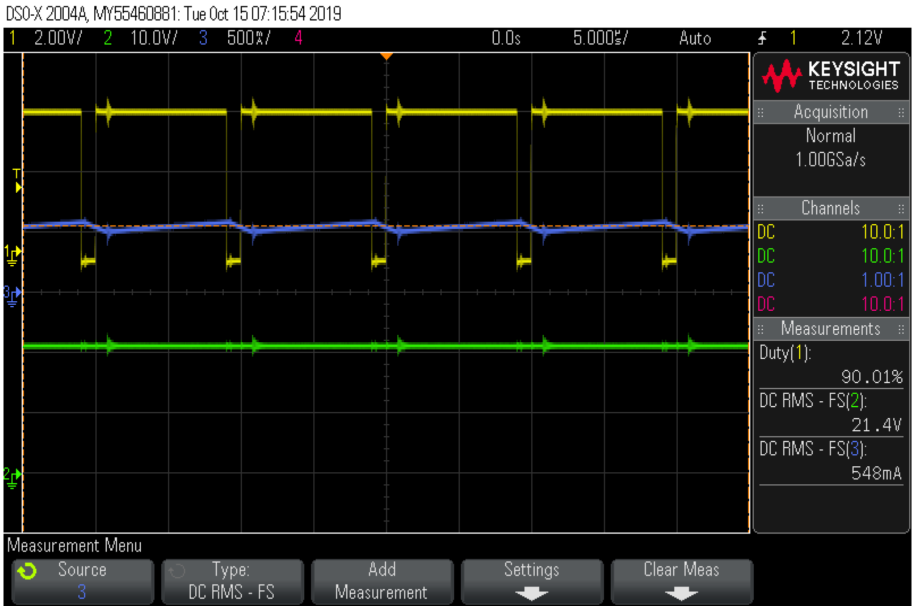

Figure 8: Duty Cycle At 528mA

On top of the oscillascope images we also got a .csv file of the changes to see how the system works numerically. An image of the file can be seen below. The middle value is the voltage and in the third colum is the current. We can see near the end that when the current is stable the voltage will oscillate with minor magnitude this proves that it is in the P&O section. Looking at the transient response we can see that the voltage changes fast when the current does and this reinforces that the system reacts first fast in the constant voltage section.

6.257813E+00	+11.206423E+00	+322.156E-03
6.250000E+00	+11.218986E+00	+322.659E-03
6.242188E+00	+11.231548E+00	+323.161E-03
6.234375E+00	+11.231548E+00	+323.412E-03
6.226563E+00	+11.244111E+00	+323.664E-03
6.218750E+00	+11.244111E+00	+323.664E-03
6.210938E+00	+11.244111E+00	+323.915E-03
6.203125E+00	+11.244111E+00	+323.915E-03
6.195313E+00	+12.726523E+00	+385.221E-03
6.187500E+00	+13.229036E+00	+381.453E-03
6.179688E+00	+13.229036E+00	+381.453E-03
6.171875E+00	+13.229036E+00	+381.453E-03
6.164063E+00	+13.241599E+00	+381.453E-03
6.156250E+00	+13.241599E+00	+381.453E-03

Figure 9: Voltage Change .csv

6 Analysis

Overall the performance of our algorithim worked as expected it enabled duty cycle of the buck converter to work effectively and once in the P&O loop in the code the oscillations were kept to a minimum resulting in greater performance. When the current was chnaged at a large rate the code was able to adapt fast to move outside of the P&O loop to the constant voltage loop. We were able to see this happen visually via the oscillascope but also we took .csv captures to see the changes numerically reinforcing what was seen visually. We can see from the .csv image above that when the current increased the voltage moved very fast and once in the P&O loop the oscillations around the MPP were very minimal. When comparing the results with lab 4 the response to the reference was very fast due to the large duty cycle steps at values far away from the MPP. However the oscillations were the same size, therefore this proves that it was successful as it had a faster transient repsonse however the steady state response was the same.

When comparing with the constant voltage the speed to get to the maximum voltage was very similiar, however this method was a lot closer to the MPP, which is important in the practical world to ensure that the power recieved is effecient. The more accurate the solution can result in great cost savings and effeciency of the system. The maximum voltage has it's advantages as does the P&O method, so combining the advantages to tackle the others disadvantages has worked great and ensured that the algorithm works as needed.

Something that could be added in the future to the algorithim would be another condition that is related to the voltage once it moves back outside the limits. The purpose of this section of the code would be to use the measured voltage value. Currently if the value is right near the limits once in the P&O section it may oscillate inside and outside the limits. This causes an issue because once the duty cycle changes and the voltage moves just outside of the limits it will then add 10% to the duty cycle which move the voltage a lot further away from the MPP. This is not ideal because it will then not run as effeciently and will take time to get back to the MPP. The code can fix this by stating once the value goes inside the limits for example the voltage between 12 to 14V it must go under 11 or over 15V to then move back to the Constant Voltage loop. This is important so that if the MPP is right at the normal limits it wont keep moving between techniques, it must change a lot in magnitude for it to change tecniques. This is something that could be easily implemented to improve performance of the algorithm.

7 Conclusion

Overall the project was a great success, it allowed the group to further enhance their understanding of MPPT. By combining further research with the skills already learnt, it allowed us to work similiar to a professional engineer in the workforce. Working on practical solutions allows for further understanding of the subject and ensures that the group must comprehend the technology in order for it to work. As discussed in previous sections the project was not only a success in terms of deeper learning but also the performance of our improved MPPT was great and it went exactly to plan. This was made possible by the collaborative approach from all of the team members, we dealt with any technical issues respectivefully and this ensured a successful project. Moving forward I believe that each of the group members can take something out this project whether it be working in a team environment or applying the technical knowledge in the field.

8 References

[1] S. K. Kollimalla and M. K. Mishra, "Variable perturbation size adaptive p o mppt algorithm for sudden changes in irradiance," *IEEE Transactions on Sustainable Energy* , vol. 5, no. 3, pp. 718–728, July 2014.

[2] Harrabi, Naziha et al. (2017). "Comparative study of fuzzy logic peak power trackers for a photovoltaic system". *In: 2017 18th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)* . IEEE, pp. 539–545.

[3] Vázquez, Nimrod., Azaf, Yuz., Cervantes, Ilse., Vázquez, Esli., & Hernández, Claudia. 2015, 'Maximum Power Point Tracking Based on Sliding Mode Control', *International Journal of Photoenergy* vol. 2015, pp. 1-8.

9 Appendix

```
#include <Arduino.h>
#include "PWM.h"

// Processor Frequency
int32_t clkFreq = 16000000;

// Constants
const int pin_PWM = 10;
const int ADC_Vout = A0; // Buck-Boost output pin
const int ADC_Vpv = A4;  // PV output voltage pin
const int defaultDuty = 50;
const int vref = 18;
const int32_t defaultFreq = 100000; //frequency (in Hz)

// Globals
int Duty = defaultDuty;
int32_t Freq = defaultFreq;
String Command;

// Gets value to set analogWrite function
int getAWrite(int32_t freq, int duty);
// Updates the duty cycle for pin_PWM
void setDutyCycle(int duty);
// Updates the frequency for pin_PWM
```

```
void setup()
{
  Serial.begin(9600);
  //initialize all timers except for 0, to save time keeping functions
  InitTimersSafe();
  //sets the frequency for the specified pin
  bool success = SetPinFrequencySafe(pin_PWM, defaultFreq);
  if (success)
  {
    pinMode(13, OUTPUT);
    digitalWrite(13, HIGH);
    pinMode(pin_PWM, OUTPUT);
  }
  pinMode(ADC_Vout, INPUT);
  pinMode(ADC_Vpv, INPUT);
  analogWrite(pin_PWM, getAWrite(defaultFreq, defaultDuty));
}

int getAWrite(int32_t freq, int duty)
{
  int32_t x = clkFreq / (2 * freq);

  return (x * duty) / 100;
}

void setDutyCycle(int duty)
{
  Duty = duty;
```

```
  analogWrite(pin_PWM, getAWrite(Freq, duty));
}

void setFreq(int32_t freq)
{
  Freq = freq;
  SetPinFrequencySafe(pin_PWM, freq);
  analogWrite(pin_PWM, getAWrite(freq, Duty));
}

unsigned long period = 1000;
bool toggle = 0;
unsigned long waitTime = 0;
double Vout_adjV = 0;
double Vpv_adjV = 0;
double Vout_CV = 0;
double Vpv_CV = 0;
double newP, oldP;
double V1 = 16;
double V2 = 21;
int largeStep = 10;
int smallStep = 1;

void peturb(double newP)
{


}

void loop()
{
  // time_now = millis();
  while (millis() > waitTime)
  {
    // toggle LED to indicate sample time
    DDRB |= (1 << PORTB5);
    if (toggle)
      PORTB |= (1 << PORTB5);
    else
      PORTB &= ~(1 << PORTB5);

    toggle = !toggle;
```

```
  Vout_CV = analogRead(ADC_Vout) * 0.0049;
  Vout_adjV = Vout_CV / 0.18;
  Vpv_CV = analogRead(ADC_Vpv) * 0.0049;
  Vpv_adjV = Vpv_CV / 0.18;
  Serial.print("Out:");
  Serial.println(Vout_adjV);
  Serial.print("PV:");
  Serial.println(Vpv_adjV);

  if (Vpv_adjV < V1)
  {
    Duty = Duty - largeStep;
  }
  else if (Vpv_adjV > V2)
```

```
    {
      Duty = Duty + largeStep;
    }
    else //peturb
    {
      newP = Vout_adjV * Vpv_adjV;

      if (oldP > newP)
        Duty = Duty - smallStep;
      else
        Duty = Duty + smallStep;

      oldP = newP;
    }

    if (Duty > 90)
      Duty = 90;
    else if (Duty < 10)
      Duty = 10;

    setDutyCycle(Duty);

    waitTime = millis() + period;
  }
}
```

Acknowledgement of Contribution

48550 Group Project

The form is to acknowledge the contribution of each member to the group project. You are required to fill out the following Table and sign your name if you agree or disagree with the **average contribution** from a member in your group. It is essential to complete the project mark for yourself and other members.

Date:20/10/2019

Name of the Group member	**Key Contribution(s)**	**Agree**	**Disagree**
Robert Carey	Initial research, Coding and general lab operations		
Joel Goodwin	Initial research, Project Report and general lab operations		
William Rooke	Initial research, Coding and general lab operations	W.	