Maximum Power Point Tracking (MPPT) for a Solar PV System

48550 Renewable Energy Systems - Labs 3 and 4

Robert Carey 99139382
Joel Goodwin 98055953
William Rooke 12051342

September 22, 2019

UNIVERSITY OF TECHNOLOGY SYDNEY

Contents

List of Figures

1 Introduction

The purpose of this report is to develop two Maximum Power Point Tracking (MPPT) algorithms, Constant Voltage Reference and Perturb and Observe, for use in a solar photovoltaic (PV) renewable energy system. In place of a functional PV panel, a PV emulator, consisting of a string of diodes, is used in order to make the experiments easier to conduct and the algorithms easier to troubleshoot.

2 Constant Voltage Reference MPPT

2.1 Pre-Work

2.1.1 Explain the working principle of the PV emulator.

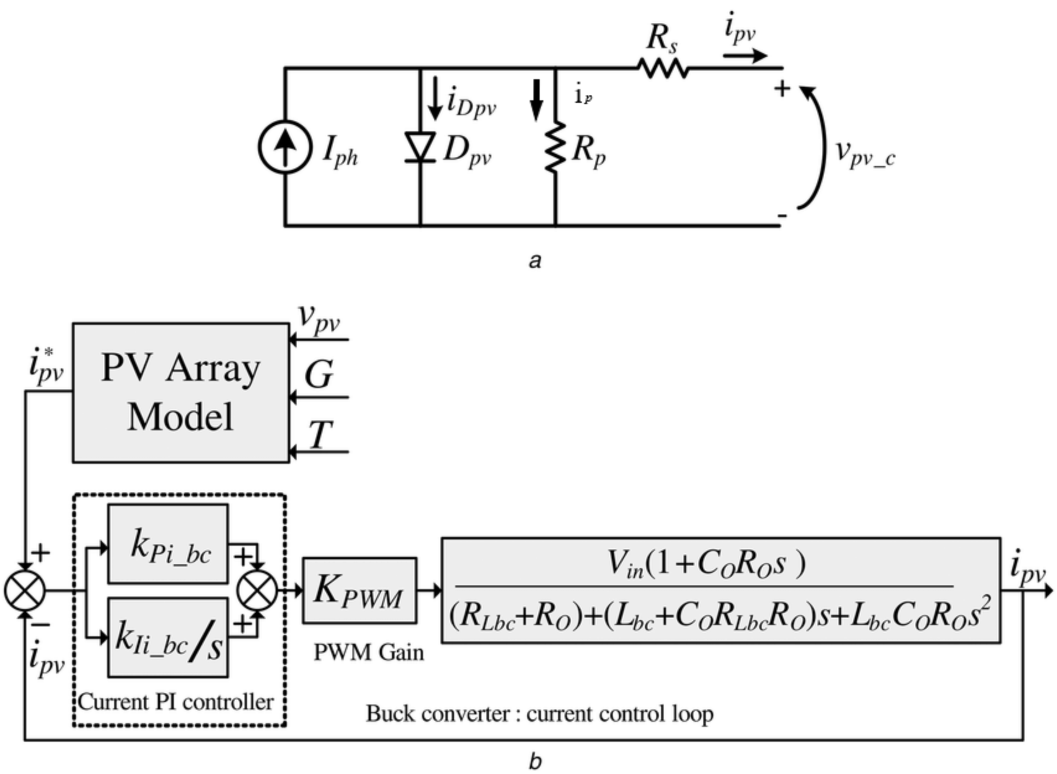

Figure 1: Diagram of PV emulator and voltage control

It can be seen from the above diagram that

$$I_{pv} = I_{ph} - i_{Dpv} - i_p \tag{1}$$

where the diode current i_{Dpv} is given by

$$i_{Dpv} = I_S(e^{\frac{V_D}{V_T}} - 1) \tag{2}$$

and the thermal voltage of the diode V_T is

$$V_T = \frac{K_B * T * A}{q} \tag{3}$$

where A is the ideality factor of the solar cell, q is the charge of an electron, K_B is the Boltzmann constant, T is the temperature in Kelvin and the voltage across the diode V_D is

$$V_D = V_{pv\_c} - I_{pv} * R_s \tag{4}$$

The leakage current I_p is given by

$$i_p = \frac{V_{pv\_c} + I_{pv} * R_s}{R_{sh}} \tag{5}$$

Combining eqs. (1) to (5) gives the output current as

$$I_{pv} = I_{ph} - I_S(e^{\frac{V_{pv\_c} - I_{pv} * R_s}{\frac{K_B * T * A}{q}}} - 1) - \frac{V_{pv\_c} + I_{pv} * R_s}{R_{sh}} \tag{6}$$

As R_{sh} does not affect the output voltage and is comparatively large compared to R_s, it can be ignored. Therefore, the power of the system is given by $P = I * V_{pv\_c}$ or

$$P = V_{pv\_c} * (I_{ph} - I_S(e^{\frac{V_{pv\_c} - I_{pv} * R_s}{\frac{K_B * T * A}{q}}} - 1)) \tag{7}$$

The equation parameters I_s, I_{ph} and R_s can either found experimentally or from technical literature.
The simulated PV emulator and the I-V and P-V curves from LTSpice are shown in figs. 2 and 3

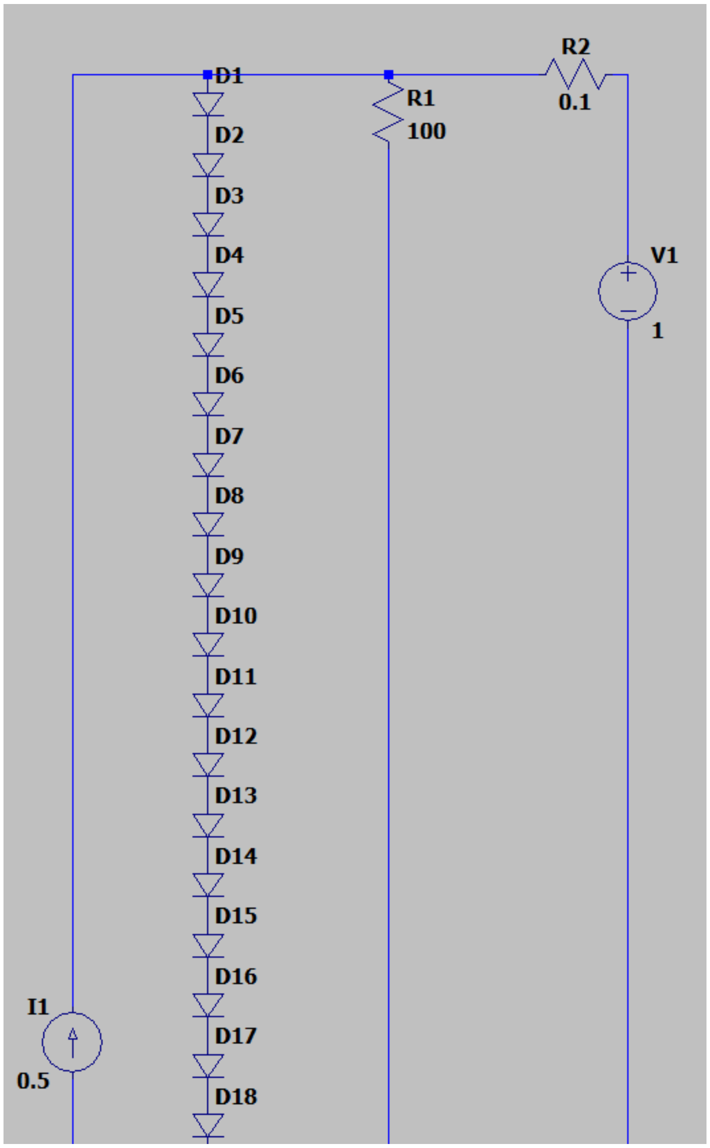

Figure 2: PV emulator LTSpice circuit

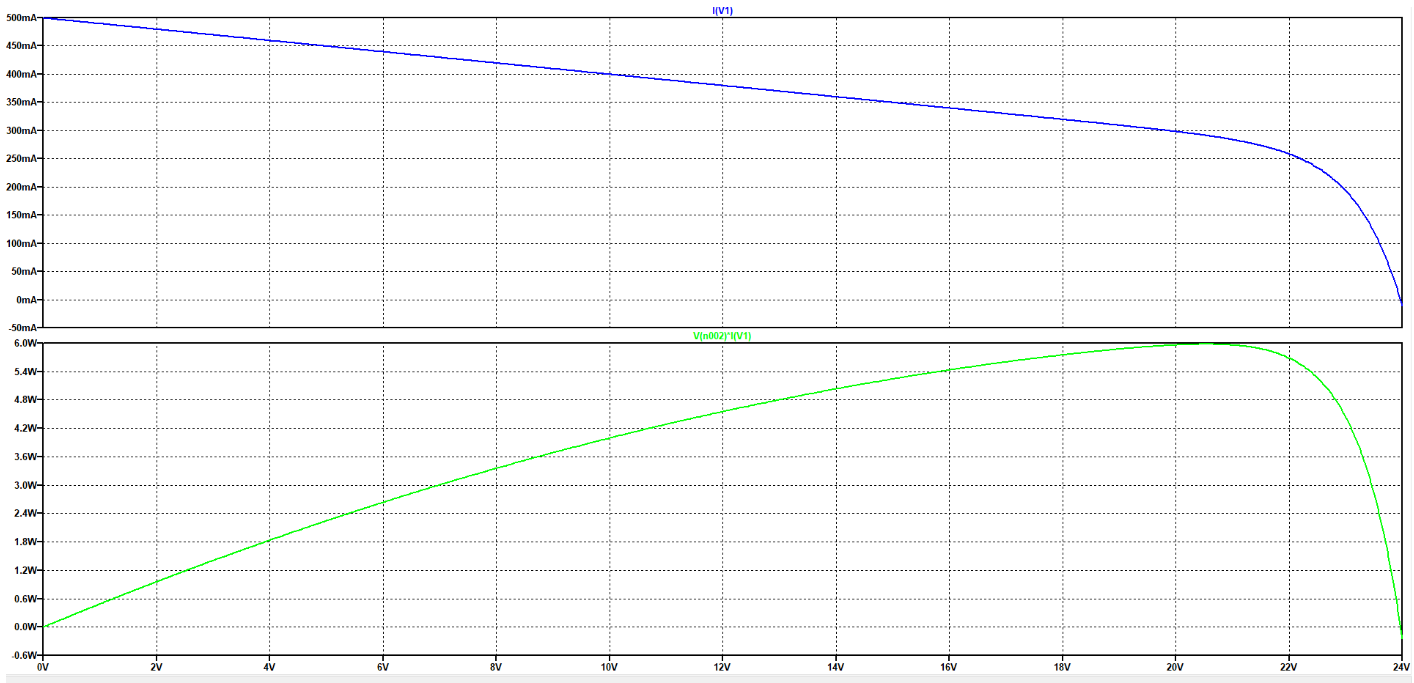

Figure 3: I-V and P-V curves of LTSpice PV emulator

2.1.2 Explain how the duty cycle should change to make the PV panel voltage constant and operate at the reference

If the PV panel voltage drops below the reference value, the duty cycle should be decreased in order to bring the voltage back to the reference value. Similarly, if the PV voltage rises above the reference value, the duty cycle should be increased to ensure the PV voltage drops to the reference voltage.

2.2 Lab Work

2.2.1 PV Emulator

In order to properly implement the constant voltage reference MPPT algorithm, the maximum power point of the PV emulator was found by connecting the PV emulator in parallel with the lab power supply and connecting a variable resistance box to the circuit. The resistance was changed between 0Ω and 270Ω and the voltage and current were measured. Open circuit conditions were also measured. The results are shown in table 1, and the I-V and P-V curves are shown in figs. 4 and 5

Resistance	Current	Voltage	Power
0	0.412	0.22	0.09064
1	0.411	0.7	0.2877
4.7	0.411	2.17	0.89187
10	0.411	4.34	1.78374
22	0.41	9	3.69
33	0.409	13.3	5.4397
47	0.396	18.5	7.326
68	0.313	20.9	6.5417
100	0.208	20.9	4.3472
220	0.095	20.9	1.9855
245	0.085	20.9	1.7765
270	0.077	20.9	1.6093
O/C	0	20.9	0

Table 1: Electrical characteristics of PV emulator

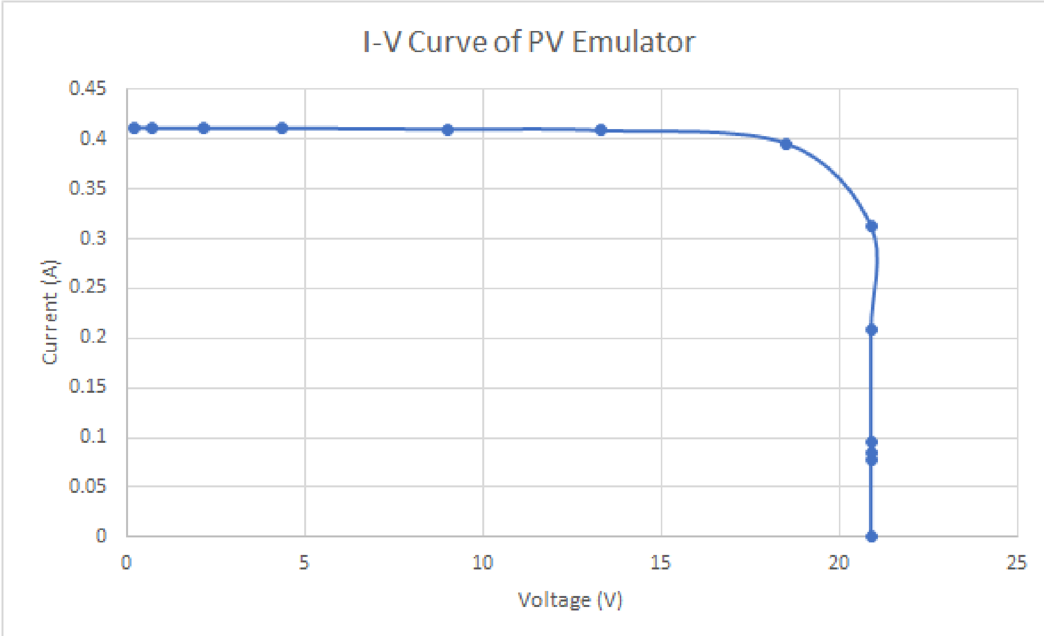

Figure 4: I-V curve of PV emulator

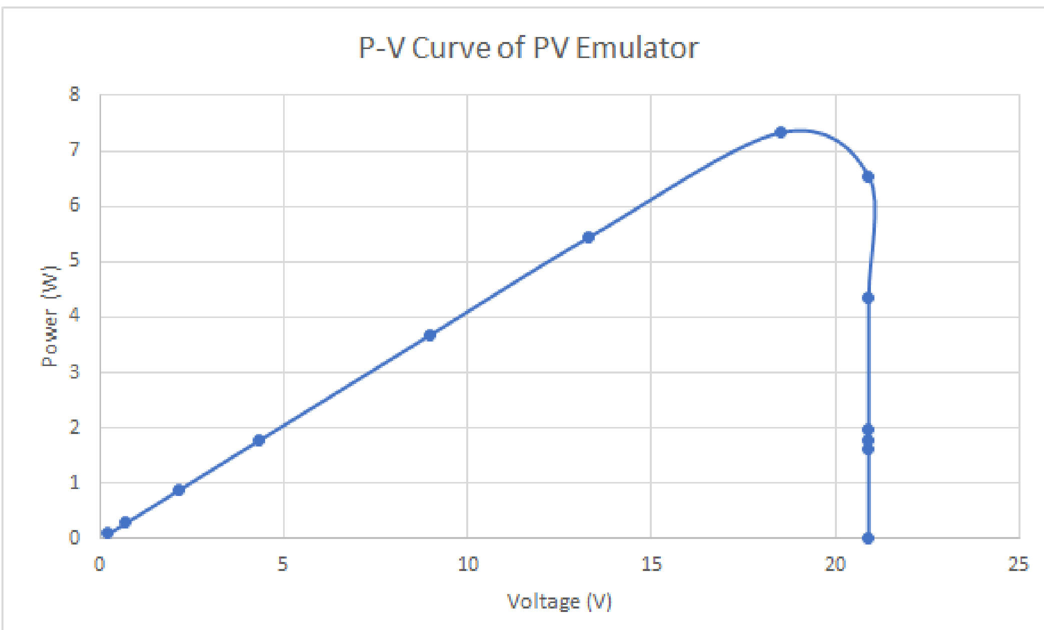

Figure 5: P-V curve of PV emulator

2.2.2 Testing of Constant Voltage Reference MPPT

The lab's buck converter was connected between the PV emulator and the variable resistance box, and set up using an Arduino as the PWM driver. The lab power supply was set to 21V and the resistance box set to 33Ω. It was verified to be operating correctly, as shown in fig. 6.

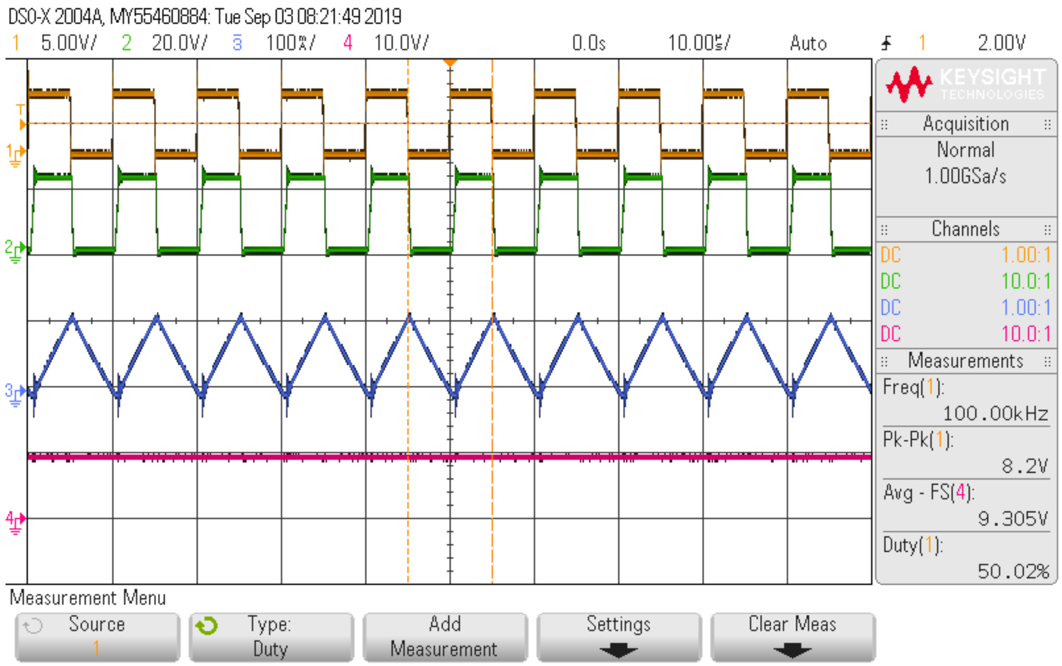

Figure 6: Correct operation of lab buck converter

The MPPT algorithm could then be tested by varying the input current and measuring the output power. Oscilloscope captures showing the duty cycle of the buck converter, displayed in orange, power output of the buck converter, displayed in pink, the input current, displayed in blue, and output voltage, displayed in red, are shown in figs. 7 to 13

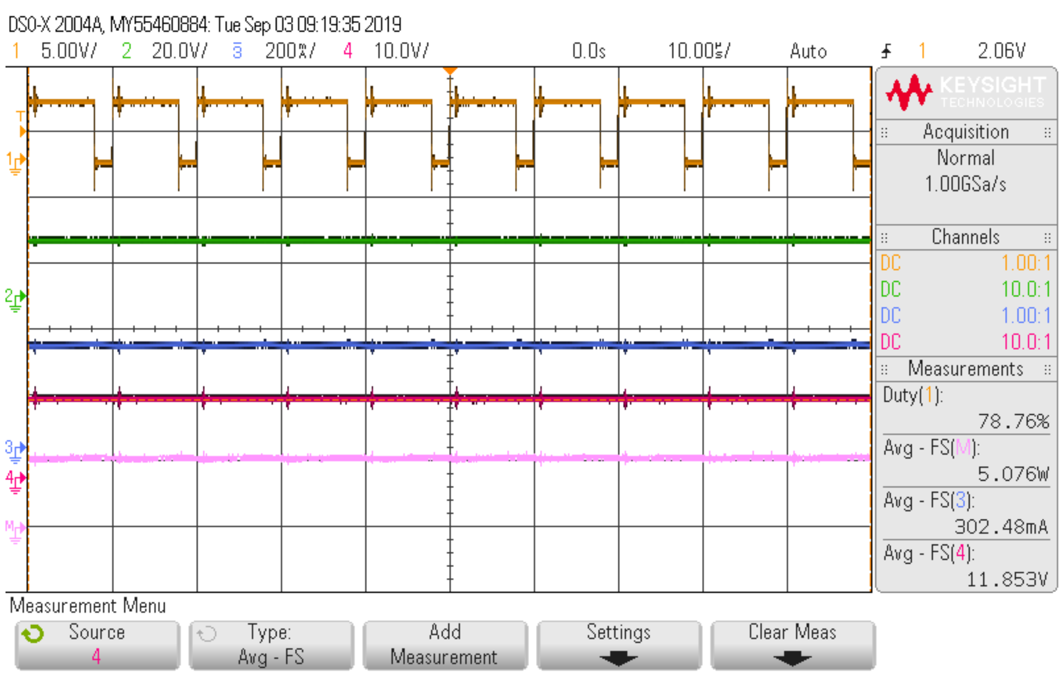

Figure 7: Constant Voltage Reference MPPT with 0.3A supply

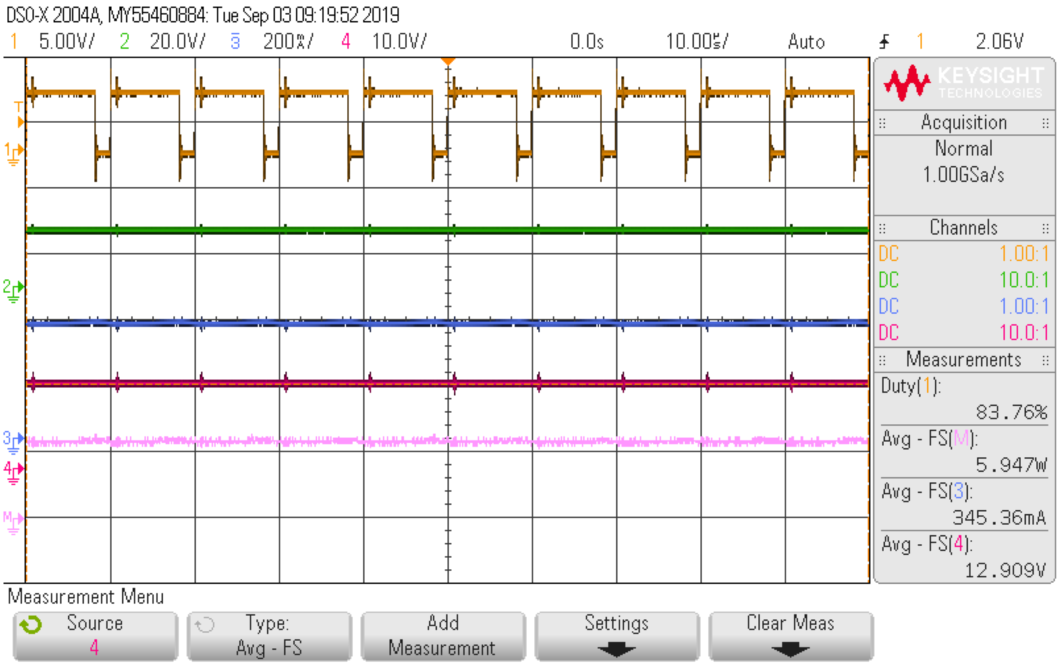

Figure 8: Constant Voltage Reference MPPT with 0.35A supply

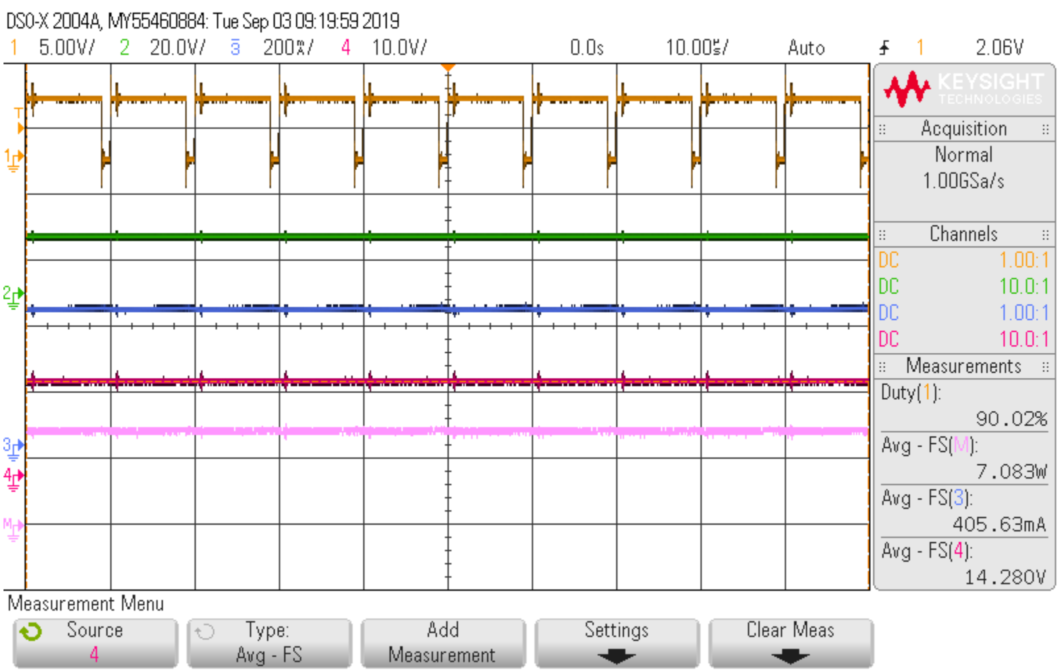

Figure 9: Constant Voltage Reference MPPT with 0.4A supply

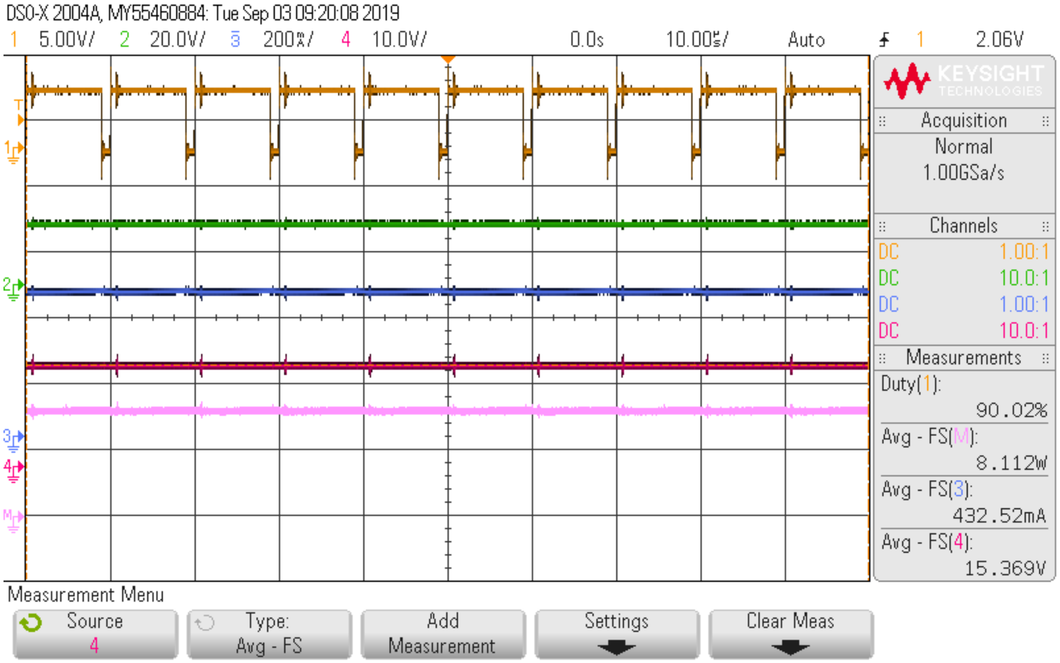

Figure 10: Constant Voltage Reference MPPT with 0.45A supply

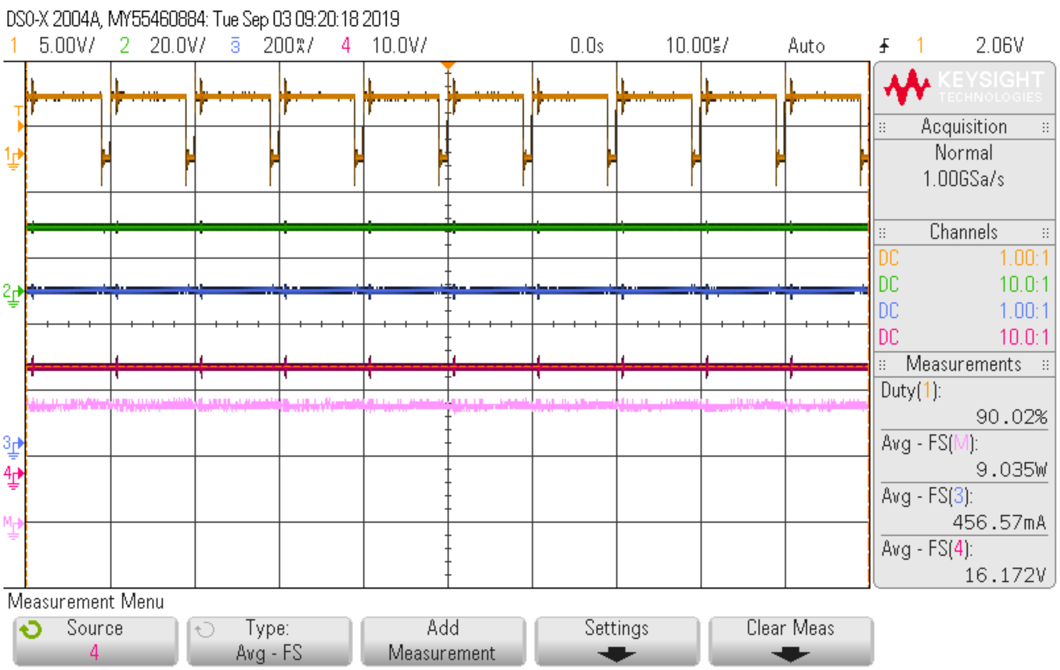

Figure 11: Constant Voltage Reference MPPT with 0.5A supply

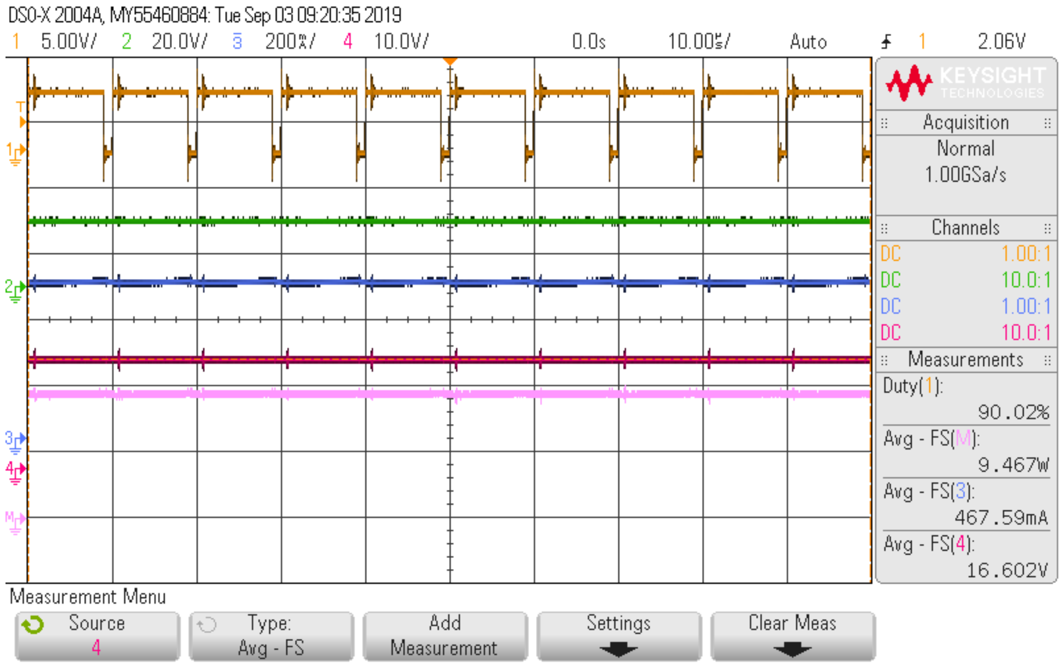

Figure 12: Constant Voltage Reference MPPT with 0.55A supply

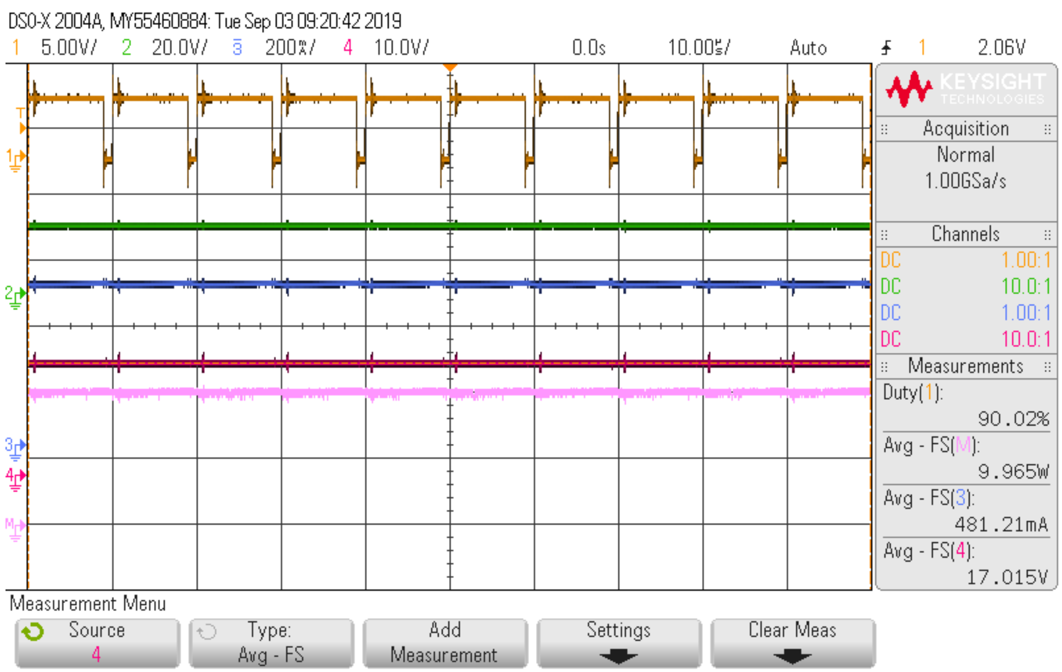

Figure 13: Constant Voltage Reference MPPT with 0.6A supply

2.3 Analysis

2.3.1 Explain, with aid of equations and diagrams, how the buck converter serves as a variable resistor to the PV panel

The electric power for the system is given by

$$V_o I_o = \eta V_i I_i$$

where V_o and I_o are the output voltage and current, η is the efficiency of the system, and V_i and I_i are the input voltage and current. Using Ohm's Law, I_o and I_i can be represented as

$$I_o = \frac{V_o}{Z_o}$$
$$I_i = \frac{V_i}{Z_i}$$

where Z_o and Z_i are the output and input impedances. Substituting these into the previous equation results in

$$\frac{V_o^2}{Z_o} = \frac{\eta V_i^2}{Z_i}$$

Assuming the buck converter is operating in continuous mode, $V_o = DV_i$, where D is the duty cycle, meaning that

$$\frac{V_o^2}{Z_o} = \frac{\eta V_i^2}{Z_i}$$
$$\implies \frac{(DV_i)^2}{Z_o} = \frac{\eta V_i^2}{Z_i}$$

This can be simplified to

$$\frac{D^2}{Z_o} = \frac{\eta}{Z_i}$$

and further to

$$D = \sqrt{\frac{\eta Z_o}{Z_i}}$$

This shows that the changing the duty cycle of the buck converter effectively changes the impedance ratio seen by the source, acting as a variable resistor.

2.3.2 Explain your part of the code to achieve constant voltage reference MPPT with the support of experimental results

```
convVolt = analogRead(pin_ADC) * 0.0049;
adjVoltage = convVolt / 0.18;
Serial.println(adjVoltage);
if (adjVoltage > vref)
        Duty = Duty + 1;
else if (adjVoltage < vref)
        Duty = Duty - 1;
if (Duty > 90)
        Duty = 90;
else if (Duty < 10)
        Duty = 10;
```

Listing 1: Code for Constant Voltage Reference MPPT

The method of operation for the Constant Voltage Reference MPPT can be seen in listing 1. The Arduino initially reads a voltage from the output of the buck converter, then converts and adjusts it so it is represented as a true voltage. This input voltage is then compared against the reference voltage `vref`. If the input voltage is greater than the reference, the duty cycle is increased to lower the input voltage. If the input voltage is less than the reference, the duty cycle is decreased to increase the input voltage. The code also limits the duty cycle such that it remains between 10% and 90% to prevent damage to the buck converter.

2.3.3 Compare the constant voltage reference MPPT performance against the identified true MPPs

Unfortunately, when gathering the I-V and P-V curves for the PV emulator in section 2.2.1, only one set of data with a current limit of 400mA was taken, meaning the performance of the MPPT algorithm cannot be shown across different insolation levels. However, in fig. 9, it can be seen that the output power is approximately 7W, close to the maximum power point identified in fig. 5. It is possible that the algorithm could have tracked closer to the MPP, however the duty cycle of the buck converter being limited to 90% may have prevented this.

3 Perturb and Observe (P&O) MPPT

3.1 Pre-Work

3.1.1 Explain why the power information at a particular point can be simplified to: Power information at point $X = V_{out} \times V_{PV}$

The power of the PV panel is given by $P_{PV} = I_{PV} \times V_{PV}$, where I_{PV} is the current and V_{PV} is the voltage produced by the PV panel. As the current produced by the PV panel can be found by multiplying the circuit output current, I_{out}, by the duty cycle of the buck converter, D, the power equation can be represented as $P_{PV} = DI_{out} \times V_{PV}$. Using Ohm's Law, the equation can be further represented as $P_{PV} = D\frac{V_{out}}{R_{out}} \times V_{PV}$. As the P&O algorithm does not need to take actual values into account, only their relative magnitudes, any constant values can be omitted and the power information can be represented as:

$$\text{Power information at point } X = V_{out} \times V_{PV}$$

3.1.2 Design a Perturb and Observe Algorithm in Your Own Words

The control system obtains the power information at a given point on the PV panel's P-V curve using the equation in section 3.1.1 and stores it. The control system then reduces the buck converter's duty cycle and measures the power information at the new operating point. If the power at the new operating point is greater than the old operating point, the control system will continue to decrease the duty cycle when measuring the change. However, if the power information at the new operating point is less than that at the old operating point, the control system will increase the duty cycle. Once the power begins to drop again, the control system will start to decrease the duty cycle. This process happens continuously, ensuring the power output will oscillate around the maximum power point of the system.

Overall, when the power drops when the duty cycle is increasing, the controller will switch to decreasing the duty cycle, and if the power drops while the duty cycle is decreasing, the controller will switch to increasing the duty cycle.

3.2 Lab Work

3.2.1 PV Emulator

In order to test the P&O MPPT algorithm, the MPPs of the PV emulator were found in a similar fashion to section 2.2.1. The PV emulator was tested with current limits of 400, 300 and 200 mA. The results are shown in tables 2 to 4 and the I-V and P-V curves are shown in figs. 14 and 15

Resistance	Current	Voltage	Power
0	0.39	0.064	0.02496
1	0.39	0.46	0.1794
4.7	0.39	1.9	0.741
10	0.39	4.14	1.6146
22	0.39	8.7	3.393
33	0.396	12.87	5.09652
47	0.38	18.76	7.1288
68	0.305	20.85	6.35925
100	0.204	20.86	4.25544
220	0.091	20.87	1.89917
245	0.082	20.87	1.71134
270	0.074	20.87	1.54438
O/C	0	20.86	0

Table 2: Electrical characteristics of PV emulator at 400mA current limit

Resistance	Current	Voltage	Power
0	0.308	0.05	0.0154
1	0.308	0.36	0.11088
4.7	0.308	1.47	0.45276
10	0.308	3.22	0.99176
22	0.307	6.76	2.07532
33	0.306	10	3.06
47	0.304	15.5	4.712
68	0.276	19.35	5.3406
100	0.203	20.87	4.23661
220	0.09	20.87	1.8783
245	0.082	20.87	1.71134
270	0.074	20.88	1.54512
O/C	0	20.88	0

Table 3: Electrical characteristics of PV emulator at 300mA current limit

Resistance	Current	Voltage	Power
0	0.201	0.123	0.024723
1	0.201	0.238	0.047838
4.7	0.201	0.96	0.19296
10	0.201	2.1	0.4221
22	0.2	4.4	0.88
33	0.2	6.5	1.3
47	0.2	9.86	1.972
68	0.198	13.6	2.6928
100	0.182	18.6	3.3852
220	0.09	20.9	1.881
245	0.082	20.9	1.7138
270	0.074	20.9	1.5466
O/C	0	20.9	0

Table 4: Electrical characteristics of PV emulator at 200mA current limit

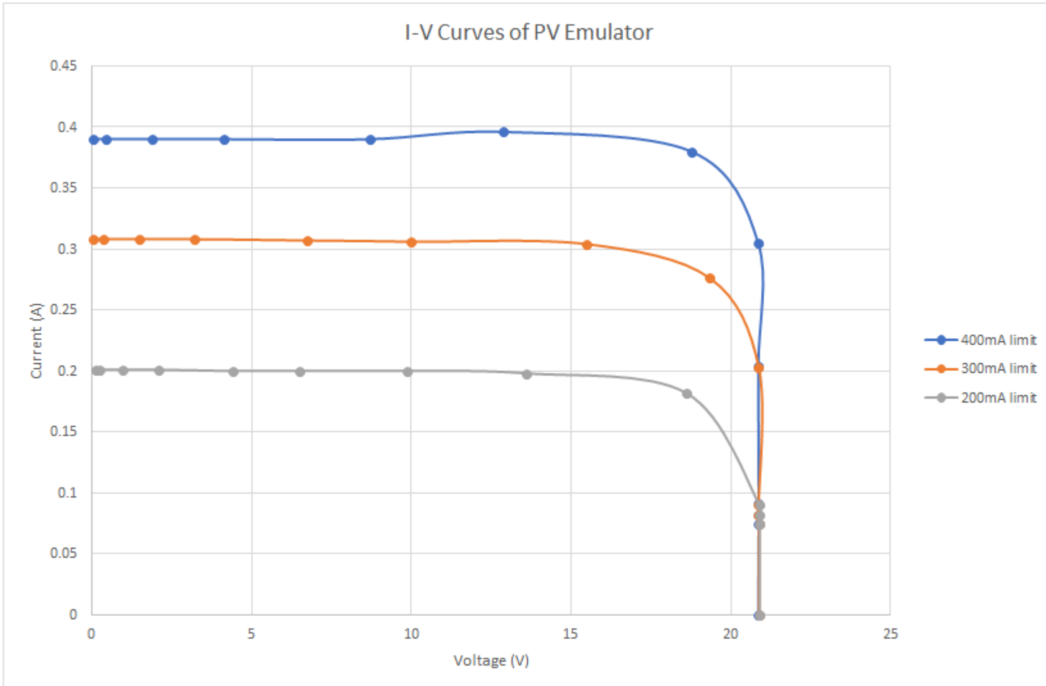

Figure 14: I-V curves of PV emulator

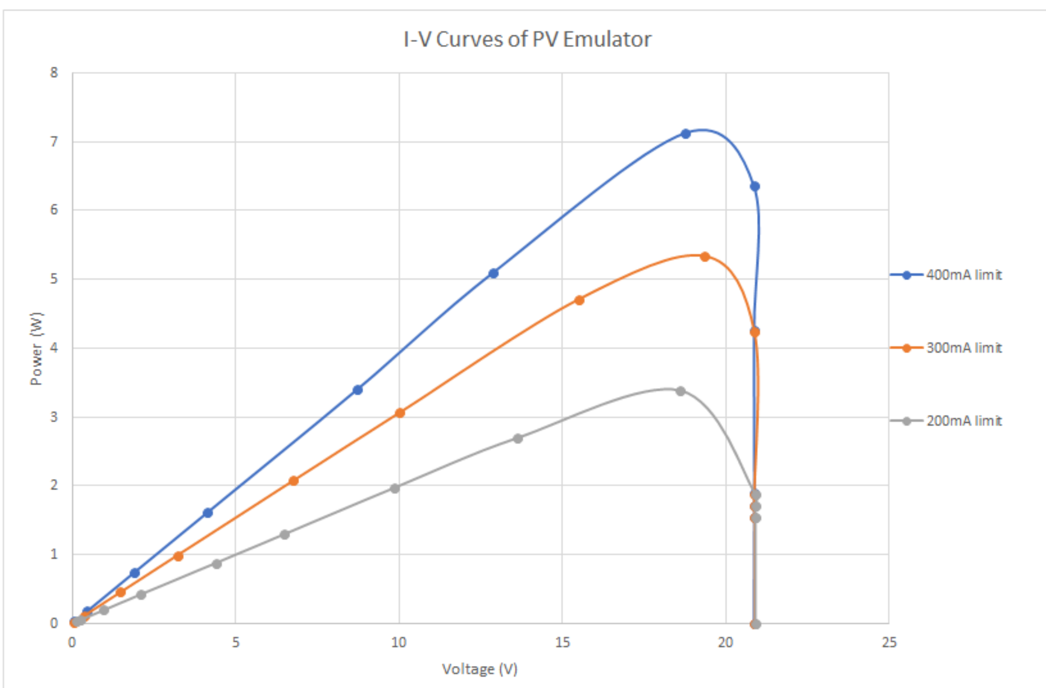

Figure 15: P-V curves of PV emulator

3.2.2 Testing of P&O MPPT Algorithm

The P&O MPPT algorithm was tested in a similar fashion to the Constant Voltage Reference MPPT algorithm in section 2.2.2. Once, the buck converter was verified to be working correctly, shown in fig. 16, the new algorithm was tested.

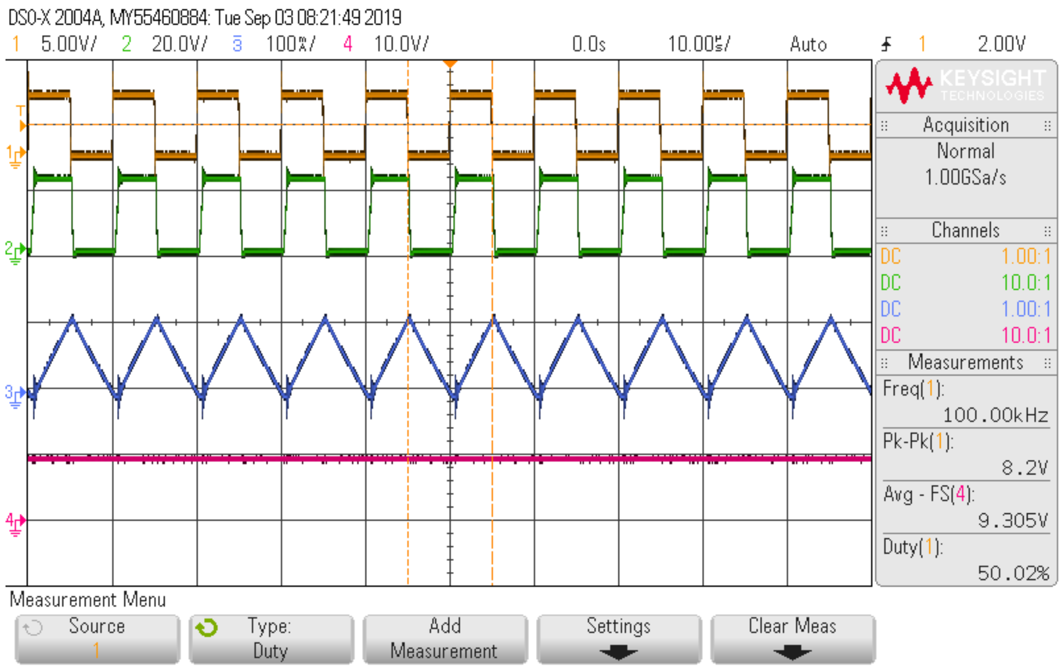

Figure 16: Correct operation of lab buck converter

The results are shown in figs. 17 to 21

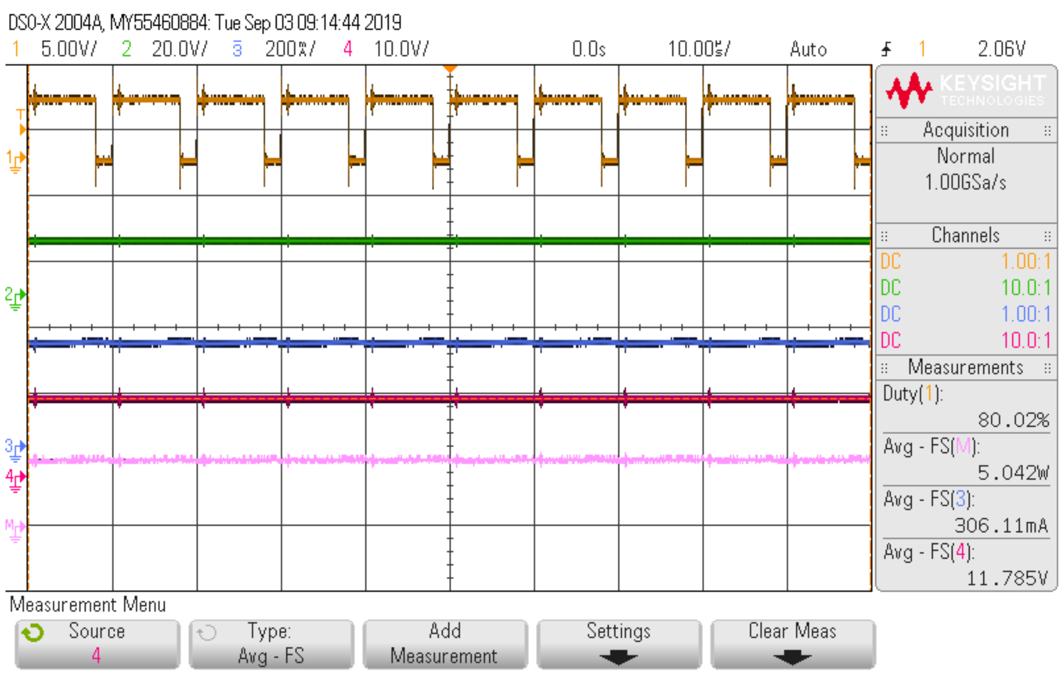

Figure 17: Perturb and Observe MPPT with 0.3A supply

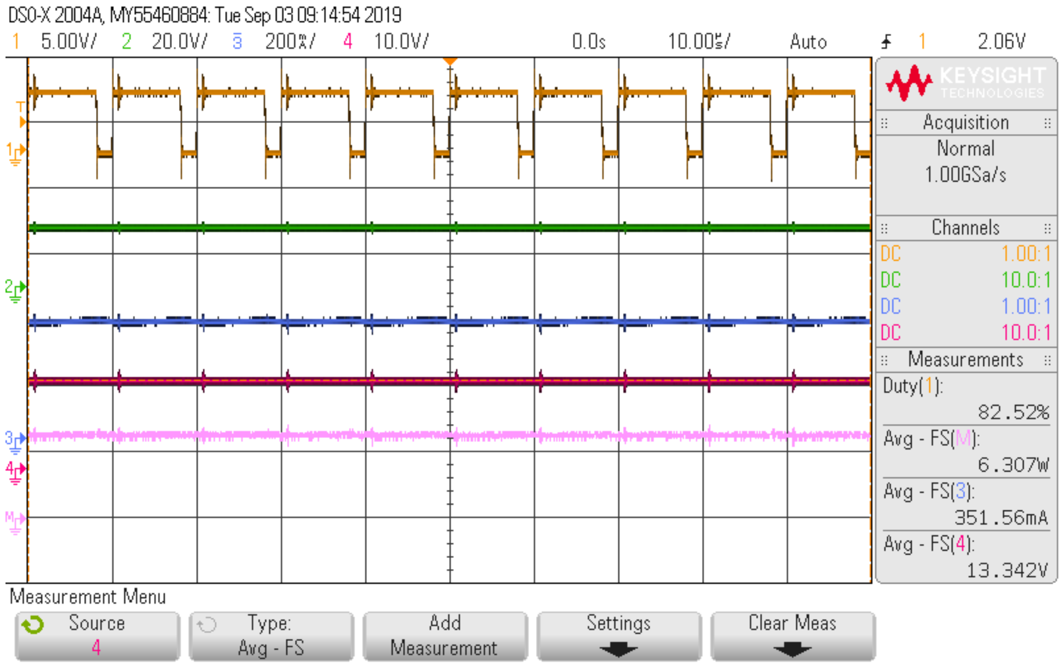

Figure 18: Perturb and Observe MPPT with 0.35A supply

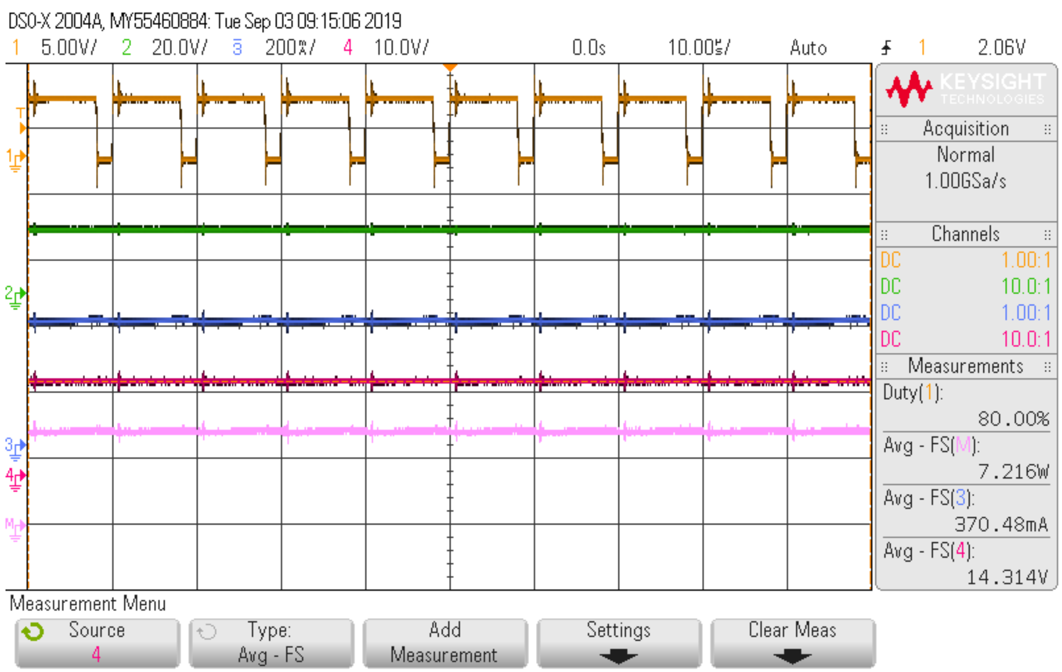

Figure 19: Perturb and Observe MPPT with 0.4A supply

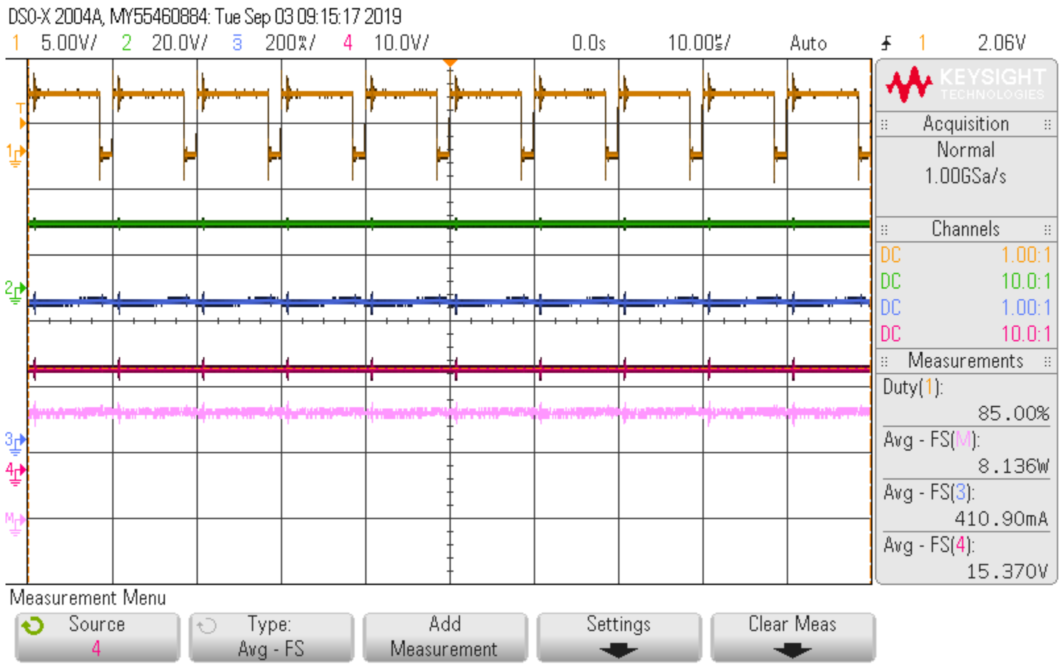

Figure 20: Perturb and Observe MPPT with 0.45A supply

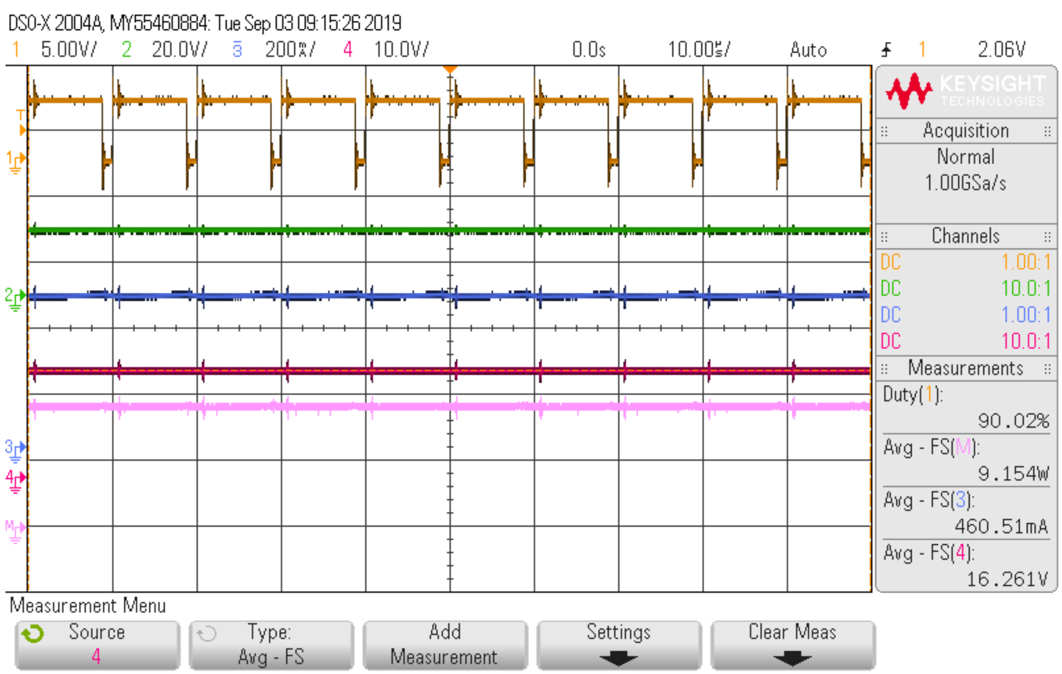

Figure 21: Perturb and Observe MPPT with 0.5A supply

3.3 Analysis

3.3.1 Explain your part of the code to achieve P&O MPPT with the support of experimental results

```
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;

newP = Vout_adjV * Vpv_adjV;

if (oldP > newP)
        Duty = Duty - 1;
else
        Duty = Duty + 1;

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

setDutyCycle(Duty);
oldP = newP;
```

Listing 2: Code for Perturb and Observe MPPT

The algorithm initially reads the output and the PV emulator voltages, then adjusts and converts them to values representing the real voltages. It then multiplies these values to get the power information at that point, as explained in section 3.1.1, completing the Observe phase of the algorithm. This power information is then compared with the old power information; if the new power is greater than the old power, the duty cycle is reduced; whereas if the new power is less than the old power, the duty cycle is increased. This completes the Perturb phase of the algorithm. The new power is then considered the old power and stored for use in the next cycle of the algorithm.

3.3.2 Compare the P&O MPPT performance against the identified true MPPs

It can be seen from fig. 15 that the MPPs exist at approximately 7.2W and 5.4W for 400mA and 300mA current limits respectively. The experimental results shown in figs. 17 and 19 show that the power points the algorithm tracked to are approximately equal to the MPPs at the given current limits.

3.3.3 Is P&O a perfect maximum power point algorithm? Why or why not? Explain your argument with the experimental results if possible

Perturb and Observe could be considered a near-perfect MPPT algorithm due to the ease of which it tracks to the maximum power point. Its performance, however, depends greatly on the size of the disturbance used in the Perturb phase of the algorithm. If the disturbance is too large, the algorithm will never be able to reach the MPP of the system due to large, constant oscillation. Similarly, if the disturbance is too low, the algorithm may not be fast enough to cope with changing insolation conditions. Both of these scenarios result in reduced efficiency levels and, as such, the size of the disturbance used must be closely considered.

3.3.4 Suggest a way to improve the tracking accuracy of P&O

Using an adaptive disturbance would improve the tracking accuracy of the P&O algorithm. The size of the disturbance used in the Perturb phase of the algorithm would change depending on whether the algorithm was close to the MPP or not. If the current power point is far from the maximum, the disturbance size would be larger than if the maximum had already been reached. This would allow the MPP to be found faster while still maintaining a high level of efficiency when the MPP has been found.