Design Manual 113B

Smit Malde

This design manual will detail the process to design, test and run an MPPT algorithm on a DCDC Buck converter.

Table of Contents

Introduction

So why is MPPT necessary?

MPPT stands for Maximum Power Point Tracking and as the name suggests, it tracks the maximum output power of the converter and aims to keep it running at that operating point.

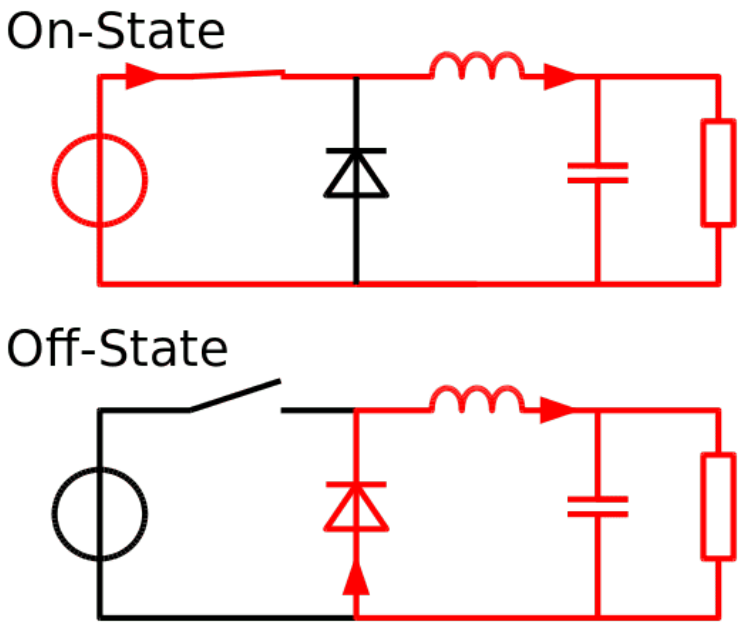
This is necessary because when using a PV panel or a Solar panel, there is a lot of dependability on external factors like weather conditions and hence at any given point you want to try and make the most out of your power source by drawing the maximum power. Hence we design the MPPT algorithm to track the maximum power output and adapt dynamically to any external changes to provide the maximum power.

The Buck Converter

I designed a buck converter for a PV panel input and as such runs on the MPPT algorithm. This topology is great for the MPPT algorithm because the only variable you have to adjust for a basic MPPT algorithm is the duty cycle and implement a simple feedback loop to achieve the maximum power output.

Operation

There are 2 operating states for the buck converter. In each state, only one of the switches is turned on to prevent shorting the source. You can see the 2 states below along with the flow of current through the switches in each state. The image below shows a diode instead of a switch, this works the same way except that you have lower losses when using a switch.



In real life these switches are not ideal and hence have some delay in turning on and off, this delay is denoted as rise time and fall time. So, when you switch between states, you can imagine that there will be a short amount of time, where the fall time of one switch and the rise time of the other switch are overlapping, for this brief period you would be shorting the source and hence we introduce deadtime into the circuit. Deadtime is some amount of time that you wait after turning off one switch to turn on the other to ensure that there is no time of overlap between both switches.

Frequency Selection and Component Sizing

How to select the switching frequency?

Getting the right switching frequency can greatly affect efficiency as well as the current ripple in your converter. Increasing the frequency will reduce your ripple but at the cost of efficiency as your switching loss will increase because you are increasing how many times per unit time you are switching. This is seen using the following equations:

- $\Delta i_{L,pp} = \frac{V_L.D}{\mathrm{L}.f_{sw}}$
- $\Delta P_{loss,ov} = \left[\left(\frac{1}{2}t_f I_L V_{in}\right) + \left(\frac{1}{2}t_r I_L V_{in}\right)\right] f_{sw}$
- $\Delta P_{coss} = (C_{oss}V_{sw}^2)f_{sw}$

For my design since I was meeting my ripple spec by a decent amount at 100KHz and hence decided to reduce my switching frequency by 10% at 90KHz to reduce my switching losses and increase over convertor efficiency.

Choosing Passives

My passive components were sized based on the ripple spec requirements at the highest operating point.

The inductor specification was <10% ripple, hence using the following equation, I calculated the minimum inductance:

- $\mathrm{L_{min}} = \frac{\mathrm{V_{in}.D(1-D)}}{\mathrm{\Delta i_{L,pp}.f_{sw}}}$ = 72µH

For sizing the input capacitor I used the following equation to get a minimum capacitance for the input:

$\mathrm{C_{in,min}} = \frac{\mathrm{(P/Vin^2).(1-D)}}{\mathrm{\Delta V_{C,pp}.f_{sw}}}$ = 20.58µF

The value for the output capacitor was calculated by the following equation:

$\mathrm{C_{out,min}} = \frac{\mathrm{\Delta I_{L,pp}}}{\mathrm{8.D.f_{sw}}}$ = 1.736uF

Switch Stresses

The switches are selected based on the max current and voltage they will experience. These values can be calculated using:

Voltage stress = $V_{in} + \frac{\Delta V_{in,pk-pk}}{2}$

And the Current stresses would be RMS currents through the switches calculated using the RMS equations learned in 113A.

Below is a table that summarizes the switch stresses:

Operating Point	Voltage Stress [V]		RMS Current [A]	
	Switch 1	Switch 2	Switch 1	Switch 2
Nominal	21	21	6.458	5.273
24V 100W	25.2	25.2	6.458	5.273
16V 100W	16.8	16.8	6.458	5.273
24V 50W	25.2	25.2	3.229	2.6365
16V 50W	16.8	16.8	3.229	2.6365

Switches

Switch Losses

One of the losses is due to the resistance Rdson, this is called conduction loss. This loss occurs in both the switches and is calculated by:

- $P_{diss,cond} = i_{sw,rms}^2 . R_{DS,ON}$

Another loss is the overlap loss. Since the switches are not ideal, they take time to turn On and Off, this loss is caused due to the voltage and current overlap during switch transitions. During the transition, when the high-side switch turns off and the low-side switch turns on, there is a delay for the current through the high-side switch to drop to 0 (as well as the voltage across it to increase to V_{in}) and the current through the low-side switch to rise to the inductor current, and voltage across it drop to 0V. This loss is calculated by:

- $\Delta P_{loss,ov} = \left[\left(\frac{1}{2} t_f I_L V_{in}\right) + \left(\frac{1}{2} t_r I_L V_{in}\right)\right] f_{sw}$

Whereby, t_r is the rise time of the switch and t_f is the time for the current to fall to 0A of fall time.

There is also a power loss due to the output capacitance (Coss loss). In the switching operation, the power source charges the output capacitor to store the energy during the

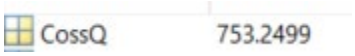
turn-on phase. When the MOSFET is turned off, the stored energy in *the* output capacitor discharges via the body diode and causes the turn-on loss. It is calculated using:

- $\Delta P_{coss} = (C_{oss} V_{sw}^2) f_{sw}$

C_{oss} is the output capacitance which is the sum of the capacitance between the C_{ds} and C_{gd}.

This is the Coss loss of the switch from the matlab script:

47A - 2.384 W



Overlap loss and Coss loss is only on to the high-side switch.

The following table summarizes the switch losses at different operating points:

	Nominal	24V 100W	16V 100W	24V 50W	16V 50W
Switch Loss [W]	1.1408	1.1644	1.1685	0.4896	0.4149

Thermals

According to my calculations only the high side switch requires a heatsink.

The switch has an operating range of -55^{0}C to 175^{0}C. The temperature increases due to the power dissipated through the RDS,ON. However, the RDS,ON also increases as the temperature increases, and this is seen from the graph in the datasheet of the switch. Therefore, I decided to put a heatsink on my high-side switch to keep the temperature of the switch lower hence reducing conduction loss.

Magnetics

In the Buck topology that I made; the only magnetic component is the Inductor which deals with primarily DC. The only AC aspect of the system is the current ripple on the inductor which in my spec is less than 10% in the worst case (< 830mA at 100W or < 415mA at 50W) and hence it is a very small component of the inductors current. Another factor in choosing magnetic components is the size as these tend to be the

How to select and design the Inductor

For our application where we are primarily dealing with DC, we need to look for the Bmax from the core materials datasheet and ensure that this value is less than or equal to Bsat for the switching frequency at which you want to run your converter at.

Some of the losses that come about from magnetic components like inductors are the Eddie currents and other core losses primarily seen when AC is running through the component. In terms

of DC losses for an inductor, you have winding losses from the resistance of the copper wire that is wound around the inductor. As stated above the AC aspects of our circuit is very minimal and hence we can ignore the AC losses from the core.

To calculate to the winding loss seen for the buck convertor, you want to calculate the resistance of the winding and then use $P = I^2R$ to find the power loss. To calculate resistance, you can use the following formula: Resistance = Resistivity (of copper probably) * Length / Cross sectional area or:

$$R = \frac{\rho L}{A}$$

As an example I will walk through the process of designing my inductor:

I chose the Ferrite core for 2 reasons;

1. It was the only core material available in lab
2. It also has the smallest specific power loss per volume and is great for a wide range of frequencies

I used the Kg method to reach an optimum inductor design. I started by calculating the maximum current and the RMS current through the inductor:

1. Max Current = Avg Current (100 / 12 = 8.33 A) * 1.05 = 8.75 A
2. I_RMS = 8.34 A (using the RMS formulas learned in 113A)

Now that we have these, we can set a max power loss of 1 W for our winding loss to get a max resistance for the winding.

- $P = I^2R \Rightarrow R = 1/8.34^2 = 14.4\text{m}\Omega$

Now we want to find the Kg constant that is defined as: $Kg = \frac{L^2\, I\text{max}^2\, \rho}{\text{Bmax}^2\ \text{R Ku}}$

- $Kg = 7.29 * 10^{-12}\ \text{m}^5$

Using the Kg factor we can choose a core size from a lookup table and hence choose and RM12 Core that has a Kg factor of $1.8 * 10^{-11}\ \text{m}^5$.

After this we can calculate the Minimum Number of turns required for the using the following equation:

- $Nmin = \frac{LI_{Max}}{B_{Max}A_C}$ which is roughly comes to about 14 turns

Now we need to calculate the gap required to achieve the desired inductance of 72μH

- $g = \frac{\mu_0 A_{C,e} N^2}{L} = 480\mu\text{m}$

Now we need to calculate the wire size to ensure that the number of turns we need are about to fit in the inductor.

So A_W (max wire cross sectional area) = $\frac{W_A(Winding\ Area)K_u(Packing\ \text{Factor of } 0.5)}{N} = 2.61\text{mm}^2$

- $2\sqrt{\frac{2.61\text{mm}^2}{\pi}}$ = 1.82 mm , therefore we will use 14Awg wire which has a diameter of 1.628 mm.

Using this spec we expect our resistance to be 6.38 mΩ and a power loss of 0.44 W at the 100W operating points and about 0.11 W at the 50 W operating points.

I am ignoring core losses for this convertor because the AC component is very small but the core losses can be calculated with the peak flux density and comparing this to the graph of Pv against B on the datasheet.

According to the datasheet the temperature rise for this inductor is 23°C/W and hence for 0.44W we see a temperature increase of 10.12°C above the ambient temperature.

Capacitors

There are 2 main types of capacitors used for the power stage, MLCC and Electrolytic Capacitors.

- Electrolytic Caps have very high capacitance but also have high ESR for the same capacitance as compared to MLCCs
- MLCCs are much smaller in both volume and capacitance

Hence a balance of both Electrolytics and MLCCs are recommended to ensure that all the input and output current is supported as well as to minimize voltage ripples.

ESR is a big factor to consider in selecting the capacitors are higher ESR would mean higher losses from the current going in and out of the capacitors.

Based on the ripple specifications we need 20.56uF capacitance on the input and 1.736uF on the output.

Below is a table for my input and out Caps along with some of their properties:

Input Caps:

Part number	Type	Capacitance [µF]	ESR [mΩ]	Current ripple rating [mA]
50ZLJ470M10X20	Electrolytic	470	38	1430
50ZLJ470M10X20	Electrolytic	470	38	1430
C1206X7R500-106KNE-CT	MLCC	10	2.7	
C1206X7R500-106KNE-CT	MLCC	10	2.7	
C1206X7R500-106KNE-CT	MLCC	10	2.7	
C1206X7R500-106KNE-CT	MLCC	10	2.7	

I selected these caps to ensure that I had enough to handle the inrush currents and max current through operation.

Output Caps:

Part number	Capacitance [µF]	ESR [mΩ]
C1206X7R500-106KNE-CT	10	2.7
C1206X7R500-106KNE-CT	10	2.7
C1206X7R500-106KNE-CT	4.7	4.9

The total power loss for my output capacitors was about 1.5mW.

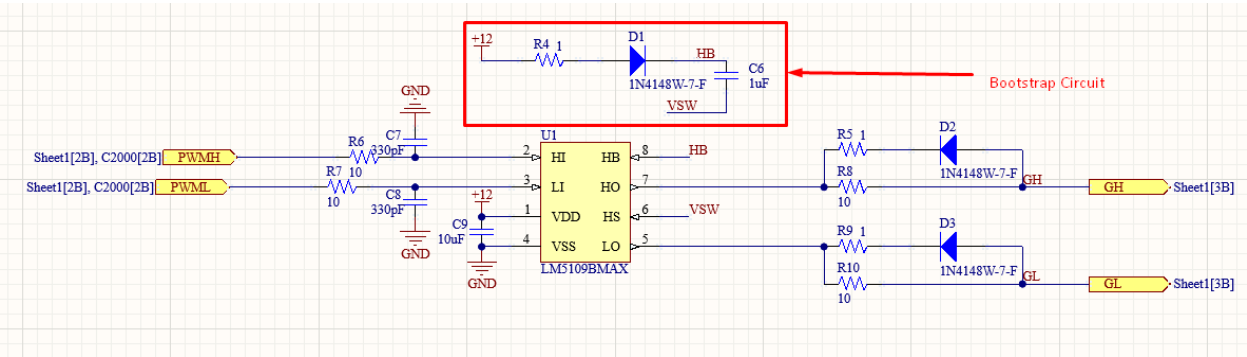
Gate Drive Circuit

To drive both the high side and the low side switches in my converter we need to implement a bootstrap circuit that allows our gate driver to sufficiently power the gate of the high side switch such that the Vgs > Vth.

If the Vgs for the switches is too high, we may increase the gate drive losses and if the Vgs is too low, the Rdson increases and hence increasing the conduction losses of the switch. Hence we need to find a god balance of Vgs in order to optimize for efficiency.

Hence I powered my gate drive circuit with a 14V Supply. And hence the gate driver drove the switches with Vgs at 14V and using a complimentary PWM for each switch from the microcontroller.

Below is the Schematic for both switches and the gate driver including all surrounding circuitry



Sensing Circuit

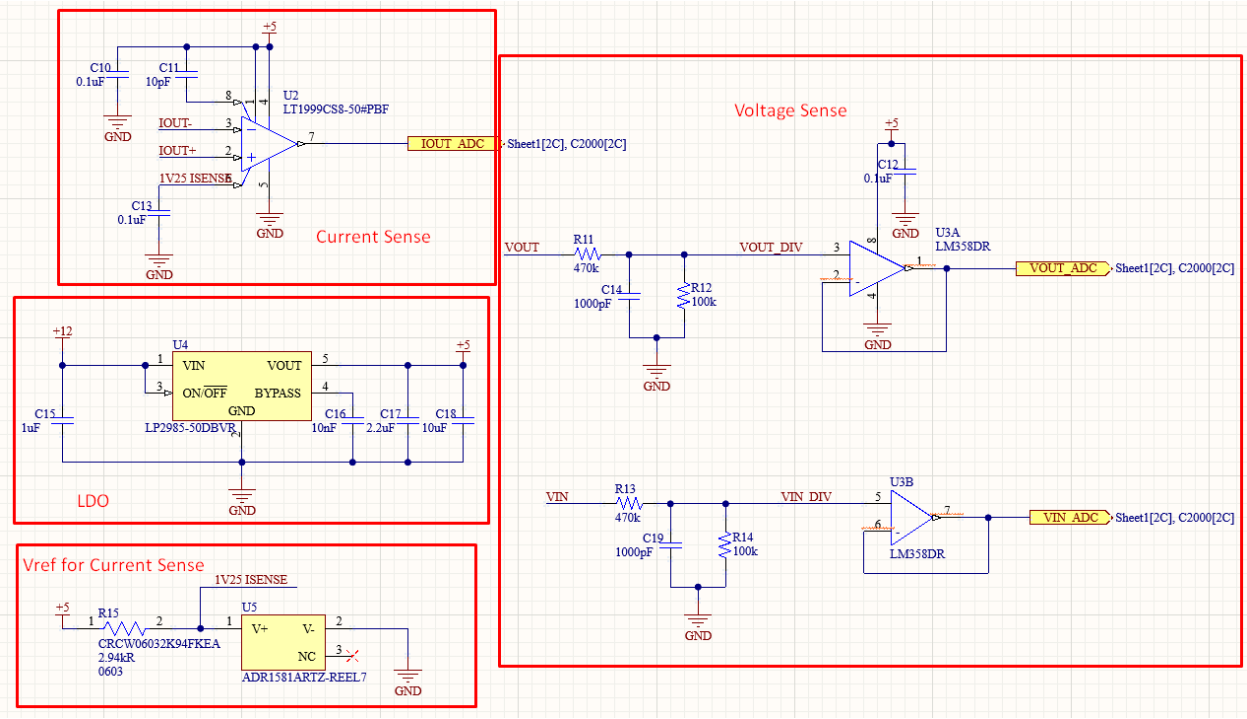
In order to design for an MPPT algorithm we realistically and minimally only need to sense the output voltage and output current. However I also included the input voltage sensing which can

help to optimize the MPPT algorithm but mostly because the sensing circuitry IC that I used had 2 Op-amps for voltage sensing and does not affect my efficiency significantly.

The MCU ADCs can take in voltages up to 3V hence the input and output voltages have to be stepped down to and mapped to a range of 3V. I used simple voltage dividers with high resistance and minimal ratio to minimize power loss and maximize resolution through the voltage dividers. My input divider was made using a 47Kohm and a 470Kohm resistor which divides my voltage by 11 hence a max input voltage of 24 corresponds to 2.18V on the ADC. For my Output voltage divider, I used 100Kohms and 470kohms to get a ratio of 1/5.7. The output voltage from the dividers is fed into unity gain opamps to supply current to the ADC hence ensuring no current is drawn from the dividers to maintain there ratios.

For current sensing a shunt resistor is used on the Vout node. There is a voltage drop across the shunt when current flows through it. Hence a differential opamp IC is used to measure the voltage drop across the shunt. I used a shunt of 2.5mOhms to reduce the power loss through the shunt while still maximizing the resolution. The Differential opamp is connected to the MCU using filters to read the voltage drop which is then converted to a current reading in firmware by using Ohms law and dividing that voltage by the shunt's resistance.
The voltage and current sense Circuitry was powered by an LDO that steps down 12V-5V.

Below are the voltage and current sensing circuitry:



Control

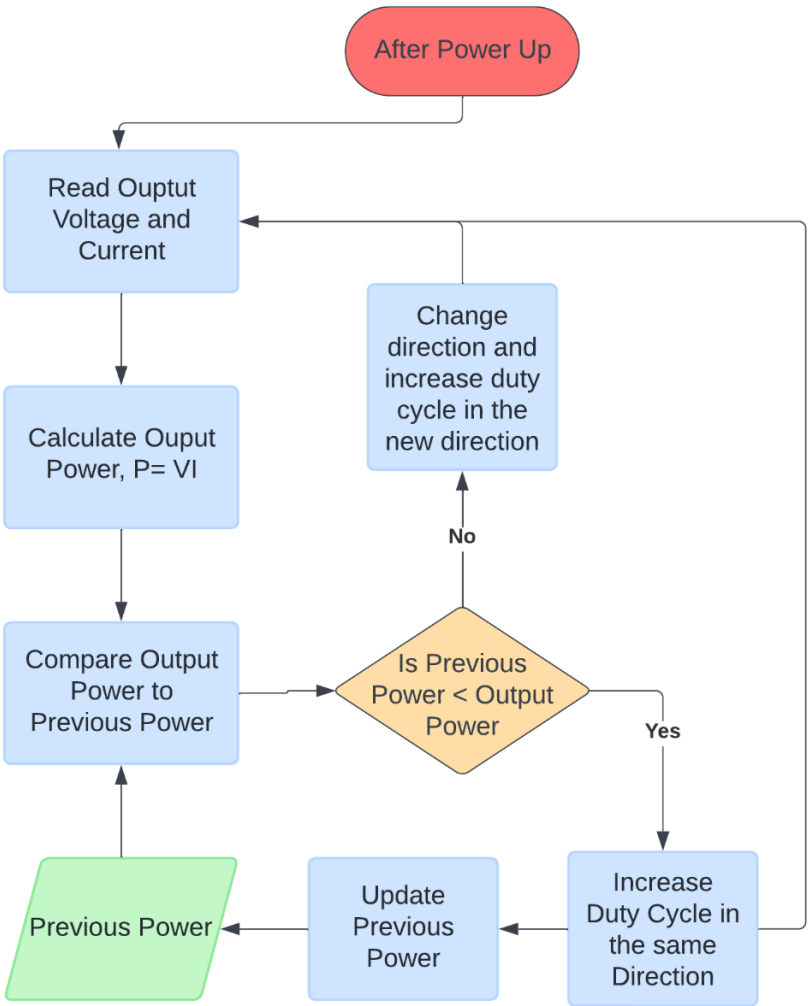
The MCU has certain pins that can send PWM signals, in my case I used pins 39 and 40 on the C2000 Launchpad board for the gate low and high signals respectively. The MCU has a standard

time known as a click which is the operating speed of the MCU itself. To set a frequency of 100KHz, around 500 clicks are set as the period for one PWM pulse. The signals are set as complementary to each other.

Digital PWM and Analog PWM are both generated by comparing a modulating waveform with a carrier signal. Digital PWM uses a discrete time-domain carrier signal, which can only achieve a limited number of steps within a certain time frame, controlled by a counter that increments with each cycle of the EPWMCLK until TBPRD is reached. Conversely, analog PWM employs a continuous carrier signal. On the C2000, you must set up both the frequency/period register (TBPRD) and the duty ratio register. The period register sets the TBPRD value, marking the peak before the TBCTR resets, defining the time period. The duty register adjusts the modulating waveform to set the PWM's duty ratio, with TBPRD configured according to the required switching frequency.

MPPT Algorithm

The below flowchart shows a simple MPPT algorithm implementation:



The MPPT Algorithm can be further optimized by starting the duty cycle at a predetermined calculated duty cycle to give maximum power output. We can also implement a feature where the duty_step is increased every few cycles when we don't meet the expected power output to ensure that it does not get stuck on a local maximum. Another strategy is to average over multiple cycles to get a more accurate reading, this would slow down the rate at which the algorithm can adapt slightly but would result in better performance.

Below is an implementation of my MPPT code with relevant variable definitions highlighted where I have implemented starting the algorithm at 0.7 duty cycle when the output power is below 40 W to ensure that the startup is not slow and also to ensure that it does not get stuck at a local maximum. I have also set a deadtime of 2 clicks which is about 20nS so that my switches are not on at the same time.

```
float prev_pout = 0;
float duty_direction = 1;

volatile Uint16 duty_cmp = EPWM_CMP_INIT;
float duty_step = .005f; //Duty cycle step size
float duty = 0.5f; //Start MPPT Search at the larg
float duty max = 0.75f; //Maximum duty cycle to se
uint32_t wait_time = 20e3; //amount of time to wai

float vin_ratio = 11.0f;
float vout ratio = 57.0f / 10.0f;
            else if(Perturb_Observe){
                //TODO: Implement Perturb and Observe Algorithm
                //The perturb and observe algorithm first _perturbs_ the current operating point (via a change in duty cy
                //First measure the current output power and compare it to the previously measured output power
                //If the current power is higher than the previous power, and the current duty cycle is larger than the p
                //The algorithm should handle both cases where the maximum power requires an increase or decrease in duty
                //Your controller should be able to continually find the maximum power point of the PV panel (even if the
                adc_vout = AdcaResultRegs.ADCRESULT0;
                vout = (adc_vout / (4095.0f / 3.0f)) * vout_gain;
                adc_iout = AdccResultRegs.ADCRESULT0;
                current_out = ((adc_iout / (4095.0f / 3.0f)) - 1.25f) / (r_shunt_out * iout_gain);
                pout = current_out * vout;
                if (pout < 40) {
                    duty = 0.7;
                } else {
                    duty_step = .005f;
                    if (pout > prev_pout) {
                       if (duty_direction == 1) {
                           duty += duty_step;
                       } else {
                           duty -= duty_step;
                       }
                    } else {
                       duty_direction *= -1;
                       if (duty_direction == 1) {
                          duty += duty_step;
                       } else {
                          duty -= duty_step;
                       }
                    }
                }
                prev_pout = pout;
            }
            //Add code to update EPWM register
            //In this basic example, we will implement the controller in the main function
            if (duty >= 0.95) {
                duty = 0.95;
            } else if (duty <= 0.05) {
                duty = 0.05;
            }
            duty_cmp = duty * EPWM_TBPRD;
            EPwm1Regs.CMPA.bit.CMPA = duty_cmp;

            DELAY_US(wait_time); // Insert a delay longer than the switching period to ensure that the duty ratio
                                 //   is updated at most once per switching period (we are implementing single-update PWM)
}
```

PCB Layout

Parasitics

Parasitic inductance and Capacitance introduce behaviors that are not desired in your convertor, parasitic capacitance can couple high switching nodes in your circuit to other sensitive circuitry and signals in your convertor. The same can apply to parasitic inductance where it may cause delays and over stress some components in the high switching nodes of your converter.

To minimize the parasitic capacitance, one can minimize area of traces to and remove copper from the layer directly above or below those nodes. To minimize the parasitic inductance, you can minimize loops in your circuit for example the gate drive loops.

Other Layout Considerations

Placement of sensitive analog circuitry should be away from high dv/vt and di/dt nodes, a ground shield can be implemented on one of the layers to shield sensitive circuitry from noise and coupling. Ground pours can be placed to minimize ground return path distance and also aid in dissipating heat. It is also recommended to use via farms to reduce the distance between the ground pours and various layers of the PCB.

In general you should prioritize minimizing parasitics in critical nodes by reducing loop lengths and area for high dv/dt nodes. You should maximize area for high current carrying nodes to reduce the resistive loses and prevent heating up the PCB. You should also prioritize placement of components such that they are not affected by coupling and noise and ensure adequate cpacing between traces to avoid the same.

My schematic and Layout can be found at the end of the document in the Appendix which demonstrates some of these layout considerations.

Experimental Setup and Operation

A good systematic strategy for assembling and ramping up a power converter would be to assemble and test in batches. First solder on your gate drive circuitry and probe your PWM high and low as well as you Gate high and low signals to ensure that it is working as expected. You should also ensure that the bootstrap circuitry is working as intended. From here you should solder you switches and ensure that they switch as intended. Solder in the rest of your power stage and ensure that you converter is able to output correctly under load. After that solder in your sensing and filter circuitry and ensure it is working at no load and lower voltages first and then ensure that it is working at correctly under load. Throughout this process you should probe various nodes and current using the oscilloscope to ensure that they are as you expect especially the Vin, Vout, Vsw nodes and current.

To emulate the loads for your convertor, you can connect the output to an Eload in the CC (constant current) setting and set the current to either 4.165 A or 8.33 A for either 50 W or 100 W load emulation respectively. To emulate the source, you should set the Eload to a constant Voltage

setting at 12V and connect the PV panel in Parallel to you convertor, this will emulate the PV panel source input to your converter.

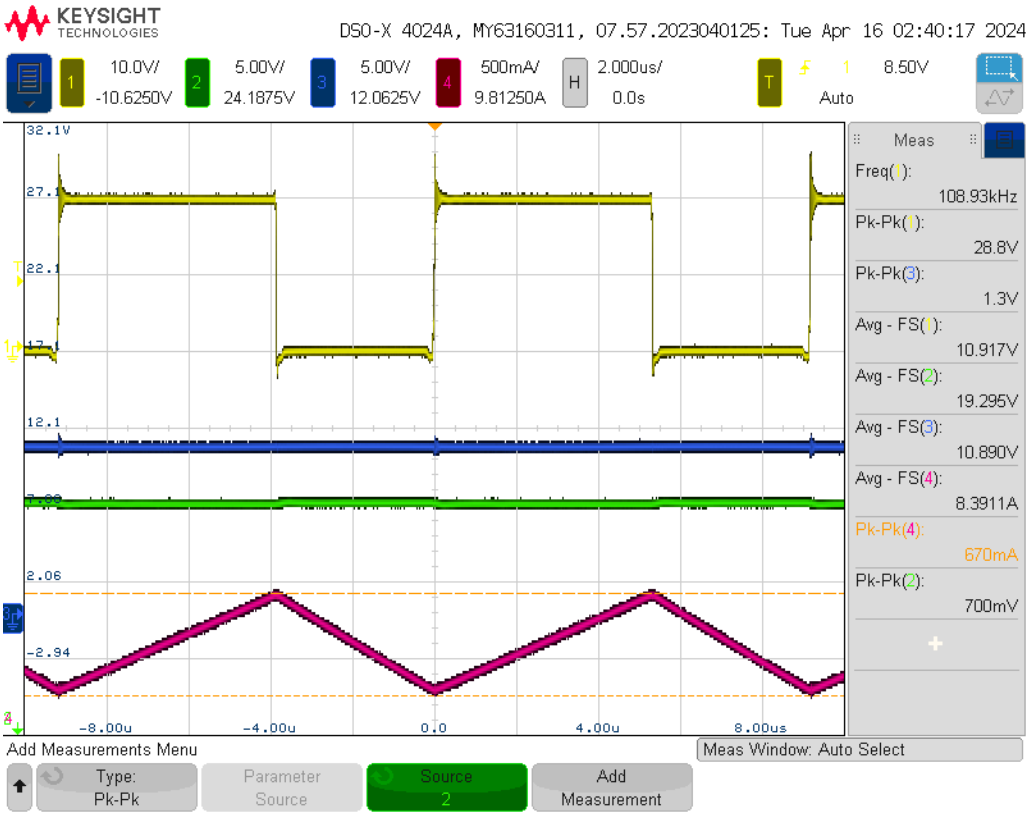
The Oscilloscope is a very useful tool in the debugging arsenal and can help you visualize how different signals look so that you can compare them to what you expect them to be. You can also use the oscilloscope to measure values for different signals like pk-pk measurements, avg measurements, frequency and much more. In addition to the oscilloscope and other useful tool is the multimeter that allows you to verify component health by probing them for either resistance or capacitance to make sure they have the correct values, probing voltage at different nodes and checking for shorts in your circuit using the continuity mode. Thermal camera is also useful to see the heat generating and temperature of your convertor to see where your hotspots are and how you can improve these in layout/using heatsinks or different component choices.

As an example for the thermal observations, my converters high side switch was the hottest component and maxed out at about 50°C with a heatsink as observed from the thermal cameras. My inductor was a far second place about 35°C followed by my low side switch that was slightly above ambient at 30°C. These observations align with my calculations and expected temperatures.

My converters power density was about 0.957 cm^3/W at the 100W operating point.

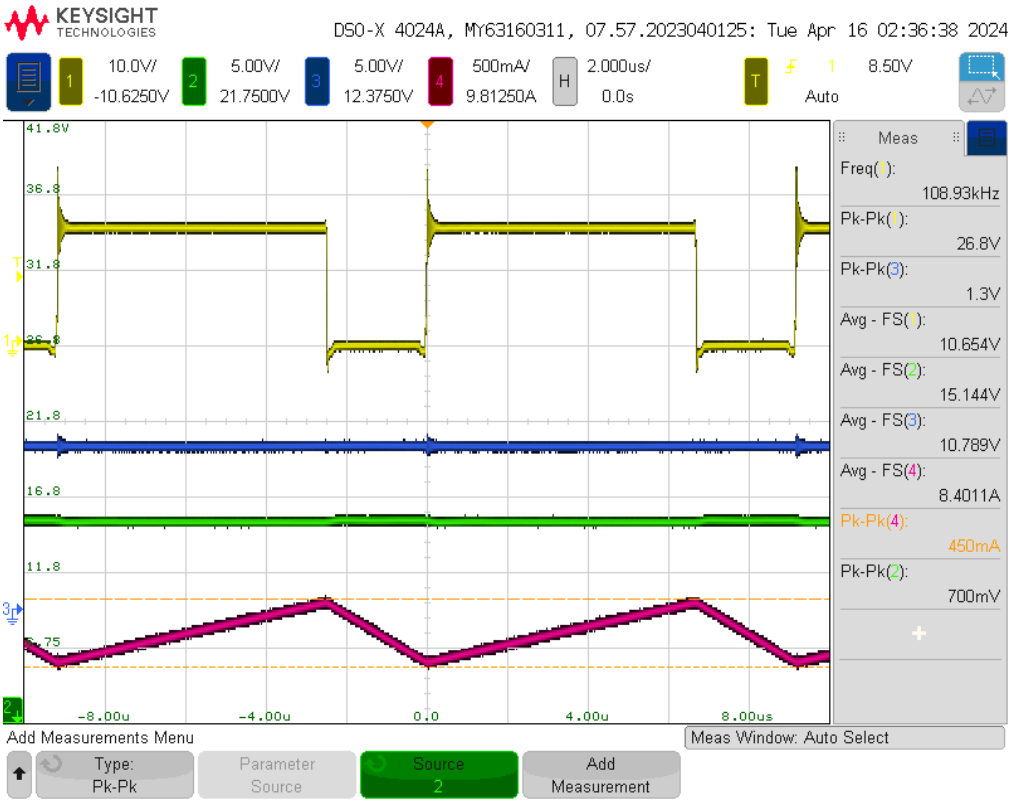
Below are some scope shots for important nodes (Vin, Vout, Vsw) and current at all operating points to visualize how these signals look.

Nominal:

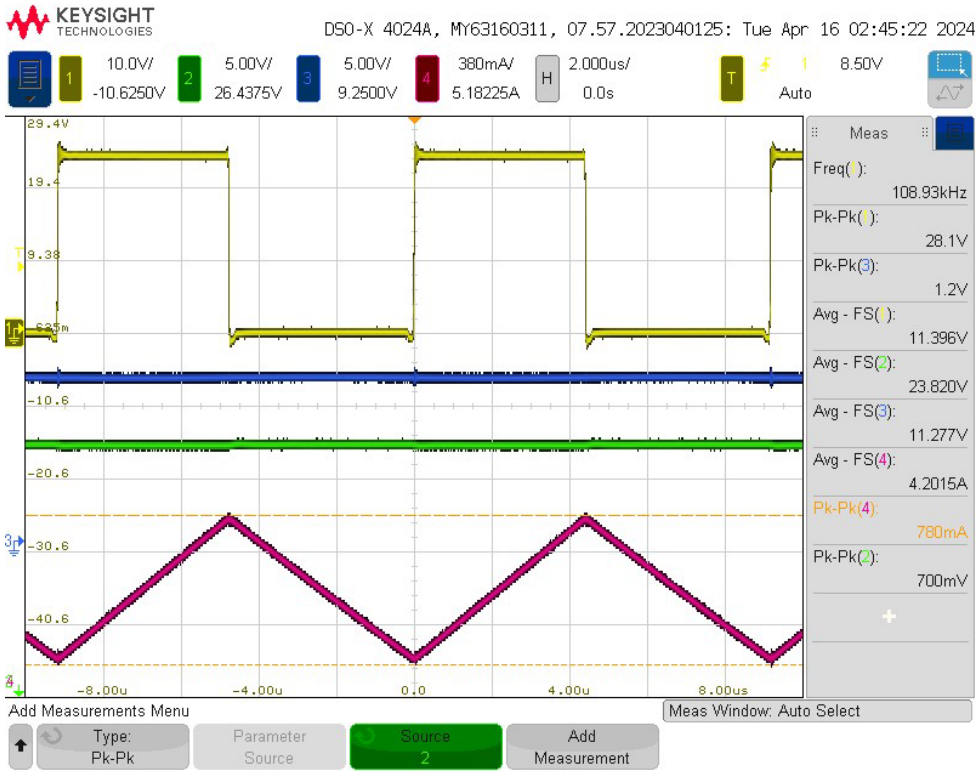


24V @ 100W:

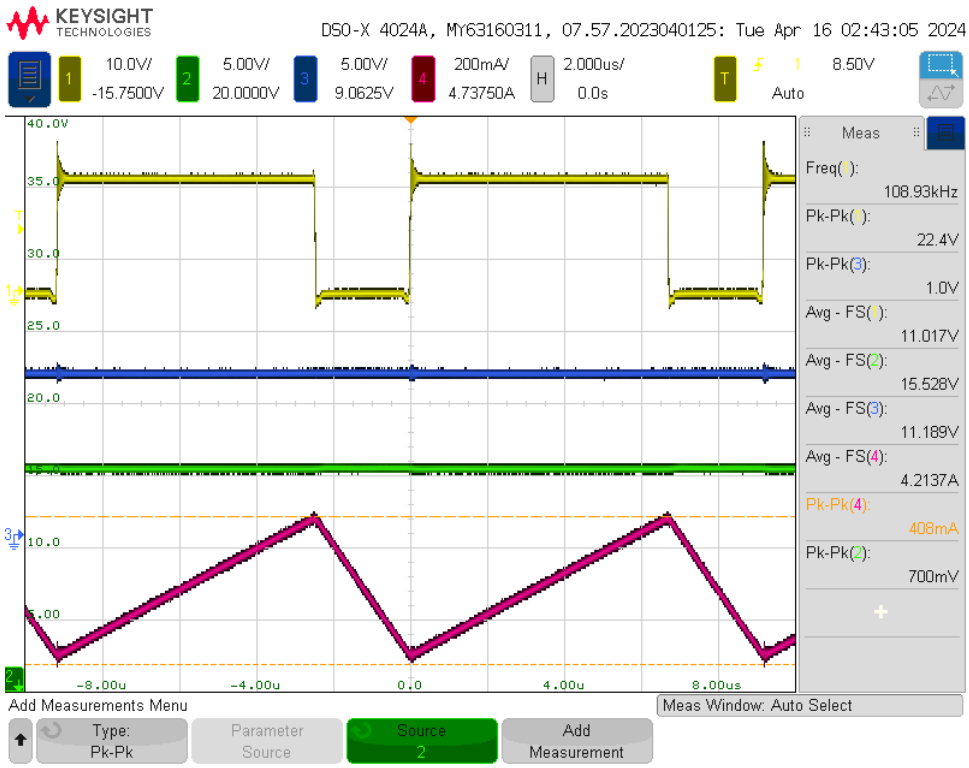
16V @ 100W:



24V @ 50W:



16V @ 50W:



Experimental Efficiency

To measure the power stage efficiency, you can calculate the input power using the Oscilloscopes input voltage value (since it is the voltage on the input of the converter hence eliminating drop from the wires from PSU) and multiply this with the input current from the PSU. To get the Output power, you can again use the output voltage value from the oscilloscope and multiply this with the current value on the Eload. Then get efficiency by; Pout / Pin.

To get Efficiency Sweeps, you can create a script to run your Eload and and Power supply to sweep through multiple operating points and plot efficiency values at these operating points. While doing this, you should use the 4 sense pins on the Eload and PSU to sense the voltages at the input and output of your converter hence mitigating voltage drops from the wires.

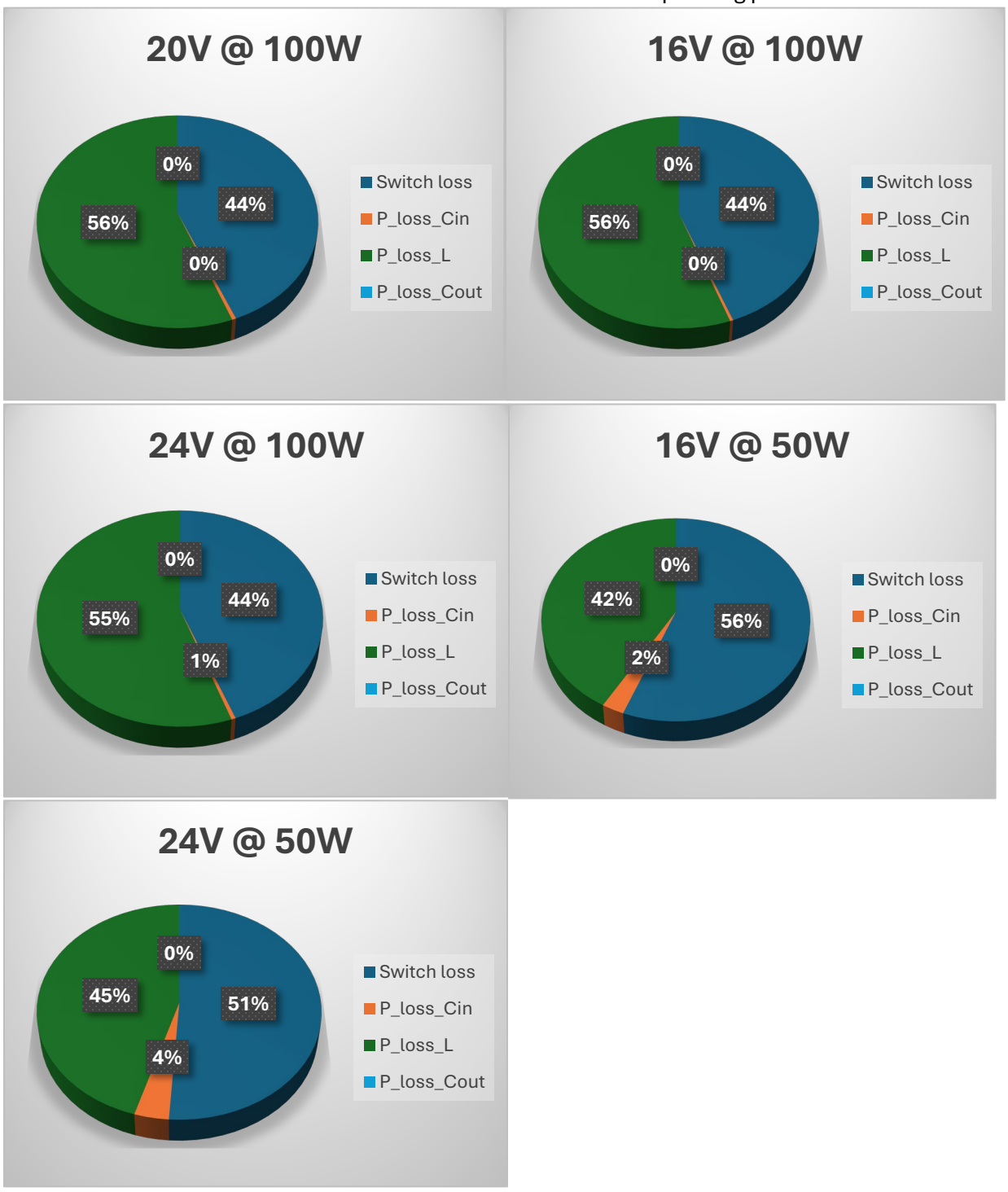
There are a plethora of ways to improve efficiency of the converter from minimizing Power losses in component selection, reducing resistive losses in windings by increasing gap to meet inductance requirements, reducing trace lengths and increasing thickness or using pours on high current nodes. Minimizing overlap loss and adding sufficient deadtime to prevent shorting the source. Adding adequate decoupling to ensure that noise is not coupled with gate signals to turn on gate when not desired. And ensuring that all solder joints are well done to minimize contact and joint resistance.

Loss Breakdown

Here is a table that shows the breakdown of losses in my converter and my expected efficiency at different operating points.

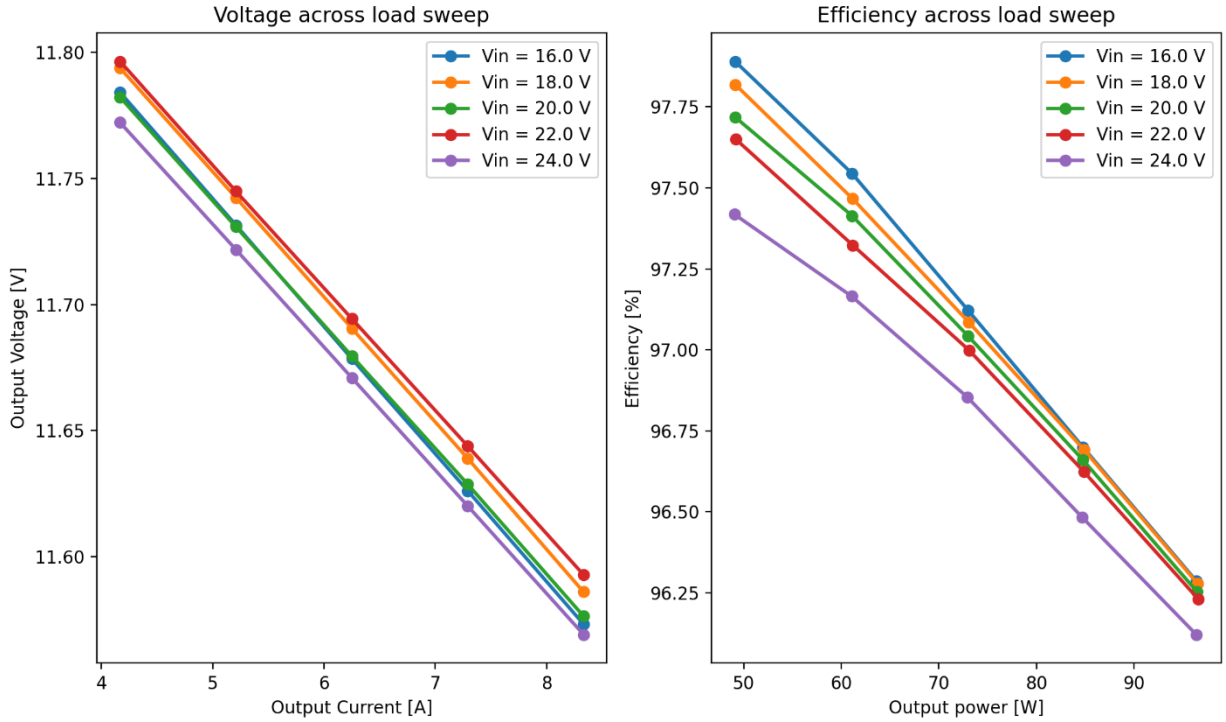
Operating Point	Switch loss	P_loss_Cin	P_loss_L	P_loss_Cout	Total Loss Calculated	P_loss_Measured	Efficiency
20V @ 100W	1.14	0.0112	1.4558	0.0001	2.6071	5.2125	94.5943
16V @ 100W	1.16	0.0088	1.4566	0	2.6254	5.082	94.65312
24V @ 100W	1.16	0.0117	1.455	0.0001	2.6268	5.3304	94.47878
16V @ 50W	0.49	0.022	0.364	0	0.876	1.41855	97.08165
24V @ 50W	0.41	0.029	0.3637	0.0001	0.8028	1.6303	96.67476

Below are some Pie charts to visualize these losses at different operating points:



My converter is the least efficient at the highest operating point which is 24V at 100W because the Vsw nodes ringing is the highest which couples with the gate signals and shorts my source briefly, it is also the highest voltage across the switches and hence increasing the conduction losses on the switches. And my most efficient point is the 16V at 50W operating point because the current is the lowest and hence lowest losses as well as lower stresses on switches and inductor.

Below are efficiency and voltage regulation sweeps for my converter across multiple different operating points:



My Gate drive/Sensing Circuit (Auxiliary) Power draw was 14V * 0.013A = 0.182 W.

- Efficiency at nominal operating point considering this power draw = 97.4 – 0.182 = 97.22 %

And my final efficiency score was:

- 0.4(96.25438)+0.15(97.8908)+0.15(96.28579)+0.15(97.41817)+0.15(96.12149) = 96.659%

Experimental MPPT

To evaluate the MPPT tracking efficiency, you can set different power conditions and measure the power output from your converter across multiple points and average these and compare them to the power you expect based on the PSU and Eload settings to get the MPPT tracking efficiency.

For an automated evaluation, MPPT can be evaluated by running a script that sweeps between different input voltages and output loads and hence predicting and x% efficient converters output, say 100%. Then the power output from the converter under test is calculated by output voltage and current. This is repeated with different operating points for a set amount of time at each point while recording the output from the converter to see how the MPPT performs against dynamic conditions. These outputs are then averaged and compared against the averaged power from the script to measure the tracking efficiency.

The MPPT can be improved with better readings and hence adding low pass filters to the ADC would help get more steady and reliable reading thus improving the MPPT performance.

The steady state tracking efficiency for my MPPT algorithm is shown in the table below:

	Full	Half
IV_sweep (W)	100.397774	48.67139231
Convertor MPPT (W)	99.3	46.9
Tracking Efficiency (%)	98.9065754	96.36050619

This is not 100% because of reading limitations and this is seen at the lower power point where the current reading becomes even smaller and hence utilizes a lower portion of the ADCs range thus reducing the resolution of the readings. Another reason is because the duty step may be too large to make fine adjustments to reach 100% or closer to 100%.

As for the dynamic MPPT score, my converter averaged at an output power of 77.41 W.

BOM and Budget

PARTS: CONNECTORS LIBRARY		Unit Price	Quantity	Cost [$/board]
Header connectors (10 pos)	61301011121	0.95	1	0.95
Header connectors (4 pos)	M20-8760242	0.39	1	0.39
Test point	5001	0.38	12	4.56
Banana jack, blk	CT3151SP-0	2.45	3	7.35
Banana jack, red	CT3151-2	2.45	3	7.35
				0
PARTS: IC LIBRARY				0
Half bridge gate driver	LM5109BMAX/NOPB	1.03	1	1.03
Op amp	LM358DR	0.41	1	0.41
Current sense amplifier	LT1999CS8-50	6.24	1	6.24
Voltage reference - 1.25 V	ADR1581ARTZ-REEL7	2.26	1	2.26
LDO	LP2985-50DBVR	0.55	1	0.55
				0
PARTS: SWITCH LIBRARY				0
Diode	1N4148W-7-F	0.17	3	0.51
TO-220 55V 64A	IRFZ48NPBF	1.27	2	2.54
TO-220 heat sink	507302B00000G	0.3	2	0.6
Thermal pad	QII-0.006-00-54	0.43	2	0.86
				0
PARTS: INDUCTOR LIBRARY				0

RM 12 cores (2 piece set)	RM12/I-3C95	6.53	1	6.53
RM 12 bobbin	B65816C1512T001	1.94	1	1.94
RM 12 clamp	CLI/P-RM12/I	0.49	2	0.98
Magnet wire 16 AWG		N/A	1	
Shim stock		N/A	1	
				0
PARTS: CAPACITOR LIBRARY				0
Electrolytic, 470 uF	50ZLJ470M10X20	1.03	2	2.06
MLCC, 4.7uF, 35V, X7R, 0805	GRM21BZ7YA475KE15K	0.3	1	0.3
MLCC, 10uF, 50 V, X7R, 1206	C1206X7R500-106KNE-CT	0.26	6	1.56
2.2 uF, Commutation loop decoupling, 0603, 50V	GRM188R61H225KE11J	0.28	3	0.84
10 uF, 25 V Decoupling for gate drive, LDO, lig reg, current sense	GRM188R61E106KA73J	0.26	2	0.52
1 uF, 50V	0603X105K500CT	0.13	2	0.26
.1 uF, 25V Current sense, op amp, regulator input decoupling	C0603Z5U250-104MNP-CT	0.0094	3	0.0282
LDO bypass, 10nF, 0603, 3 V	C0603X7R250-103MNP-CT	0.01	1	0.01
Filter cap, 330 pF, 50V, NPO, 0603	C0603C0G250-331KNE-CT	0.01	7	0.07
Current sense SHDN cap, 10 pF, 5V max, 0603	C0603C100J3HAC7867	0.1	1	0.1
				0
PARTS: RESISTOR LIBRARY				0
boostrap and gate, 1 ohm, 0603, 1/5 W	AC0603FR-7W1RL	0.1	3	0.3
filter and gate, 10 ohm, 0603, 1/10 W	AGCR0603-QK-10R0FT-CT	0.01	7	0.07
current sense resistors, 2.5m, 1%, 0612	D1FCP0612D2M50FF-T5	0.57	1	0.57
current sense, 2.94kΩ, 1%, 0.1W, 100ppm/°C, 0603	CRCW06032K94FKEA	0.1	1	0.1
resistor dividers, 47k, 0603	ERJ-PB3B4702V	0.26	1	0.26
resistor dividers, 100k, 0603	ERJ-PB3B1003V	0.26	1	0.26
resistor dividers, 470k, 0603	RN73R1JTTD4703B25	0.21	2	0.42
TOTAL COST [$]				**52.7782**

Conclusion

This design process has been incredibly insightful and has taught me a lot about converters and how to design one from scratch.

I wanted to highlight some future improvement plans for the converter, I intend to reduce my Vsw decoupling loops as well as my gate loops to reduce the switching losses as a result of coupling. I intend to reduce my rise time and increase the fall times for the gate signals in order to prevent shorting on transitions by adding resistors on the gate drive signals.
For the MPPT algorithm I want to implement the changing duty step every so often to prevent being stuck at a local maximum. I also want to experiment with variably frequency in the MPPT within a certain range that meets ripple spec to increase efficiency and thus output power.

If I were to redo my design, I would reduce my parasitic inductance loops and reduce the area for my parasitic capacitances.

Overall this converter performed very well and can be even better with the slight improvements mentioned above.

Lastly I would highly recommend undergoing this experience as I have learned and grown immensely through this design and bring up.