**Final Report: Closed-loop DC-to-DC Step-down Converter**

Elick D’Rozario

Adrian Fernandez

Professor Levine

ENEE 461 Section 0101

May 18, 2023

**Introduction**

Our project consisted of building a DC-to-DC step-down converter which used a feedback loop to regulate the output voltage of the converter. The DC-to-DC step down converter will be addressed as a buck converter from this point forward. The feedback loop was needed due to the buck converter operating in discontinuous mode. This meant that if we changed the resistor load of the converter the output voltage would also change. This background information is covered in the Buck Converter Background Information section. We were then able to implement this by creating the mentioned feedback loop and using it to create a pulse width modulation signal which would drive the buck converter. This is covered in the Buck Converter with Feedback section. In addition, we implemented a buck converter with no feedback which simply used a fixed pulse width modulation signal. This is covered in the Buck Converter with No Feedback section. Finally, we review and analyze our results in the Conclusion section.

# **Hardware/Software Used**

1. Quanser Q8-USB Data Acquisition Board
2. Computer with MATLAB and Simulink
3. Quanser Universal Power Module
4. Breadboard
5. Battery 9V
6. N-Channel Power MOSFET
7. Schottky Diode
8. Inductor 100μH
9. Capacitor 1000μF
10. Resistor 100Ω
11. Resistor 1000Ω
12. Resistor 10kΩ
13. Wires of Differing Lengths

**Buck Converter Background Information**

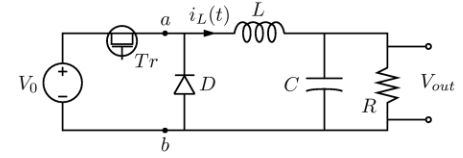
A buck converter is a DC-to-DC converter that takes an input voltage and steps it down to a smaller output voltage. Buck converter efficiency is relatively high, usually over 90% and is used in common electronic applications like converting a PC’s main power supply voltage to different lower voltages that are used by USB ports, the CPU, the RAM, etc. The usual circuit of a buck converter contains a diode, transistor, capacitor, inductor, and resistor. The capacitors and inductors work together to decrease voltage ripple. A normal buck converter operates at a frequency that ranges from 100*kHz* to about a few *MHz*.

We begin with an analysis of the buck converter shown in figure 1. The basic operation of the circuit involves having the transistor be turned on and off. When the transistor is on the diode is reverse-biased and off, and the input voltage is applied across terminals a and b. When the transistor is off the diode is turned on and no voltage is applied to terminals a and b. This allows us to simplify the circuit and eliminate the input voltage, the transistor, and the diode in favor of an equivalent voltage as can be seen in figure 2. Figure 3 displays the equivalent voltage’s rectangular pulses which involves , the period of repeated switching, and , the fraction of this period when the transistor is on. We now focus on the discontinuous mode of the buck converter which involves having the inductor current, , reach zero during the time interval when the transistor is off and stay at zero until the transistor turns on. This is crucial for our circuit since the discontinuous mode allows us to change the output voltage by changing the resistor load.

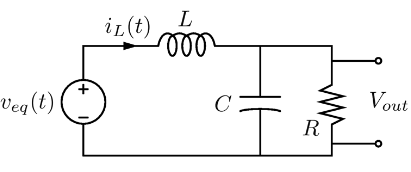
We want to derive the output voltage, , during the discontinuous mode of operation of the buck converter in terms of . We start by assuming this is an ideal buck converter with no losses where the input power equals the output power as displayed in equation 1. Next, we transform equation 1 in terms of voltage and current as shown in equation 2. We now derive an expression for the input current by analyzing figure 4 which displays the inductor current as a function of time. We substitute the input current from equation 3 into equation 2 which relates the input power and the output power and obtain equation 4. By rearranging this equation we obtain equation 5 which shows that we can solve for the output voltage by using the quadratic equation. Finally, we obtain equation 6 which expresses the output voltage in terms of the input voltage, the duty ratio, the period, the inductor, the resistor, and the capacitor. It is important to note that although equation 6 displays the average output voltage it is assumed that since we are dealing with DC voltages.

Now that we have derived the output voltage of a buck converter we focus on the controls aspect of this project. To obtain the same output voltage in the face of changing resistance we need to implement a feedback control loop. We start by displaying the transfer function of the buck-converter (the plant) in equation 7. Next, we display the overall transfer function of the system which includes the plant and the controller in equation 8. Depending on the controller we choose we will have different transfer functions for the different controllers. The feedback control system is displayed in figure 5. A detailed explanation of this entire diagram will now follow. The reference voltage is the voltage we want the buck-converter to achieve and the output voltage is the actual output voltage of the buck-converter. The error is the difference between these two voltages which then serves as the input for the controller. The controller will then output , the duty ratio of the buck-converter. This will allow us to change the resistor load and maintain the same output voltage by changing the duty ratio of the buck-converter.

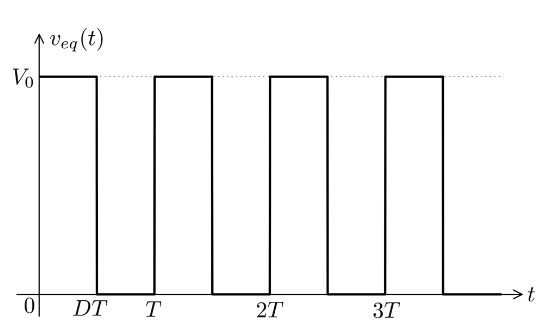
Overall, we have shown that varying the resistor load in a buck-converter in discontinuous mode will change the output voltage. However, through the use of a feedback loop we can ensure that the output voltage stays constant no matter the resistor load by varying the duty ratio.



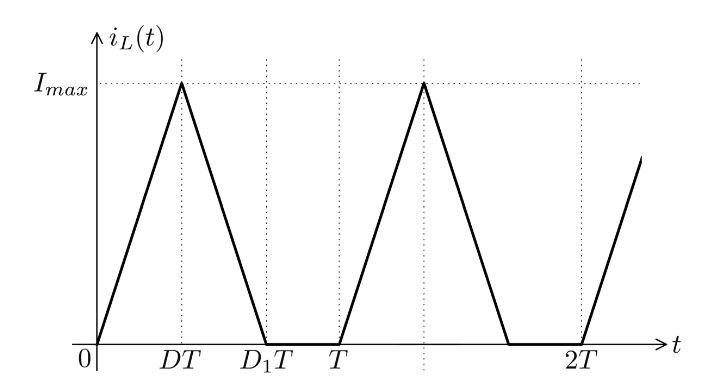
*Figure 1: Circuit diagram of a buck converter composed of an input voltage, a transistor, a diode, an inductor, a capacitor, and a resistor.*



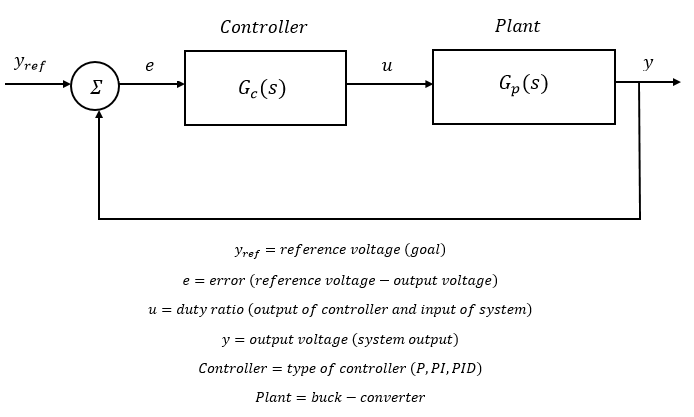
*Figure 2: Simplified buck converter when the transistor is off.*



*Figure 3: Rectangular pulses of the equivalent voltage involving the period of repeated switching, T, and the fraction of this period when the transistor is on, D.*



*Figure 4: Inductor current with respect to time of a buck converter during discontinuous mode.*



*Figure 5: Feedback control system where is the reference voltage we want the buck-converter to achieve, is the error between the reference voltage and the output voltage, is the duty ratio, and is the output voltage of the buck-converter.*

**Buck Converter with Feedback**

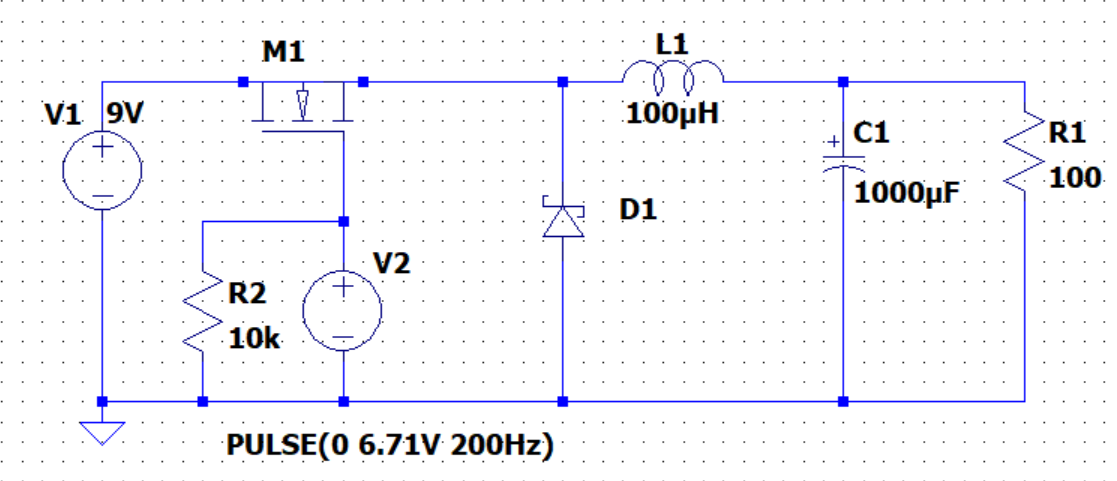
The first circuit we implemented was the buck converter with a feedback loop. The circuit was composed of two parts: the actual buck converter and the *Simulink* model which drove the pulse width modulation signal. The schematic for the buck converter is shown in figure 6 and the real-life circuit of this schematic is shown in figure 7. In addition, the *Simulink* model is shown in figure 8. The physical buck converter was created with the following four circuit elements: Vin = 9.00V, L = 100μH, C = 1000μF, and R = 100Ω or 1000Ω. The *Simulink* model received one input and produced one output which were connected to the circuit through the Quanser Q8-USB Data Acquisition Device displayed in figure 9. The *Simulink* model calculated the difference between the goal voltage and the buck converter output voltage to generate the error. The error signal was then passed through an absolute value block to ensure the error was never below zero. It was then used as one of the inputs to the relational operator which would generate the PWM signal. The second input of the relational operator was a sawtooth waveform with a frequency of 200*Hz* and an amplitude of 4.00V. These two inputs were compared and whenever the error was less than or equal to the sawtooth waveform the output would be one. If the error was greater than the sawtooth waveform the output would be zero. The output was then multiplied by 6.71 in order to properly drive the buck converter MOSFET. This approach allowed us to change the PWM duty ratio depending on the error calculated between the desired output voltage and the actual output voltage.

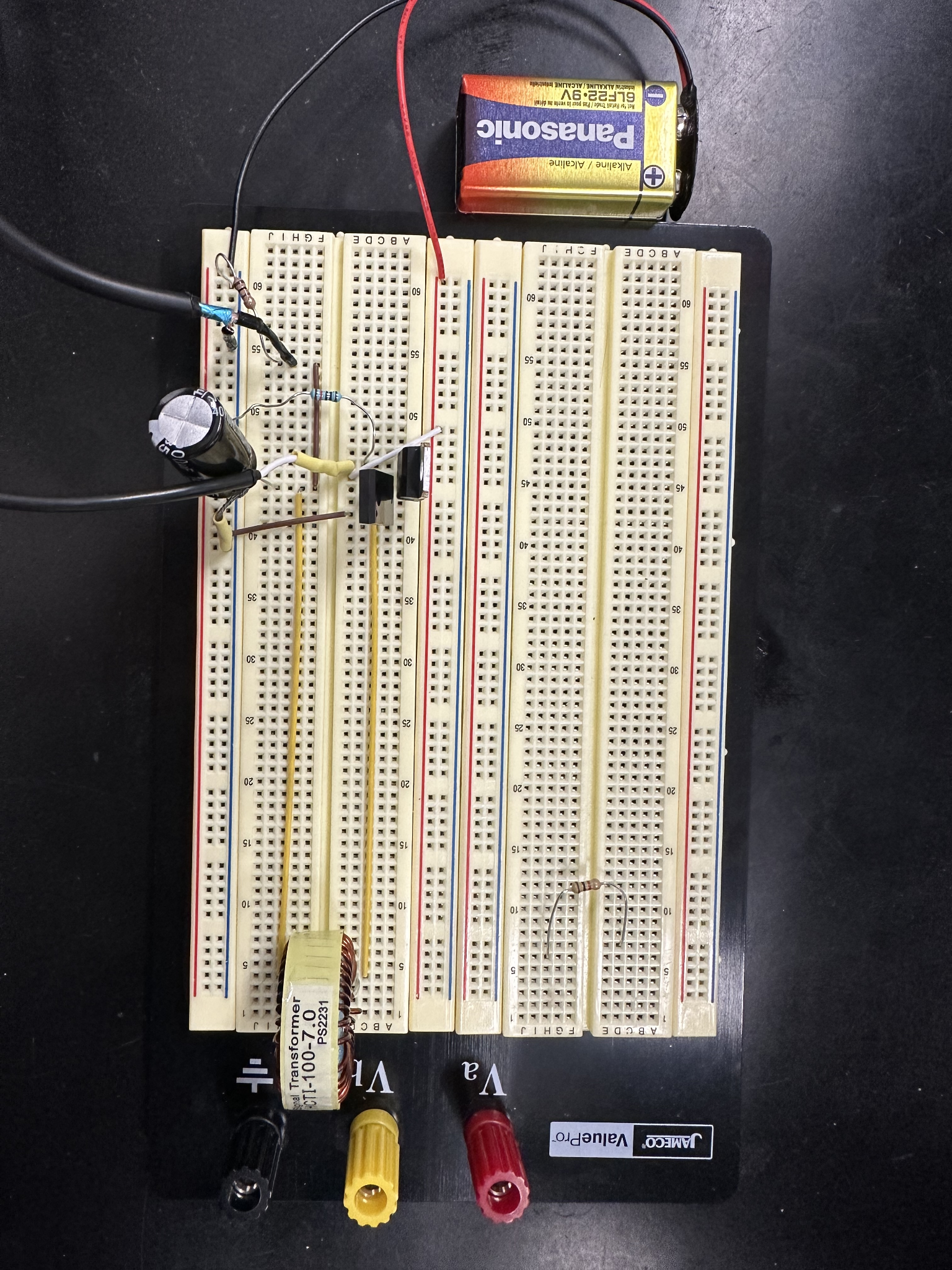
We now move on to our results of the closed-loop buck converter when the resistor load is 1000Ω. Figure 10 displays the buck converter output when the resistor load is 1000Ω. Initially, there is no output voltage since the input voltage was not connected due to the output response being extremely fast. The input voltage is connected at around the 2.9 second mark and we see the output voltage rise to 3.21V. We note that the output surpasses the goal due to the low frequency used. The frequency of 200*Hz* was chosen due to hardware limitations involving the Quanser Q8-USB device. We believe the frequency is the culprit since buck converters are usually operated at frequencies of hundreds of *kHz*. Analyzing the response of the output voltage in figure 10a reveals a rise time of 9*ms*, a settling time of 35*ms*, a steady-state error of 0.21V, and a 7% overshoot.

Finally, we move on to our results of the closed-loop buck converter when the resistor load is 100Ω. Figure 11 displays the closed-loop buck converter output when the resistor load is 100Ω. Initially, there is no output voltage since the input voltage was not connected due to the output response being extremely fast. The input voltage is connected at around the 4.4 second mark and we see the output voltage rise to 2.86V. We note that the output never reaches the goal. Analyzing the response of the output voltage in figure 11a reveals a rise time of 2*ms*, a settling time of 2*ms*, a steady-state error of 0.14V, and no percent overshoot.

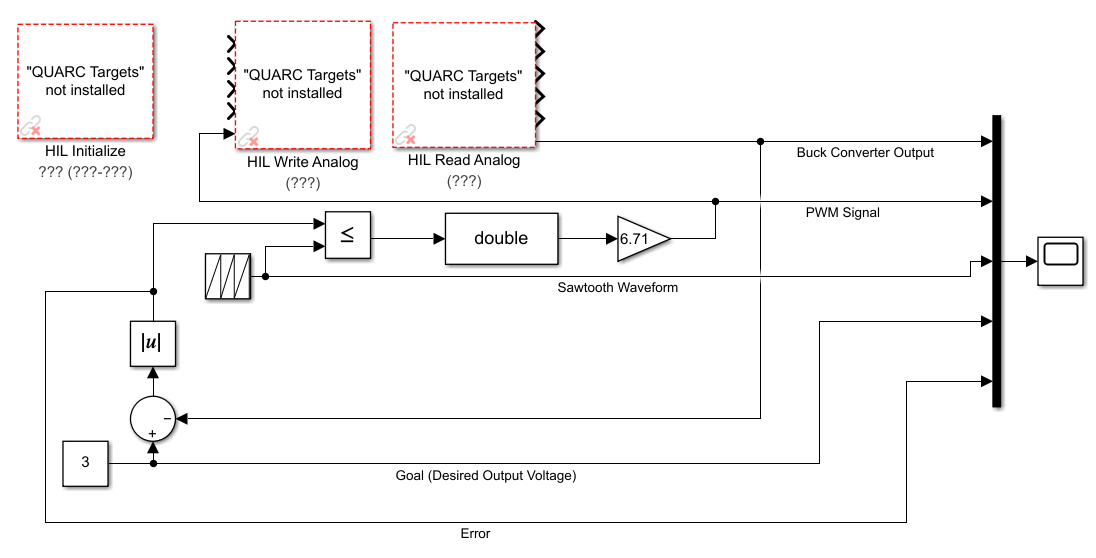
The calculation of the pulse width modulation signal is shown in figure 11b. The error is the absolute value of the difference between the goal and the output voltage. This error is compared to the sawtooth waveform shown in figure 11b. The sawtooth waveform had an amplitude of 4.00V and a frequency of 200*Hz*. Whenever the error was less than or equal to the sawtooth the output was one. Whenever the error was greater than the sawtooth the output was zero. This created a PWM signal from the feedback loop we previously discussed which regulates the output voltage effectively. There are some instances where the PWM signal turns on either too early or too late, but this can be attributed to the fact that the error comes from a feedback loop so it takes additional time for the error to be calculated.

Overall, we believe the closed-loop buck converter was a success since we were able to track the goal with a maximum steady state error of 0.21V which represents an accuracy of 93% with respect to the goal. In addition, the next circuit to be discussed involves the open-loop buck converter. In this circuit we manually changed the duty ratio of a *Simulink* generated PWM signal from zero to one. In comparison, our closed-loop buck converter was able to achieve good performance without any interaction from us. Finally, it achieved what we set out to do which involved maintaining a constant voltage in the face of a changing resistor load.

*Figure 6: Schematic of the buck converter constructed. The primary voltage source (V1) is 9 volts. The PWM signal (V2) is 6.71V with a frequency of 200Hz. The inductor (L1) is 100μH and the capacitor (C1) is 1000μF. The load resistor (R1) is either 100Ω or 1kΩ. The MOSFET (M1) is an n-channel MOSFET and the diode (D1) is a Schottky diode. The second resistor (R2) is there to ensure the MOSFET gate does not float “high.”*

****

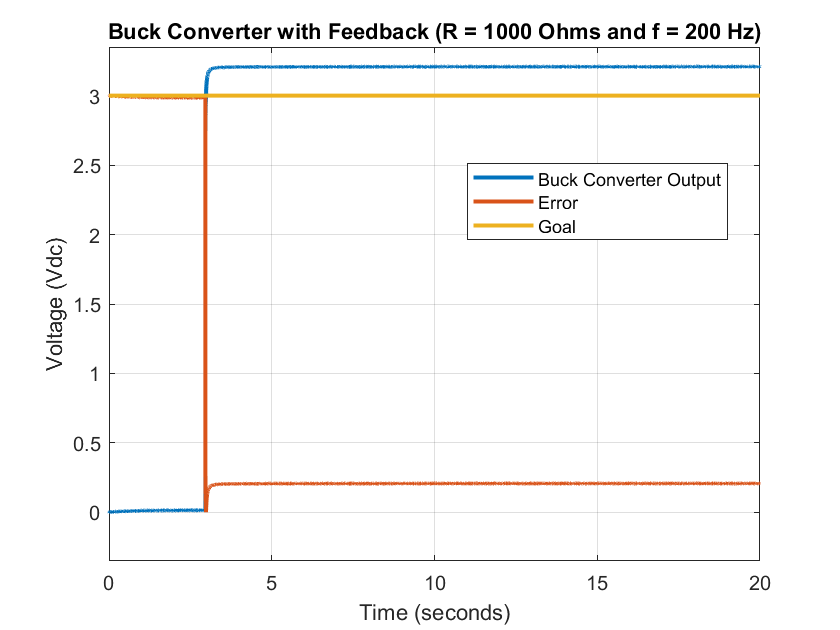
*Figure 7: Full real-world circuit set-up of the buck converter as shown in figure 6. The main circuit is in the bottom left corner. The inductor is in the bottom right corner and the capacitor is in the bottom left corner. The 9V battery is connected to the converter through the red and black wires shown on the left. The wires with the yellow plastic cladding are used to output the PWM signal to drive the MOSFET. The wires with the black plastic cladding are used to measure the buck converter output voltage from the resistor load. The single resistor in the top right corner is there for the second resistor load used on the buck converter. This is because we alternated between using a load resistor of 100Ω or 1kΩ.*

****

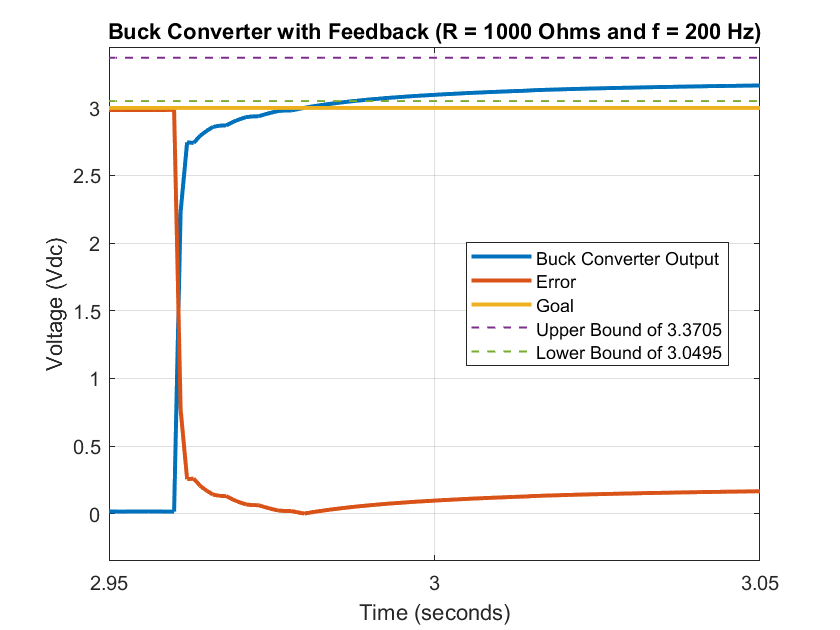
*Figure 8: Simulink model which implemented a feedback loop to the buck converter by creating a pulse width modulation signal. The pulse width modulation signal was created by calculating the difference between the goal voltage and the output voltage to generate the error. The error signal was then passed through an absolute value block to ensure the error was never below zero. It was then used as one of the inputs to the relational operator which would generate the PWM signal. The second input of the relational operator was a sawtooth waveform with a frequency of 200Hz and an amplitude of 4.00V. These two inputs were compared and whenever the error was less than or equal to the sawtooth waveform the output would be one. If the error was greater than the sawtooth waveform the output would be zero. The output was then multiplied by 6.71 in order to properly drive the buck converter MOSFET.*



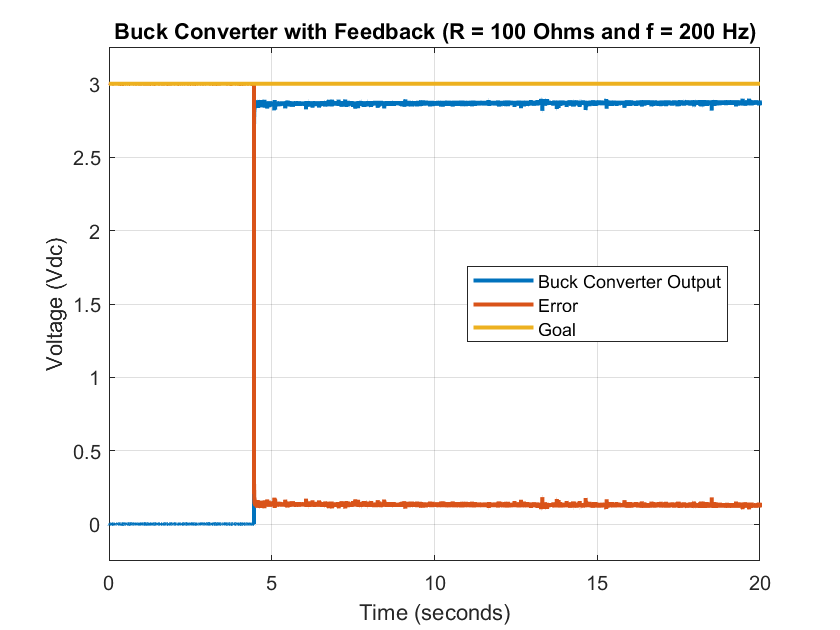
*Figure 9: This device is the Quanser Q8-USB Data Acquisition Device which helped record the buck converter output voltage and generate the pulse width modulation signal. This was done by using the analog output and analog input ports shown above. These were then connected to the circuit through pins soldered onto the cables.*

****

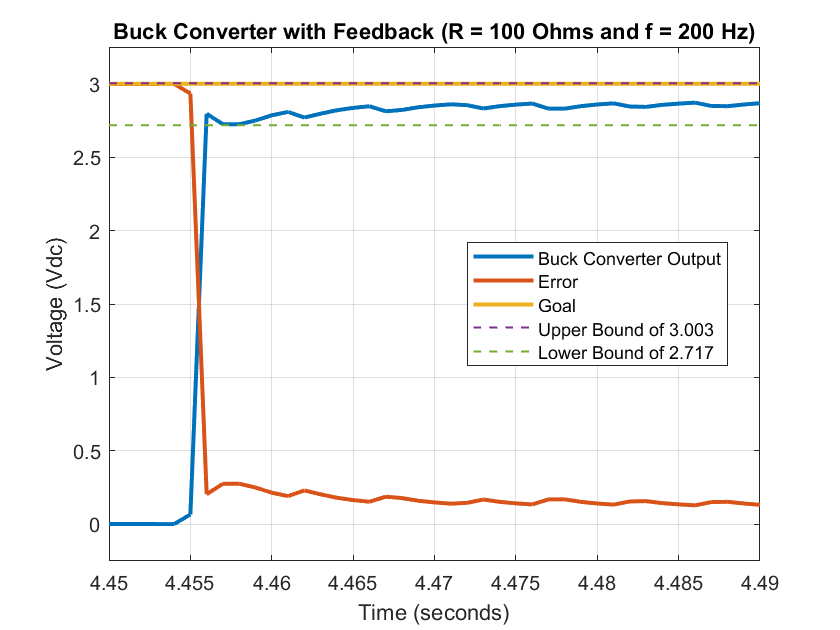
*Figure 10: This figure shows the closed-loop buck converter output when the resistor load is 1000Ω and f = 200Hz. Initially, the input voltage is disconnected from the circuit since the response of the output voltage is extremely fast. We connect the input voltage at around the 2.9 second mark and see the output voltage rise to We note that the output surpasses the goal due to the low frequency used. The frequency of 200Hz was chosen due to hardware limitations involving the Quanser Q8-USB device.*

**

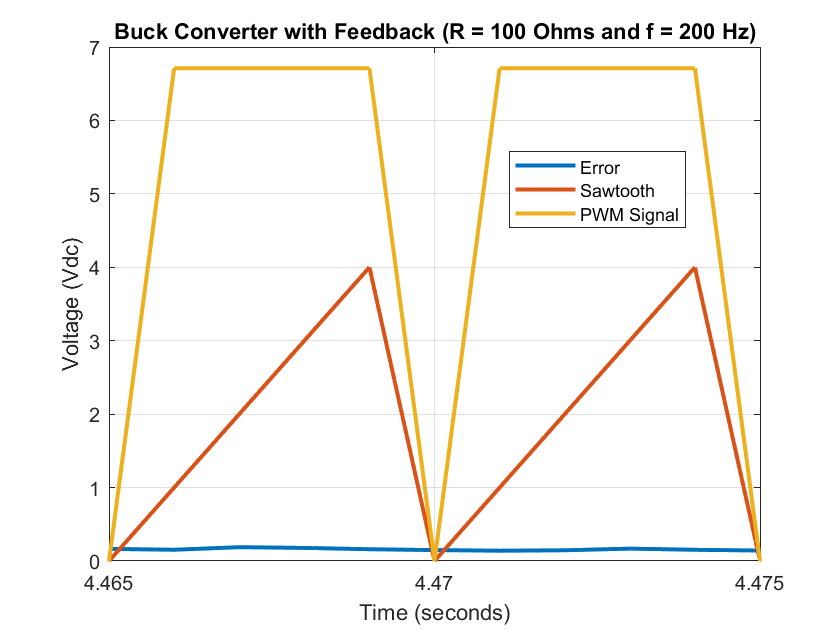
*Figure 10a: This is a zoomed-in version of figure 10 which focuses on the response of the output voltage. We note that the output voltage has a rise time of 9ms, a settling time of 35ms, a steady-state error of 0.21V, and a 7% overshoot.*

**

*Figure 11: This figure shows the closed-loop buck converter output when the resistor load is 100Ω and f = 200Hz. Initially, the input voltage is disconnected from the circuit since the response of the output voltage is extremely fast. We connect the input voltage at around the 4.4 second mark and see the output voltage rise to 2.86V. We note that the output does not reach the goal due to a low gain chosen for the PWM signal. This stems from the limitation mentioned in figure 9 since we believe the low frequency limited the ability of the closed-loop buck converter to properly track the goal.*

****

*Figure 11a: This is a zoomed-in version of figure 11 which focuses on the response of the output voltage. We note that the output voltage has a rise time of 2ms, a settling time of 2ms, a steady-state error of 0.14V, and no percent overshoot.*

**

*Figure 11b: This figure shows how the pulse width modulation signal was created for the closed-loop buck converter. The error is the absolute value of the difference between the goal and the output voltage. This error is compared to the sawtooth waveform shown above. The sawtooth waveform had an amplitude of 4.00V and a frequency of 200Hz. Whenever the error is less than the sawtooth the PWM signal turns on. Whenever the error is greater than the sawtooth the PWM signal turns off. This creates a PWM signal from the feedback loop we previously discussed which regulates the output voltage effectively. There are some instances where the PWM signal turns on either too early or too late, but this can be attributed to the fact that the error comes from a feedback loop so it takes additional time for the error to be calculated.*

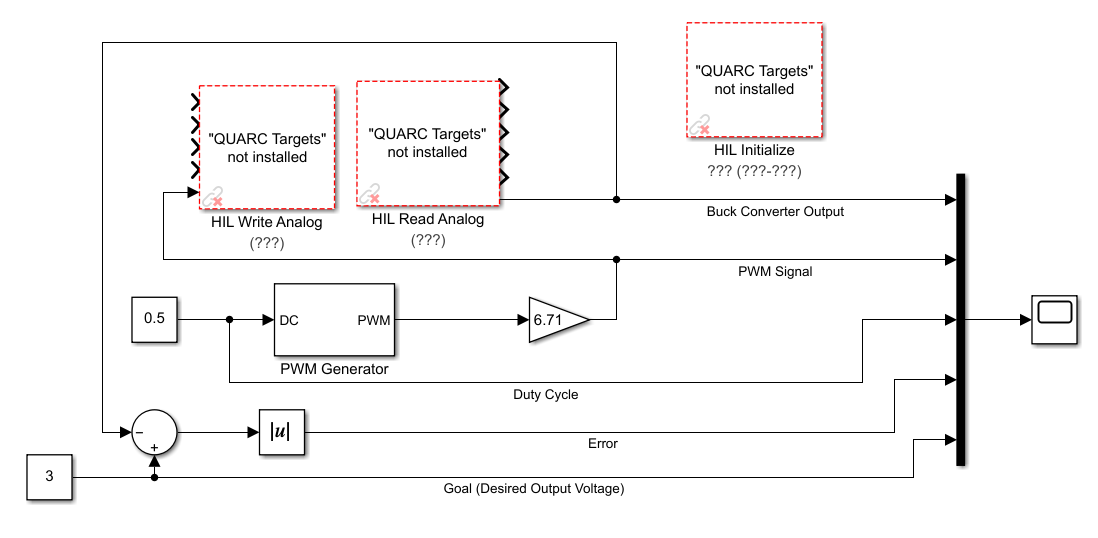
**Buck Converter with No Feedback**

The second circuit we implemented was the open-loop buck converter. The circuit was composed of two parts: the actual buck converter and the *Simulink* model which drove the pulse width modulation signal. The schematic for the buck converter is the same as the closed-loop buck converter and is shown in figure 7. The *Simulink* model is shown in figure 12. The physical buck converter has all the same components as the closed-loop version. The *Simulink* model received one input and produced one output which were connected to the circuit through the Quanser Q8-USB Data Acquisition Device displayed in figure 9. The *Simulink* model calculated the difference between the goal voltage and the buck converter output voltage to generate the error. The error signal was then passed through an absolute value block to ensure the error was never below zero. This is the key difference between the open-loop versus the closed-loop buck converter. The error in this circuit is not used to calculate anything, but instead simply to record and analyze how far the output voltage is from the goal. The PWM signal was generated through a PWM generator block in *Simulink* whose single input was the duty ratio. In addition, one of the parameters of the block was the frequency which we set to 200*Hz*. The duty ratio was manually changed for the different variations of the circuit tested. The PWM signal was then multiplied by 6.71 in order to properly drive the buck converter MOSFET. This approach relied on user input heavily since we had to manually change the duty ratio to obtain a satisfactory output voltage.

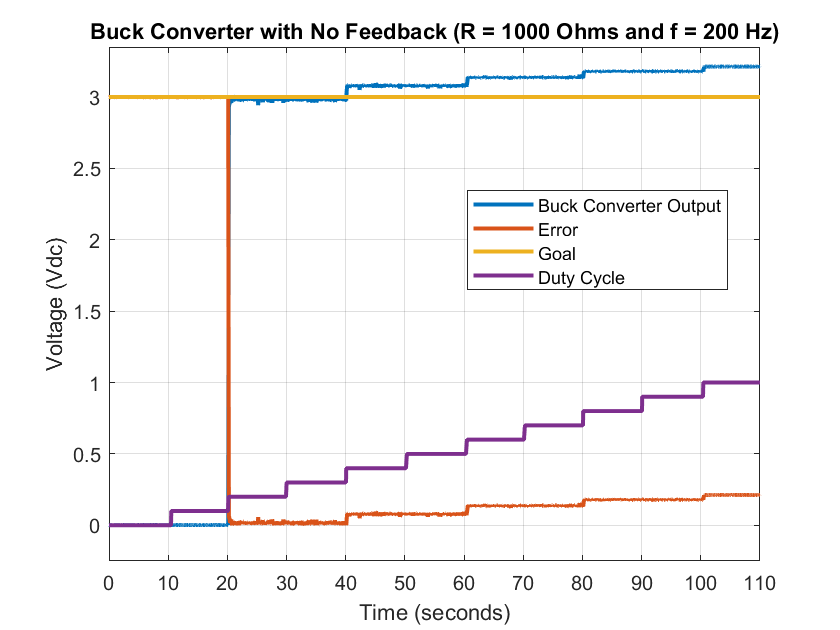
We now move on to our results of the open-loop buck converter when the resistor load is 1000Ω. Figure 13 displays the buck converter output when the resistor load is 1000Ω. For the open-loop buck converter the input voltage is always connected. However, the output voltage is initially zero due to the duty ratio always starting at zero. The duty ratio is increased by 0.1 every ten seconds until we reach a duty ratio of one. We see the output voltage rise to 2.98V as soon as the duty ratio reaches 0.2. We also note that as the duty ratio increases the output voltage also increases, eventually reaching 3.21V and surpassing the goal. As we previously stated in the Buck Converter with Feedback section we believe the low frequency used affected the performance of the converter. This is because the change in duty ratio should have a greater effect on the output voltage, but we only see a total voltage difference of 0.23V. Analyzing the initial response of the output voltage in figure 13a reveals a rise time of 12*ms*, a settling time of 32*ms*, a steady-state error of 0.02V, and no percent overshoot.

Finally, we move on to our results of the open-loop buck converter when the resistor load is 100Ω. Figure 14 displays the open-loop buck converter output when the resistor load is 100Ω. Initially, there is no output voltage since the duty ratio starts at zero. We see the output voltage rise to 2.65V as soon as the duty ratio reaches 0.2. We also note that as the duty ratio increases the output voltage also increases, eventually reaching 2.92V. We note that the output never reaches the goal. Analyzing the initial response of the output voltage in figure 14a reveals a rise time of 2*ms*, a settling time of 11*ms*, a steady-state error of 0.35V, and no percent overshoot.

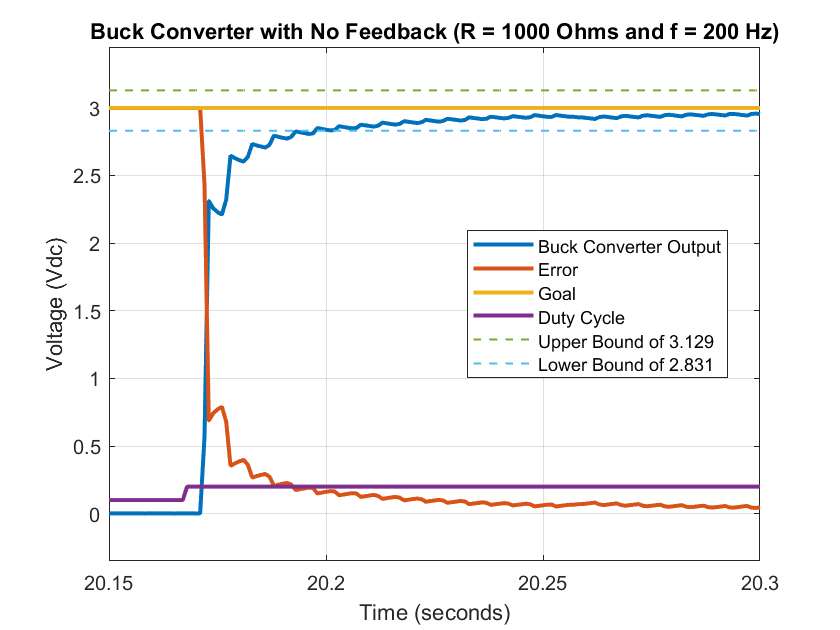
Overall, we believe the open-loop buck converter was a success since we were able to implement the same buck converter as before, but this time with a user controlled duty ratio. We saw that for a duty ratio less than 0.2 the buck converter would not turn on and it required our intervention in order for it to work. Finally, we were able to track the goal with a maximum steady-state error of 0.35V which represents an accuracy of 88% with respect to the goal.



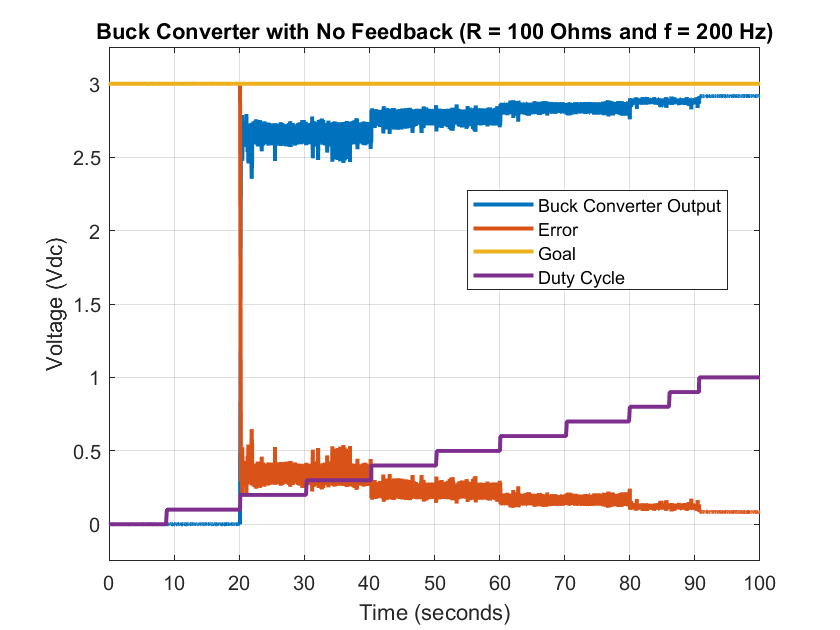
*Figure 12: Simulink model of the buck converter with no feedback. This model calculated the difference between the goal voltage and the output voltage to generate the error. The error signal was then passed through an absolute value block to ensure the error was never below zero. This is the key difference between the open-loop versus the closed-loop buck converter. The error in this circuit is not used to calculate anything, but instead simply to record and analyze how far the output voltage is from the goal. The PWM signal was generated through a PWM generator block in Simulink whose single input was the duty ratio. In addition, one of the parameters of the block was the frequency which we set to 200Hz. The duty ratio was manually changed for the different variations of the circuit tested. The PWM signal was then multiplied by 6.71 in order to properly drive the buck converter MOSFET. This approach relied on user input heavily since we had to manually change the duty ratio to obtain a satisfactory output voltage.*



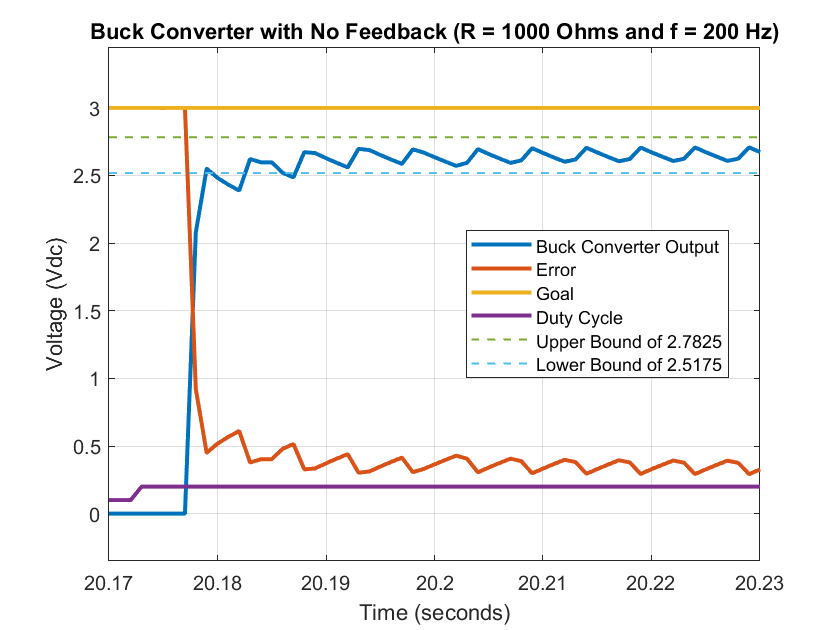
*Figure 13: This figure shows the open-loop buck converter output when the resistor load is 1000Ω and f = 200Hz. In this scenario the input voltage is always connected, but we see that the output voltage is zero for the first 20 seconds. This is because the duty ratio is initially set to zero. We manually increased the duty ratio by 0.1 roughly every 10 seconds. When the duty ratio reaches 0.2 the converter is finally triggered and we see an output voltage. We see that the output voltage is 2.98V when the duty ratio is 0.2. However, as the duty ratio increases so does the output voltage. When the duty ratio is 1.0 the output voltage is 3.21V. This means the minimum steady-state error is 0.02V, but the maximum steady-state error is 0.21V. It is important to note that we were manually adjusting the system to reach the desired goal.*

**

*Figure 13a: This is a zoomed-in version of figure 13 which focuses on the initial response of the output voltage when the duty ratio reaches 0.2. We note that the output voltage has a rise time of 12ms, a settling time of 32ms, a steady-state error of 0.02V, and no percent overshoot.*



*Figure 14: This figure shows the open-loop buck converter output when the resistor load is 100Ω and f = 200Hz. In this scenario the input voltage is always connected, but we see that the output voltage is zero for the first 20 seconds. This is because the duty ratio is initially set to zero. We manually increased the duty ratio by 0.1 roughly every 10 seconds. When the duty ratio reaches 0.2 the converter is finally triggered and we see an output voltage. We see that the output voltage is 2.65V when the duty ratio is 0.2. However, as the duty ratio increases so does the output voltage. When the duty ratio is 1.0 the output voltage is 2.92V. This means the minimum steady-state error is 0.08V, but the maximum steady-state error is 0.35V. It is important to note that we were manually adjusting the system to reach the desired goal.*



*Figure 14a: This is a zoomed-in version of figure 14 which focuses on the initial response of the output voltage when the duty ratio reaches 0.2. We note that the output voltage has a rise time of 2ms, a settling time of 11ms, a steady-state error of 0.35V, and no percent overshoot.*

**Conclusion**

The original goal of our project involved maintaining a constant output voltage in the face of a changing resistor load for a buck converter operating in discontinuous mode. This goal was somewhat achieved albeit with some important limitations. These limitations involved the range of voltages we obtained for both types of buck converters: closed-loop and open-loop. We expected to see a much wider range of voltages, but we believe the frequency we chose affected this significantly. From simulations we performed we saw that the output voltage of a buck converter was highly dependent on the duty ratio of the PWM signal. However, this is not what we observed in both types of converters implemented. We believe that this is due to the fact that buck converters operate at extremely high frequencies while we used a frequency of 200*Hz*. This frequency was chosen due to hardware limitations in the Quanser Q8-USB Data Acquisition Device. Even so we were still able to stabilize the output voltage of a closed-loop buck converter significantly. When the resistor load was 1000Ω the maximum output voltage recorded was 3.21V and when the resistor load was 100Ω the maximum was 2.86V. We believe this was a good performance from the controller since the two variations are off from the goal voltage by 0.21V and 0.14V, respectively.

In addition, after implementing the open-loop buck converter we realized that the implementation of the closed-loop buck converter was a success in and of itself. This is because in the open-loop converter we saw the output voltage change as we manually changed the duty ratio of the PWM signal. This meant that to make the open-loop buck converter work we had to manually adjust the duty ratio multiple times. On the other hand, with the closed-loop buck converter no user input was necessary. This meant that we had successfully implemented a closed-loop buck converter which achieved similar performance to the open-loop buck converter. This is significant because with the open-loop converter we had to manually adjust until we reached the best results. Overall, this project helped us learn an immense amount on buck converters, pulse width modulation, and the difference between simulations and real-life circuits.