

**Closed Loop Boost Converter using PI Controller Simulink Model**

Instructor: Dr.-Ing. Kashif Hussain

**Name: Wakeel Ahmed**

**CMS ID: 033-19-0026**

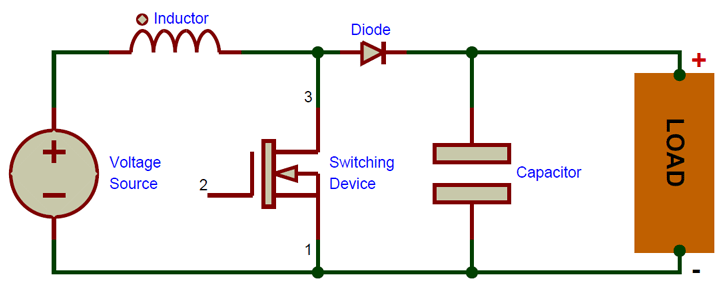
Department of Electrical Engineering

ELE-321: Control Systems (Spring 2023)

**Introduction:** The boost converter is a DC-DC power converter that steps up the input voltage to a higher output voltage. It is widely used in various applications, such as power supplies, battery charging systems, and renewable energy systems. In this report, we will design and simulate a boost converter using MATLAB Simulink.

**Design of Boost Converter:**

1. **Circuit Diagram:** The boost converter consists of an input voltage source, a power switch (MOSFET), an inductor, a diode, and an output capacitor. The circuit diagram is as follows:



1. **Control Strategy:** We will add a PI controller to regulate the duty cycle of the PWM signal. The PI controller takes the error between the desired output voltage and the actual output voltage as input and generates a control signal to adjust the duty cycle accordingly.

**MATLAB Simulink Implementation:**

1. **Simulink Model**

To implement a PI-controlled boost converter in MATLAB Simulink, you would typically require the following blocks:

* **Pulse Generator:** This block generates a periodic square wave signal that acts as the control signal for the power switch. You can set the pulse width or duty cycle based on the desired control input.
* **Switch:** The switch block represents the power switch in the boost converter circuit. It is controlled by the output of the PI controller. When the control signal is high, the switch is closed, allowing current flow through the inductor. When the control signal is low, the switch is open, causing the inductor to release its stored energy.
* **Inductor:** The inductor block models the behavior of the boost converter inductor. It stores energy when the switch is closed and releases it when the switch is open. You need to set the inductance value based on your specific design.
* **Capacitor:** The capacitor block represents the output capacitor of the boost converter. It helps smooth out the output voltage ripple. The value of the capacitor can be set based on your design requirements.
* **Gain Block:** The gain block amplifies or attenuates signals. You can use a gain block to scale the output of the PI controller to match the desired range for the duty cycle or control input.
* **PI Controller:** The PI controller block implements the Proportional-Integral-Derivative control algorithm. You can set the P, I, and D gains of the controller based on your tuning parameters. The PI controller takes the error signal (difference between the desired and actual output voltage) as input and generates a control signal based on the proportional, integral, and derivative terms.
* **Scope:** The scope block is used for visualization and monitoring of signals. You can use it to observe the output voltage, control signal, and other relevant signals during simulation.

Connect the blocks as per the circuit diagram shown below.

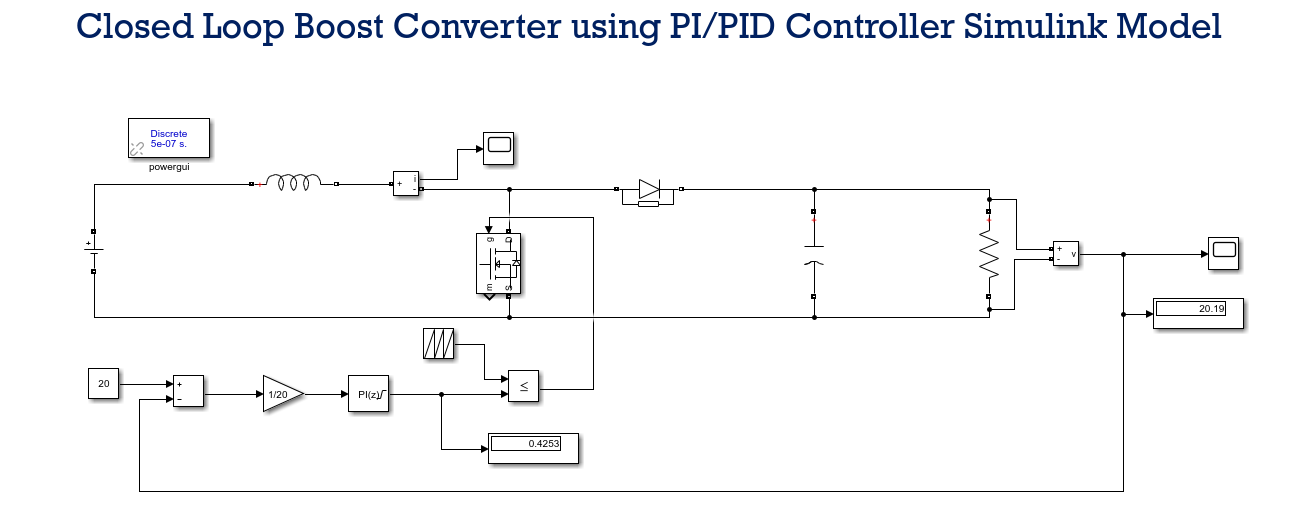


Figure 1: This is my Matlab circuit.

1. **Parameter Settings**

Set the parameters for the boost converter components:

* **Inductor (L):** Set the inductance value.
* **Capacitor (C):** Set the capacitance value.
* **Switching Frequency:** Set the frequency at which the MOSFET switches.

NOW Here are my all values of the design.

**Given:**

Vin = 12V (Input voltage)

Vout = 20V (Output voltage)

f = 25 kHz (Switching frequency)

**Calculate the duty cycle (D):**

D = Vout / (Vout + Vin)

D = 20V / (20V + 12V)

D = 20V / 32V

**D ≈ 0.625**

**Determine the inductor value (L):**

Choose an inductor ripple current (ΔI\_L) based on your requirements. Let's assume **ΔI\_L = 0.2A.**

L = (Vin \* (1 - D)) / (f \* ΔI\_L)

L = (12V \* (1 - 0.625)) / (25 kHz \* 0.2A)

L = 3.6 μH

**Calculate the capacitor value (C):**

Assuming a maximum output voltage ripple (ΔV) of 0.1V (for example):

C = (Iout \* (1 - D)) / (8 \* f \* ΔV)

**Let's assume Iout = 1A.**

C = (1A \* (1 - 0.625)) / (8 \* 25 kHz \* 0.1V)

**C ≈ 0.2 μF**

|  |  |
| --- | --- |
| **Parameter** | **Value** |
| Duty cycle | 0.625 |
| Iout | 1 A |
| ΔI\_L | 0.2 A |
| ΔV | 0.1 v |
| L | 3.6 uH |
| C | 0.2 uF |
| RL | 4 Ohm |

1. **PI Controller Settings**

To determine the P, I values for the PI controller in MATLAB Simulink, you can use

**Ziegler-Nichols Method:**

This is a popular tuning method that involves the following steps:

* Set all the gains (P, I) to zero.
* Increase the proportional gain (P) until the system starts to oscillate with a sustained and constant amplitude.
* Measure the oscillation period (Tu) and calculate the ultimate gain (Ku) during this sustained oscillation.

Use the following formulas to determine the PI gains:

**P = 0.6 \* Ku**

**I = 1.2 \* Ku / Tu**

**D = 0.075 \* Ku \* Tu**

Adjust the gains based on these values and fine-tune them if needed.

**Simulation and Analysis:**

1. **Efficiency Calculation:**

Calculate the efficiency of the boost converter by measuring the power input and power output. Efficiency can be calculated as:

Efficiency = (Power output)/ (Power Input) \* 100%

1. **Performance Evaluation:** Evaluate the performance of the boost converter with the PI controller by analyzing the settling time, overshoot, and steady-state error. Adjust the PI controller gains to optimize these performance metrics.

**Performance Optimization:**

1. **Parameter Adjustment:** Explore the effect of changing the inductance, capacitance, and switching frequency on the performance of the boost converter. Optimize these parameters to achieve the desired output voltage with high efficiency.
2. **Feedback Control:** Implement a feedback control mechanism to maintain a constant output voltage despite variations in the input voltage or load conditions.

Finally, My Matlab overall design with PI controller and its output results are shown below:

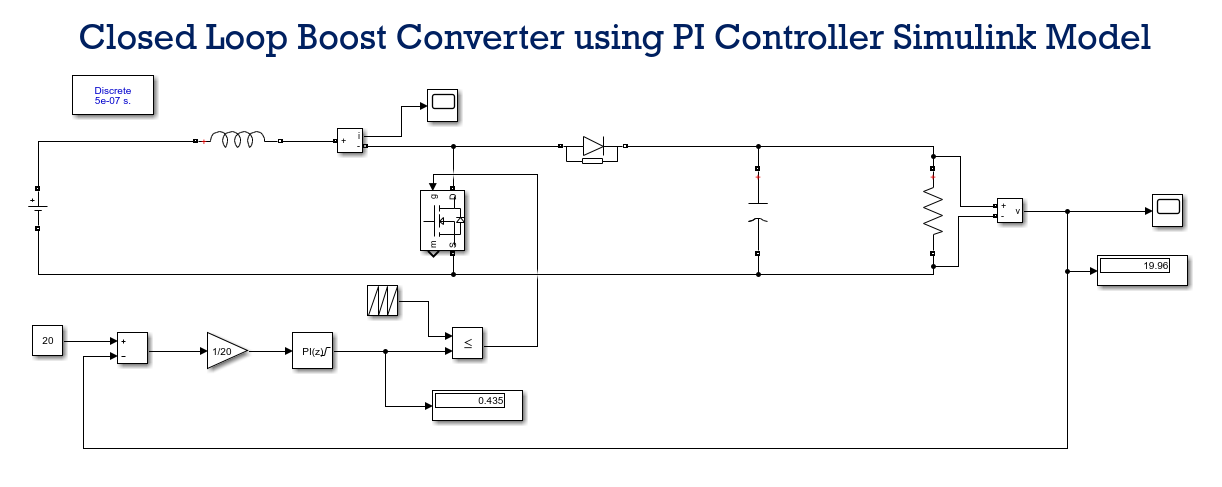


Figure 2: Circuit diagram

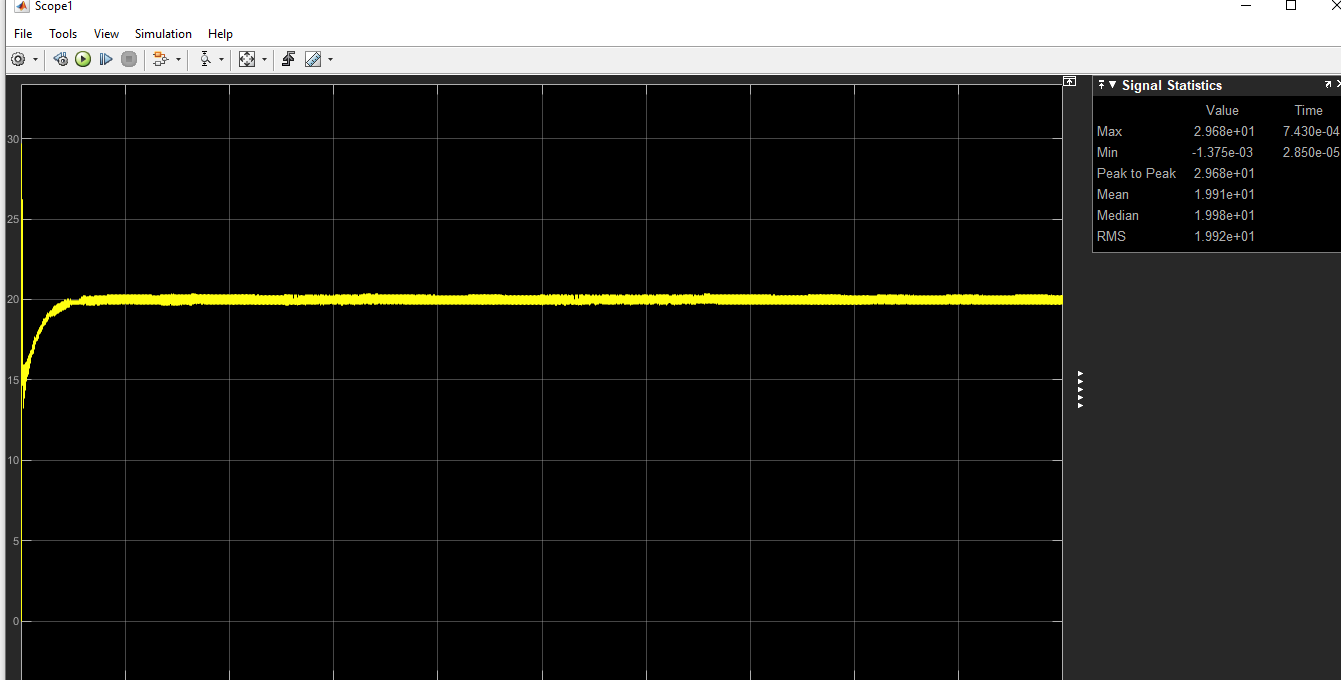


Figure Output response.

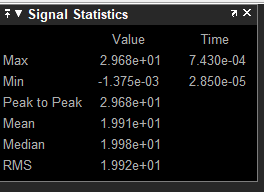


Figure Output values.

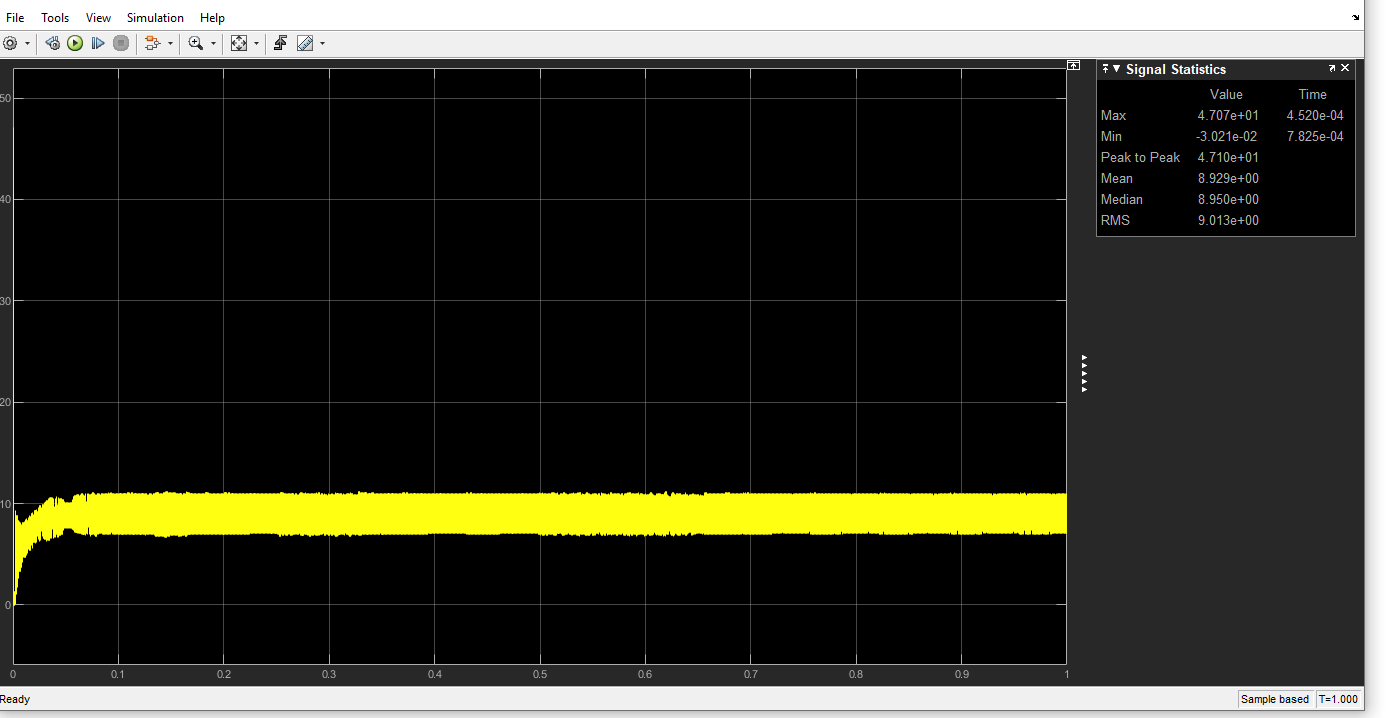


Figure current waveform.

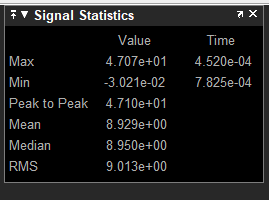


Figure Current values.

**Conclusion:**

In this report, we have designed and simulated a PI controller boost converter using MATLAB Simulink. The boost converter is a critical component in many power electronics applications, and its performance can be optimized through parameter adjustments and feedback control. Simulink provides a powerful platform for modeling and simulating such systems, enabling us to analyze their behavior and make informed design decisions.

A PI controller into the boost converter control system to improve the voltage regulation. By adjusting the PI controller gains and observing the simulation results, you can achieve better output voltage control, reduced settling time, and improved stability.

With the addition of the PI controller, the boost converter becomes more robust and capable of maintaining a stable output voltage even under varying load conditions or input voltage fluctuations.