

UNIVERSIDADE DO ESTADO DE MATO GROSSO

ELETRÔNICA DE POTÊNCIA

**RELATÓRIO TÉCNICO
CONVERSOR BUCK E CONVERSOR BOOST**

Aluna: Maria da Conceição Braga da Silva
Professor avaliador: Marcus Felipy Glatz Rodrigues

2025

Conversores CC-CC - BUCK E BOOST

Em sistemas eletrônicos e industriais modernos, é comum trabalhar com diferentes níveis de tensão, cada um adequado a um tipo específico de equipamento. Entretanto, a alimentação elétrica disponível nem sempre corresponde exatamente ao que cada dispositivo necessita. Por isso, conversores CC-CC se tornam importantes para adaptar tensões contínuas de maneira eficiente e segura.

Dois dos conversores mais utilizados são o **Buck** e o **Boost**.

- Buck - Reduz a tensão;
- Boost - Eleva a tensão.

Embora tenham finalidades opostas, ambos operam a partir do mesmo princípio básico: uma chave eletrônica comutando em determinada frequência e um filtro que suaviza essa comutação, produzindo uma tensão contínua ajustável na saída. Essa abordagem, apresentada de maneira clara por Hart (2012), serve como base para diversas aplicações da eletrônica de potência.

1. Conversor Buck (Abaixador)

O conversor Buck, também conhecido como conversor abaixador de tensão, é uma das topologias mais utilizadas em eletrônica de potência para reduzir uma tensão contínua de entrada para um valor inferior na saída (Hart, 2011). Em muitos cenários práticos, especialmente em ambientes de automação e monitoramento, existe a necessidade de converter tensões de nível industrial para valores compatíveis com dispositivos eletrônicos mais sensíveis. Um exemplo disso aparece no sistema de vigilância, onde as câmeras de segurança e alguns equipamentos auxiliares, como roteadores e DVRs, operam tipicamente a 12 V, enquanto o barramento principal de alimentação no painel elétrico industrial fornece 24 V. Assim, o conversor Buck torna-se uma ferramenta-chave para reduzir essa tensão de maneira eficiente, segura e estável.

O funcionamento do conversor baseia-se no chaveamento rápido de um interruptor eletrônico, geralmente um MOSFET, controlado por um sinal PWM (*Pulse Width Modulation*), traduzido, significa modulação por largura de pulso. Esse chaveamento alterna entre dois estados principais: o período em que a chave está ligada (Primeiro estágio) e o período em que está desligada (Segundo estágio). Quando a chave é acionada, a tensão de entrada é aplicada diretamente sobre o

indutor e sobre o restante do circuito, permitindo que o indutor armazene energia e aumentando a corrente por meio da relação $VL = L * di/dt$.

Durante esse intervalo, o diodo permanece reversamente polarizado. Na segunda etapa, quando a chave desliga, o indutor continua conduzindo devido à sua propriedade de resistir à variação instantânea de corrente. Neste momento, o diodo passa a conduzir e o indutor libera energia para a carga, mantendo o fluxo de corrente constante. O capacitor na saída atua como elemento de filtragem, reduzindo a ondulação (ripple) e estabilizando a tensão entregue à carga.

Quando o conversor opera em regime permanente e o filtro LC é considerado ideal, a relação entre a tensão de saída e a tensão de entrada é proporcional ao ciclo de trabalho do PWM. Essa relação pode ser expressa por:

$$V_{out} = D * V_{in}$$

Essa expressão mostra que o valor médio da tensão de saída depende diretamente da fração do período em que a chave está ligada. No caso deste projeto, adotou-se uma tensão de entrada de 24 V e um duty cycle de 50%, o que conduz a uma tensão ideal de saída de 12 V, valor compatível com dispositivos de monitoramento por câmera que normalmente utilizam essa alimentação.

Dentro de um painel industrial, a alimentação geral costuma ser 24 VCC, por ser segura, padronizada e robusta contra interferências. Entretanto, muitos dispositivos auxiliares (principalmente os voltados para vigilância e comunicação) operam em 12 VCC. Foi exatamente esse cenário que motivou o desenvolvimento do projeto: converter 24 V do barramento principal para 12 V, com estabilidade e eficiência.

Para analisar o comportamento do conversor, foram realizadas duas simulações distintas. A primeira delas considera apenas um período de chaveamento, permitindo visualizar detalhadamente o comportamento interno do circuito em alta frequência. Nesse caso, observa-se o estado da chave, a tensão no indutor e a corrente no indutor, formando as ondas triangulares características desse tipo de conversor. Essa análise é importante porque evidencia o papel do indutor como elemento responsável por suavizar a corrente ao longo dos intervalos ON (Ligado) e OFF (Desligado), confirmando o modelo teórico fundamental.

```

# PARTE 1 - SIMULAÇÃO DE UM ÚNICO PERÍODO (modo ideal)

dt = T / 1000.0                                # passo de tempo

t = numpy.arange(0.0, T, dt)

N = len(t)

iL = numpy.zeros(N)                             # corrente no indutor

vL = numpy.zeros(N)                             # tensão no indutor

s = numpy.zeros(N)                             # estado da chave

IL = 0.0                                       # corrente inicial

for n in range(1, N):

    if t[n] < D * T:

        # Chave ON (MOSFET ligado)

        vL[n] = Vin - Vout_ideal                 # tensão no indutor

        dIL = vL[n] / L                         # di/dt

        s[n] = 1.0

    else:

        # Chave OFF (diodo conduz)

        vL[n] = -Vout_ideal

        dIL = vL[n] / L

        s[n] = 0.0

    IL += dIL * dt                           # integração da corrente

    iL[n] = IL

```

A segunda simulação observa o comportamento da corrente e da tensão ao longo de vários períodos consecutivos, permitindo analisar o processo de carregamento do capacitor e a dinâmica da tensão de saída até atingir o regime permanente. No início, a tensão de saída parte de zero, já que o capacitor se encontra descarregado. Com o passar dos ciclos, a energia fornecida pelo indutor passa a carregar o capacitor, elevando gradualmente a tensão de saída até estabilizar-se no valor ideal de 12 V. Ao mesmo tempo, a corrente no indutor passa por um comportamento transitório até encontrar sua forma periódica típica. Esse tipo

de simulação é importante para avaliar a resposta real do conversor, pois mostra como a energia flui e se distribui no sistema até que se estabeleça o equilíbrio entre a energia fornecida pela fonte e a energia consumida pela carga.

```
# PARTE 2 - SIMULAÇÃO LONGA (vários períodos, ainda ideal)

t_total = 5e-3                                # simular 5 ms

dt2 = T / 200.0

t2 = numpy.arange(0.0, t_total, dt2)

N2 = len(t2)

iL2 = numpy.zeros(N2)                            # corrente no indutor

vC2 = numpy.zeros(N2)                            # tensão no capacitor (saída)

IL = 0.0

Vc = 0.0

passos_por_período = int(T / dt2)

for n in range(1, N2):

    posicao = n % passos_por_período

    # PWM ideal

    if posicao < D * passos_por_período:

        # ON: Vin aplicado ao indutor + carga

        vL_temp = Vin - Vc

    else:

        # OFF: indutor alimenta a carga

        vL_temp = -Vc

    # Atualiza corrente do indutor

    dIL = vL_temp / L

    IL += dIL * dt2

    # indutor não deixa a corrente inverter o sentido

    if IL < 0.0:

        IL = 0.0

    iL2[n] = IL

    # Corrente na carga
```

```

I_carga = Vc / R
# Corrente no capacitor

I_cap = IL - I_carga

# Atualiza tensão no capacitor

dVc = I_cap / C

Vc += dVc * dt2

vC2[n] = Vc

```

A escolha dos componentes também segue a abordagem ideal do conversor, considerando MOSFET, diodo, indutor e capacitor perfeitos, sem perdas internas. Esse modelo simplificado permite compreender facilmente a operação básica do Buck sem complicações relacionadas a resistências parasitas ou quedas de tensão. A intenção é oferecer uma visualização clara dos princípios fundamentais que regem o funcionamento da topologia e validar a equação central do conversor.

Assim, tanto pela teoria quanto pelos resultados simulados, observa-se que o conversor Buck apresenta comportamento coerente: reduz de 24 V para 12 V com eficiência ideal e responde de acordo com as previsões da literatura. Isso confirma que a escolha desse conversor é adequada para aplicações como o fornecimento de energia estável a sistemas de monitoramento por câmeras em painéis industriais, garantindo operação segura e confiável desses equipamentos.

Código completo:

```

# Projeto Final - Parte 1 - Eletrônica de Potência

# Conversor BUCK (Abaixador de Tensão)

# Autora: Maria da Conceição Braga da Silva

# objetivo: reduzir de 24V para 12V - Sistema de Monitoramento

import numpy

import matplotlib.pyplot

# Parâmetros do conversor buck ideal

Vin = 24.0          # Tensão de entrada (V)

D = 0.5              # Duty cycle (50%) → Vout esperado = 12 V

R = 10.0             # Resistência de carga (ohms)

```

```

L = 330e-6          # Indutor (H)
C = 100e-6          # Capacitor (F)
f = 50e3             # Frequência de chaveamento (Hz)
T = 1.0 / f          # Período (s)

Vout_ideal = Vin * D  # Tensão de saída ideal (12 V)

# PARTE 1 - SIMULAÇÃO DE UM ÚNICO PERÍODO (modo ideal)

dt = T / 1000.0       # passo de tempo

t = numpy.arange(0.0, T, dt)

N = len(t)

iL = numpy.zeros(N)      # corrente no indutor
vL = numpy.zeros(N)      # tensão no indutor
s = numpy.zeros(N)        # estado da chave
IL = 0.0                 # corrente inicial

for n in range(1, N):
    if t[n] < D * T:
        # Chave ON (MOSFET ligado)
        vL[n] = Vin - Vout_ideal      # tensão no indutor
        dIL = vL[n] / L              # di/dt
        s[n] = 1.0
    else:
        # Chave OFF (diodo conduz)
        vL[n] = -Vout_ideal
        dIL = vL[n] / L
        s[n] = 0.0
    IL += dIL * dt                # integração da corrente
    iL[n] = IL

# PARTE 2 - SIMULAÇÃO LONGA (vários períodos, ainda ideal)

t_total = 5e-3           # simular 5 ms
dt2 = T / 200.0
t2 = numpy.arange(0.0, t_total, dt2)

```

```

N2 = len(t2)

iL2 = numpy.zeros(N2) # corrente no indutor

vC2 = numpy.zeros(N2) # tensão no capacitor (saída)

IL = 0.0

Vc = 0.0

passos_por_período = int(T / dt2)

for n in range(1, N2):

    posicao = n % passos_por_período

    # PWM ideal

    if posicao < D * passos_por_período:

        # ON: Vin aplicado ao indutor + carga

        vL_temp = Vin - Vc

    else:

        # OFF: indutor alimenta a carga

        vL_temp = -Vc

    # Atualiza corrente do indutor

    dIL = vL_temp / L

    IL += dIL * dt2

    # indutor não deixa a corrente inverter o sentido

    if IL < 0.0:

        IL = 0.0

    iL2[n] = IL

    # Corrente na carga

    I_carga = Vc / R

    # Corrente no capacitor

    I_cap = IL - I_carga

    # Atualiza tensão no capacitor

    dVc = I_cap / C

    Vc += dVc * dt2

    vC2[n] = Vc

```

```

# FIGURA 1 - UM PERÍODO (formas de onda fundamentais do Buck)

fig1, ax = matplotlib.pyplot.subplots(3, 1, figsize=(9, 6))

ax[0].plot(t * 1e6, s)

ax[0].set_title("Estado da Chave (um período)")

ax[0].set_ylabel("S")

ax[0].grid(True)

ax[1].plot(t * 1e6, vL)

ax[1].set_title("Tensão no Indutor vL (ideal)")

ax[1].set_ylabel("vL (V)")

ax[1].grid(True)

ax[2].plot(t * 1e6, iL)

ax[2].set_title("Corrente no Indutor iL (ideal)")

ax[2].set_ylabel("iL (A)")

ax[2].set_xlabel("Tempo (μs)")

ax[2].grid(True)

matplotlib.pyplot.tight_layout()

# FIGURA 2 - VÁRIOS CICLOS + CURVA Vout × D (ideal)

fig2, ax2 = matplotlib.pyplot.subplots(3, 1, figsize=(9, 6))

# Corrente no indutor em vários ciclos

ax2[0].plot(t2 * 1e3, iL2)

ax2[0].set_title("Corrente no Indutor (vários ciclos, ideal)")

ax2[0].set_ylabel("iL (A)")

ax2[0].grid(True)

# Tensão de saída ao longo do tempo

ax2[1].plot(t2 * 1e3, vC2)

ax2[1].set_title("Tensão de Saída Vout - Regime Ideal")

ax2[1].set_ylabel("Vout (V)")

```

```

ax2[1].set_xlabel("Tempo (ms)")

ax2[1].legend()

ax2[1].grid(True)

# Curva teórica Vout x Duty (Buck ideal)

Dcurve = numpy.linspace(0.0, 1.0, 500)

Vcurve = Dcurve * Vin

ax2[2].plot(Dcurve, Vcurve)

ax2[2].set_title("Relação Teórica Vout x Duty Cycle")

ax2[2].set_xlabel("Duty Cycle D")

ax2[2].set_ylabel("Vout (V)")

ax2[2].grid(True)

matplotlib.pyplot.tight_layout()

matplotlib.pyplot.show()

```

Resultados Obtidos:

Figure 1

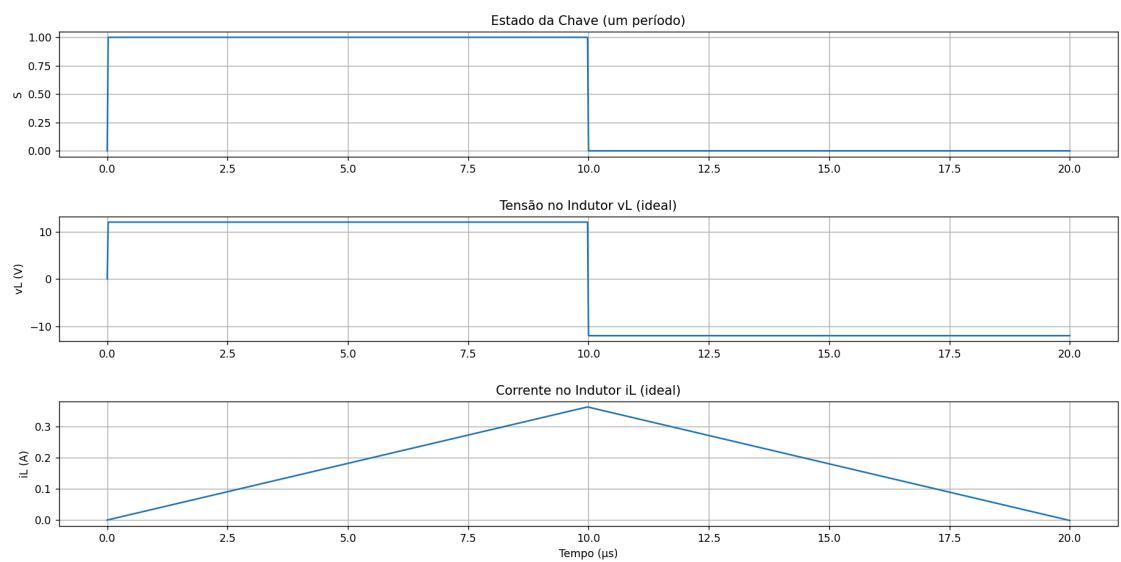
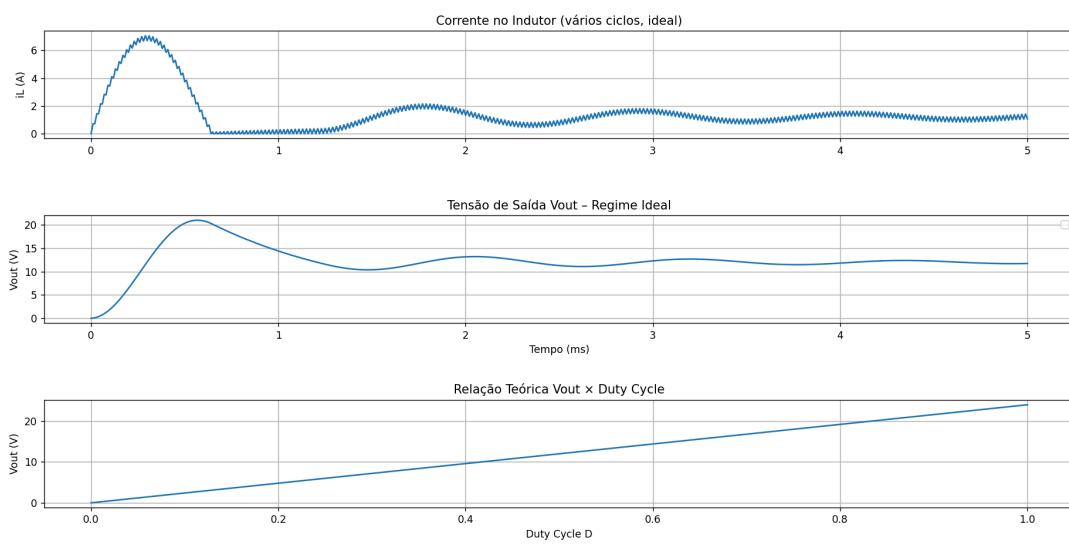


Figure 2



2. Conversor Boost (Elevador)

O conversor Boost é uma topologia de eletrônica de potência utilizada para elevar uma tensão contínua de entrada para um valor mais alto na saída. Diferentemente do conversor Buck, que reduz a tensão, o Boost tem como função principal aumentar o nível de tensão disponível, aproveitando o princípio de armazenamento de energia no indutor. Esse tipo de conversor é largamente empregado em dispositivos alimentados por bateria, nos quais a tensão nominal do elemento de armazenamento não é suficiente para atender diretamente à carga. Um exemplo bastante comum é o de carregadores portáteis (power banks), que utilizam baterias de íon-lítio com tensão típica de 3,7 V e precisam fornecer 5 V na saída para alimentar dispositivos conectados à porta USB, como celulares, fones de ouvido e pequenos equipamentos eletrônicos.

Nesse contexto, o conversor Boost modelado neste trabalho representa justamente essa aplicação prática: elevar a tensão de uma bateria de 3,7 V para aproximadamente 5 V. A escolha inicial da relação de conversão leva em conta a expressão ideal do conversor, na qual a tensão de saída está relacionada à tensão de entrada e ao duty cycle D pela relação $V_{out} = V_{in} / (1 - V_{out})$. A partir dessa expressão, considerando $V_{in} = 3,7\text{ V}$ e $V_{out} = 5\text{ V}$, obtém-se um duty cycle em torno de 26%. Esse valor é adotado na simulação para representar o comportamento ideal da topologia, sem controle de malha ou compensações adicionais.

O circuito básico do Boost é composto por uma fonte de entrada, um indutor em série, uma chave eletrônica, um diodo, um capacitor de saída e a carga. O funcionamento pode ser dividido em dois intervalos principais. No período em que a chave está ligada (intervalo ON), o indutor é conectado diretamente à fonte e armazena energia sob a forma de campo magnético, enquanto o diodo encontra-se reversamente polarizado e a carga é alimentada pelo capacitor. Nesse instante, a tensão sobre o indutor é aproximadamente igual à tensão de entrada e a corrente no indutor cresce de maneira aproximadamente linear. No segundo intervalo, quando a chave é desligada (intervalo OFF), o campo magnético do indutor começa a se desfazer; para manter a corrente contínua, o indutor força a condução do diodo e passa a transferir energia para o capacitor e a carga. Nesse momento, a tensão na saída resulta da soma da tensão da fonte com a tensão induzida no indutor, o que permite que a tensão de saída seja maior que a tensão de entrada.

Na simulação desenvolvida, considerou-se um modelo ideal, em que componentes como indutor, diodo, MOSFET e capacitor não apresentam perdas, resistências parasitas ou quedas de tensão. Os parâmetros adotados foram: tensão de entrada de 3,7 V, duty cycle de 26%, carga resistiva de $5\ \Omega$ (equivalente a uma corrente de 1 A em 5 V), indutor de $100\ \mu\text{H}$, capacitor de $100\ \mu\text{F}$ e frequência de chaveamento de 100 kHz. Esses valores representam uma situação compatível com conversores de pequeno porte utilizados em power banks, em que se busca um compromisso entre tamanho dos componentes, ondulação de corrente, ondulação de tensão e facilidade de implementação.

Assim como no estudo do conversor Buck, a análise numérica foi dividida em duas etapas. Na primeira etapa, é simulado apenas um período de chaveamento. Esse recorte de tempo permite observar com clareza o comportamento da chave, da tensão no indutor e da corrente no indutor dentro de um ciclo PWM. Durante o intervalo ON, o gráfico mostra a chave em nível alto, a tensão no indutor igual à tensão da bateria e a corrente do indutor aumentando. Na transição para o intervalo OFF, a chave vai a zero, a tensão no indutor torna-se negativa (aproximadamente $V_{in} - V_{out}$) e a corrente do indutor passa a decrescer. O resultado é uma corrente de forma triangular, característica dos conversores em modo contínuo e em regime ideal.

```

# PARTE 1: um período

dt = T / 1000

t = numpy.arange(0, T, dt)

iL = numpy.zeros(len(t))

vL = numpy.zeros(len(t))

s = numpy.zeros(len(t))

IL = 0

Vout = Vin / (1 - D)

for n in range(1, len(t)):

    if t[n] < D * T:

        vL[n] = Vin

        dIL = vL[n] / L

        s[n] = 1

    else:

        vL[n] = Vin - Vout

        dIL = vL[n] / L

        s[n] = 0

    IL += dIL * dt

    iL[n] = IL

```

Na segunda etapa, a simulação considera vários ciclos de chaveamento ao longo de alguns milissegundos, permitindo acompanhar a evolução da corrente no indutor e da tensão no capacitor de saída desde o instante em que o conversor é ligado até o estabelecimento do regime permanente. Inicialmente, o capacitor encontra-se descarregado, de modo que a tensão de saída é nula. À medida que o conversor opera, o indutor começa a transferir energia para a saída e o capacitor vai se carregando, fazendo com que a tensão de saída cresça rapidamente. Esse processo provoca oscilações transitórias tanto na corrente do indutor quanto na tensão de saída. O gráfico da tensão mostra um pico inicial que ultrapassa o valor desejado, seguido de oscilações que vão se amortecendo até que o sistema se estabilize em torno de 5 V. A corrente no indutor também apresenta um

comportamento transitório, com amplitudes maiores no início, diminuindo gradualmente até atingir uma forma de onda periódica mais limitada e compatível com o regime estacionário.

```
# PARTE 2: vários ciclos

t_total = 5e-3

dt2 = T / 200

t2 = numpy.arange(0, t_total, dt2)

iL2 = numpy.zeros(len(t2))

vC2 = numpy.zeros(len(t2))

IL = 0

Vc = 0

for n in range(1, len(t2)):

    if (n % int(T/dt2)) < int(D*(T/dt2)):

        vL_inst = Vin

    else:

        vL_inst = Vin - Vc

    # corrente do indutor

    dIL = vL_inst / L

    IL += dIL * dt2

    iL2[n] = IL

    # capacitor

    Iload = Vc / R

    dVc = (IL - Iload) / C

    Vc += dVc * dt2

    vC2[n] = Vc
```

Além dos gráficos de formas de onda, foi traçada a curva teórica da relação entre V_{out} e o *duty cycle*, utilizando a equação ideal do conversor Boost. Essa curva mostra claramente que, conforme o *duty* se aproxima de 1, a tensão de saída cresce de forma acentuada, tendendo a valores muito elevados no modelo ideal. Isso ajuda a visualizar o compromisso existente entre ganho de tensão e esforço de comutação: para ganhos moderados, como o caso de 3,7 V para 5 V, o *duty cycle* necessário permanece relativamente baixo, o que facilita a operação prática do conversor.

Os resultados das simulações confirmam o comportamento previsto pela teoria. O conversor foi capaz de elevar a tensão da fonte de 3,7 V para um valor em torno de 5 V, após o período transitório, e as formas de onda de tensão e corrente observadas no indutor reproduzem o que é descrito nos modelos analíticos da literatura. Dessa forma, o estudo numérico do conversor Boost aplicado a um cenário de carregador portátil contribui para consolidar a compreensão da topologia e ilustra, de forma clara, como os conceitos de eletrônica de potência podem ser aplicados em dispositivos do cotidiano, como power banks utilizados para carregar equipamentos em portas USB.

Código Completo:

```
# Projeto Final - Parte 2 - Eletrônica de Potência
# Conversor BOOST (Elevador de Tensão) - Bateria Power Bank
# Autora: Maria da Conceição Braga da Silva
# objetivo: Aumentar tensão de 3,7 V para 5V

import numpy as numpy
import matplotlib.pyplot as pyplot

# Parâmetros

Vin = 3.7          # tensão da bateria
D = 0.26           # duty para elevar 3.7 para 5V
R = 5               # carga simulando 1A
L = 100e-6          # indutor típico de boost portátil
C = 100e-6          # capacitor de saída
```

```

f = 100e3           # frequência
T = 1 / f           # período

# PARTE 1: um período

dt = T / 1000

t = numpy.arange(0, T, dt)

iL = numpy.zeros(len(t))

vL = numpy.zeros(len(t))

s = numpy.zeros(len(t))

IL = 0

Vout = Vin / (1 - D)

for n in range(1, len(t)):

    if t[n] < D * T:

        vL[n] = Vin

        dIL = vL[n] / L

        s[n] = 1

    else:

        vL[n] = Vin - Vout

        dIL = vL[n] / L

        s[n] = 0

    IL += dIL * dt

    iL[n] = IL

# PARTE 2: vários ciclos

t_total = 5e-3

dt2 = T / 200

t2 = numpy.arange(0, t_total, dt2)

iL2 = numpy.zeros(len(t2))

vC2 = numpy.zeros(len(t2))

IL = 0

Vc = 0

```

```

for n in range(1, len(t2)):

    if (n % int(T/dt2)) < int(D*(T/dt2)):

        vL_inst = Vin

    else:

        vL_inst = Vin - Vc

    # corrente do indutor

    dIL = vL_inst / L

    IL += dIL * dt2

    iL2[n] = IL

    # capacitor

    Iload = Vc / R

    dVc = (IL - Iload) / C

    Vc += dVc * dt2

    vC2[n] = Vc


# FIGURA 1

fig1, ax = pyplot.subplots(3,1,figsize=(9,6))

ax[0].plot(t*1e6, s)

ax[0].set_title("Estado da Chave (Boost)")

ax[0].grid(True)


ax[1].plot(t*1e6, vL)

ax[1].set_title("Tensão no Indutor")

ax[1].grid(True)


ax[2].plot(t*1e6, iL)

ax[2].set_title("Corrente no Indutor")

ax[2].grid(True)

pyplot.tight_layout()

# FIGURA 2

```

```
fig2, ax2 = pyplot.subplots(3,1,figsize=(9,6))

ax2[0].plot(t2*1e3, iL2)

ax2[0].set_title("Corrente no Indutor (vários ciclos)")

ax2[0].grid(True)

ax2[1].plot(t2*1e3, vC2)

ax2[1].set_ylabel(Vout)

ax2[1].set_title("Tensão de Saída (Boost)")

ax2[1].grid(True)

ax2[1].legend()

# curva teórica

Dcurve = numpy.linspace(0.01, 0.99, 500)

Vcurve = Vin / (1 - Dcurve)

ax2[2].plot(Dcurve, Vcurve)

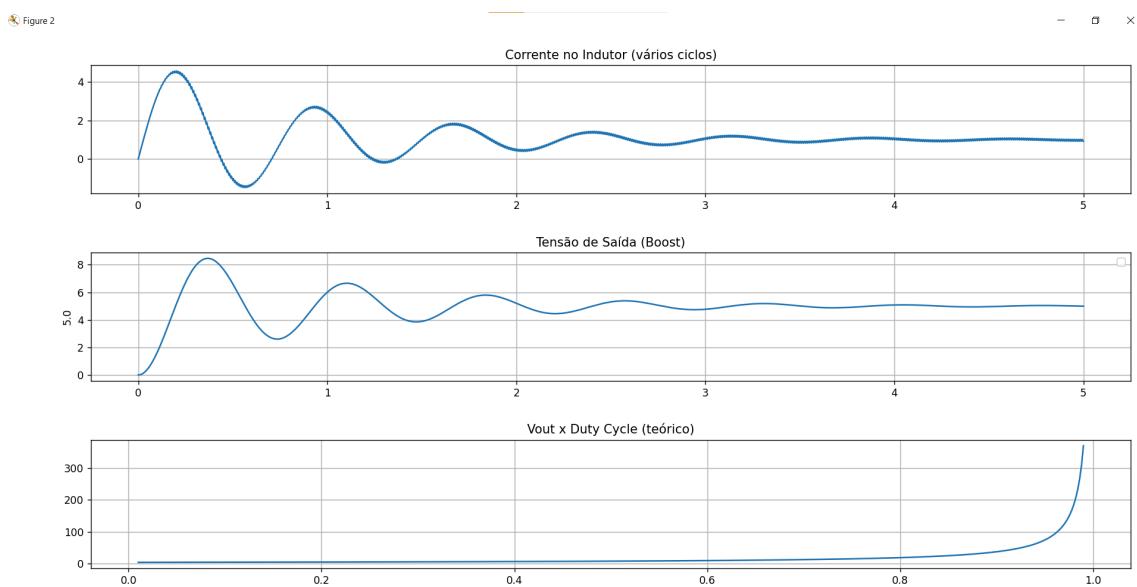
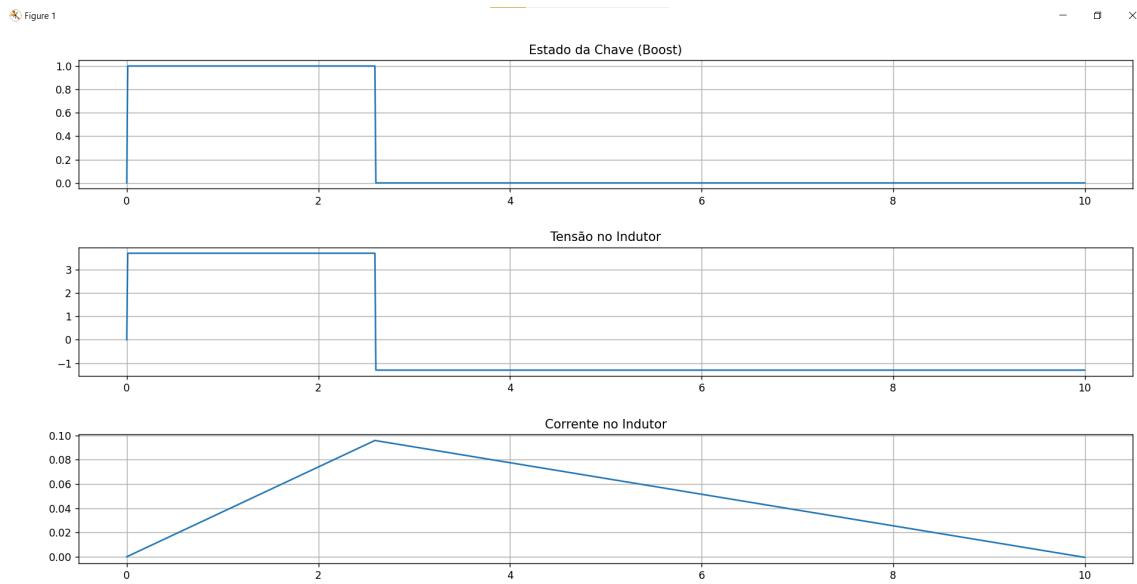
ax2[2].set_title("Vout x Duty Cycle (teórico)")

ax2[2].grid(True)

pyplot.tight_layout()

pyplot.show()
```

Resultados obtidos:



Referências

HART, Daniel W. Eletrônica de potência: análise e projetos de circuitos. Tradução: Romeu Albo; revisão técnica: Antonio Pertence Júnior. Porto Alegre: AMGH, 2012. 480 p.

GLATZ, Marcus. Eletrônica de Potência – Aula 05: Conversores Buck e Boost. Material de aula.pdf. FACET – Universidade Estadual de Mato Grosso (UNEMAT), Sinop, 2025.