

Documentação rRocket v1.5.6

4 de janeiro de 2023

1 Introdução

Esta é uma versão preliminar da documentação do rRocket que servirá de base para a elaboração do artigo. O texto da introdução deverá:

- Apresentar a necessidade de altímetros com capacidade de dois eventos de ejeção de paraquedas, contextualizando a disponibilidade deste tipo de equipamento no Brasil.
- Descrever a proposta de altímetro de projeto aberto (*open hardware* e *open software*) do rRocket.
- Alertar o usuário sobre a necessidade de aplicação de regras de segurança no lançamento de minifoguetes, destacando que nenhum sistema eletrônico é infalível e que, em caso de falha do rRocket, deve-se prever um local em que a queda do minifoguete não cause danos à vida e ao patrimônio. Isentar os autores de qualquer responsabilidade sobre acidentes que venham a ocorrer.

2 Modelagem

Para projetar o controle do sistema de recuperação, é necessário conhecer previamente o comportamento típico da trajetória de minifoguetes e as situações adversas que podem ocorrer.

2.1 Trajetória ideal

Em um cenário ideal, a componente vertical h da trajetória do minifoguete varia com o tempo t de acordo com a Fig. 1. Entre A e B, o minifoguete aguarda o lançamento em repouso. Em B ocorre a ignição do motor e o foguete inicia o movimento acelerado. Entre B e C ocorre a queima do propelente (fase propulsada). Entre C e E ocorre o movimento balístico, com apogeu no ponto D. No ponto E ocorre a implantação do paraquedas auxiliar. Entre E e F, o minifoguete cai com velocidade constante. No ponto F ocorre a implantação do paraquedas principal. Entre F e G, o minifoguete cai com velocidade constante, porém, menor do que no intervalo de E a F. No ponto G o minifoguete entra em repouso e permanece assim até o ponto H.

2.2 Adversidades

As principais adversidades relacionadas ao sistema de recuperação que devem ser levadas em conta no *software* de controle do altímetro, no manuseio do altímetro e na montagem do minifoguete são listadas abaixo:

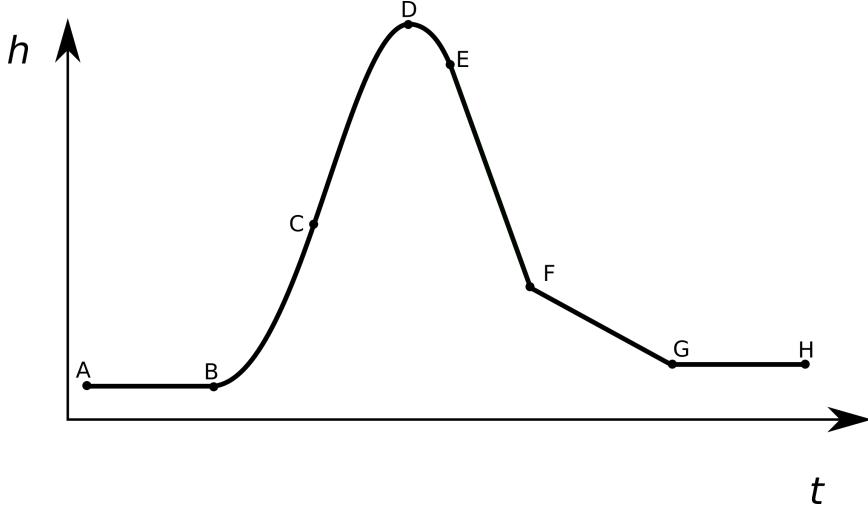


Figura 1: Trajetória típica em um voo de minifoguete com recuperação em dois estágios.

1. Os sistemas de recuperação baseados em barômetro relacionam a pressão e a temperatura ambiente à altitude. Para que h seja determinado corretamente a partir da altitude registrada pelo barômetro, é necessário garantir que a pressão e a temperatura ambientais sejam as mesmas no interior do minifoguete. Esta condição é satisfeita se houver furos na fuselagem do minifoguete com diâmetros adequados e em posições adequadas[1] que permitam a equalização da pressão interna à externa.
2. O sensor de pressão fornece h com incerteza $\pm U_h$. O ruído em $h(t)$ é particularmente perceptível entre A e B e entre G e H, como ilustrado na Fig 2a.
3. O sistema de recuperação é ligado em uma altitude e lançado em outra. Durante o transporte do local de montagem para o local de lançamento pode ocorrer a detecção de voo (Fig 2b).
4. O sistema de recuperação fica ligado por tempo suficientemente grande para que as condições atmosféricas mudem e, consequentemente, haja alteração no registro de h , causando detecção errada de voo (Fig 2c).
5. Durante o voo do minifoguete, é possível que ocorram oscilações na leitura da pressão devido à instabilidade na trajetória, vibrações, rajadas de vento laterais, que causam perturbações no escoamento. Aumentos de pressão também podem ocorrer quando o foguete ultrapassa a velocidade do som ou imediatamente após o lançamento devido à aceleração. As oscilações na pressão causam variações falsas em h , como ilustrado nos pontos P e Q da Fig. 2d, que podem indicar uma situação de queda e levar à implantação prematura das paraquedas.
6. Durante a montagem do minifoguete, com o sistema de recuperação ligado, é possível que variações de pressão causem detecção de voo (Fig 2e).
7. A altitude de pouso é geralmente diferente da altitude de decolagem.
8. Reinicialização do dispositivo em voo.
9. Falha de inicialização de componentes críticos (barômetro e acionador de paraquedas).
10. Falha de funcionamento de componentes críticos em voo.
11. Falha de implantação de paraquedas.

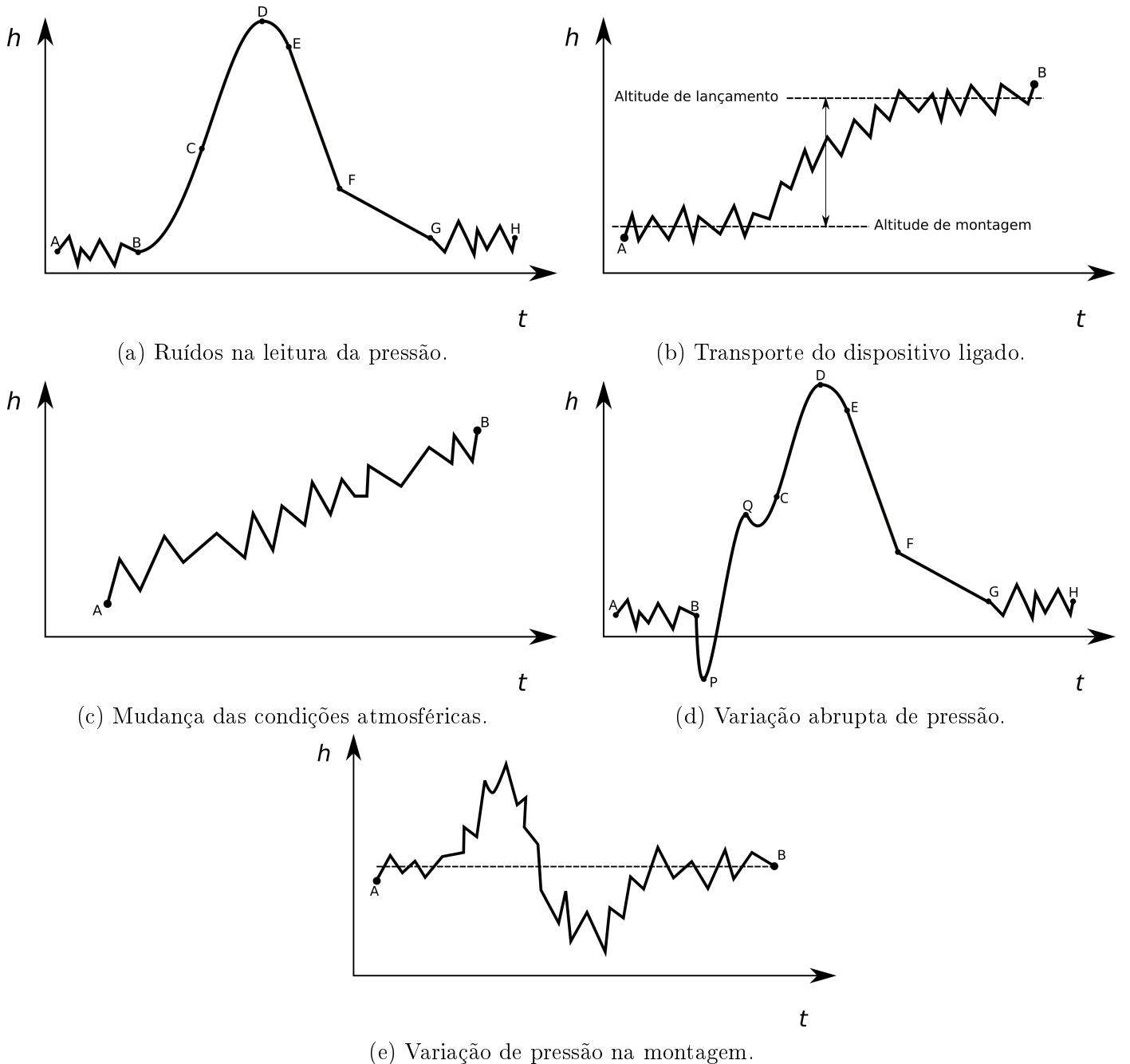


Figura 2: Situações adversas na leitura do sensor de pressão.

As situações adversas 1 e 6 exigem cuidado do usuário na construção e na montagem do minifoguete. As situações 2 a 8 são levadas em consideração através dos critérios de detecção de eventos de voo (Seç. 2.3). As situações 8 e 9 são consideradas no algoritmo do sistema de recuperação (Seç. 4). As situações 10 e 11 não podem ser resolvidas pelo altímetro, embora o algoritmo esteja preparado para mitigar a situação 11, fazendo um número configurável tentativas de acionamento de cada paraquedas.

2.3 Critérios de detecção de eventos de voo

Neste documento, são denominados eventos de voo (i) a decolagem, (ii) o apogeu, (iii) a queda, (iv) a altitude de lançamento de paraquedas e (v) o pouso. Para determiná-los, considere que $h(t)$ seja medido com período T e que as últimas $N + 1$ medidas sejam armazenadas em um vetor h_k ($1 \leq k \leq N$).

As condições de decolagem, apogeu e queda são baseadas na componente vertical v_y da velocidade. No entanto, como mencionado na Seç. 2.2, as medidas de h estão sujeitas à incerteza de leitura do barômetro

U_h e às oscilações na pressão, que podem causar falsos eventos de voo.

Uma das formas de suavizar oscilações é o uso de média. Por isso, neste documento, para calcular o valor médio \bar{v}_y da componente vertical da velocidade v_y utiliza-se a média móvel de h . O vetor h_k é dividido em duas partes e, então, calcula-se h_B e h_F , que são os valores médios de h_k na primeira e segunda metade do vetor, respectivamente.

Utilizando a regra dos trapézios, h_B e h_F são calculados como

$$h_B = \frac{1}{N/2} \left[\frac{h_0 + h_{N/2}}{2} + \sum_{k=1}^{N/2-1} h_k \right], \quad h_F = \frac{1}{N/2} \left[\frac{h_N + h_{N/2}}{2} + \sum_{k=1}^{N/2-1} h_{N-k} \right]. \quad (1)$$

Uma vez conhecidos h_B e h_F , \bar{v}_y é obtido por diferenças finitas como

$$\bar{v}_y = \frac{h_F - h_B}{NT/2}. \quad (2)$$

Note que as médias (1) são calculadas no período $\tau = NT/2$. A capacidade de suavização da média é mais efetiva quando $T_p \ll \tau$, onde T_p é período das perturbações. O valor de τ , portanto, deve ser suficientemente grande para suavizar as oscilações e suficientemente pequeno para captar adequadamente a dinâmica do voo.

Com base em h_k e no procedimento de suavização da velocidade descrito acima, os critérios de detecção de eventos de voo são:

- **Condição de decolagem:**

$$\bar{v}_y > v_{\text{lift off}}, \quad (3)$$

onde $v_{\text{lift off}}$ é a velocidade mínima para detectar a decolagem. $v_{\text{lift off}}$ deve ser suficientemente grande para que perturbações na pressão não causem acionamento prematuro do sistema de recuperação, mas não tão grande para que o lançamento não seja detectado. Sugere-se

$$20 \text{ m/s} \leq v_{\text{lift off}} \leq 40 \text{ m/s}.$$

Note que o uso do procedimento de suavização de velocidade e a condição de decolagem proposta evitam as situações adversas 2 a 6.

- **Condição de apogeu:**

$$\bar{v}_y < v_{\text{apogee}}, \quad (4)$$

onde v_{apogee} é a velocidade para detecção de apogeu. Em uma trajetória suave, o apogeu ocorre em $v_y = 0$. No entanto, devido ao atraso causado pelo processo de suavização no cálculo da velocidade, admite-se que o apogeu ocorre em uma velocidade um pouco maior que zero. Sugere-se

$$0 \leq v_{\text{apogee}} < \frac{gNT}{3} \quad \text{e} \quad v_{\text{apogee}} \ll v_{\text{lift off}},$$

onde g a aceleração gravitacional. Além disso, é fundamental que

$$v_{\text{apogee}} \ll v_{\text{lift off}},$$

para que a condição de apogeu não seja detectada logo após o lançamento.

- **Condição de implantação do paraquedas principal:**

$$h_N < H_{\text{parachute}}, \quad (5)$$

onde $H_{\text{parachute}}$ é a altitude a partir do ponto de lançamento onde o paraquedas principal deve ser acionado. O valor de $H_{\text{parachute}}$ deve ser suficientemente grande para que o paraquedas seja ejetado e o minifoguete atinja o solo com velocidade terminal. Sugere-se

$$200 \text{ m} < H_{\text{parachute}} < 300 \text{ m}$$

- **Condição de queda:**

$$-\bar{v}_y > v_{\text{fall}}, \quad (6)$$

onde v_{fall} é a velocidade mínima para detectar a queda. Esta condição é importante em caso de reinicialização do dispositivo durante o voo (situação adversa 8). v_{fall} deve ser suficientemente grande para que perturbações na pressão não causem acionamento prematuro do paraquedas auxiliar, mas não tão grande para que o paraquedas auxiliar seja ejetado em alta velocidade. Sugere-se

$$20 \text{ m/s} \leq v_{\text{fall}} \leq 40 \text{ m/s}.$$

- **Condição de pouso:**

$$\max_{0 \leq k \leq N} (h_k) - \min_{0 \leq k \leq N} (h_k) < D_{\text{land}}, \quad (7)$$

onde D_{land} é o deslocamento máximo aceitável no intervalo de tempo $\Delta t = N \times T$. O intervalo sugerido para D_{land} é

$$2U_h \leq D_{\text{land}} \leq 4U_h. \quad (8)$$

Note que a condição de pouso proposta evita a situação adversa 7.

3 Hardware

Esta seção ainda está em elaboração...

**Fazer uma apresentação geral do hardware, descrevendo a sua organização e propósito.
Mencionar a restrição de memória EEPROM e a forma de armazenamento dos dados de voo**

O esquema elétrico do rRocket é ilustrado na Fig. 3 (**atualizar a figura para a versão mais recente**). O altímetro é composto pelos seguintes elementos

1. Placa Arduino Nano (microcontrolador ATMega 328P). Esta placa foi escolhida pelo tamanho, preço acessível e facilidade de uso.
2. Módulo BMP280 (barômetro).
3. Bateria 6LR61 (9 V).
4. Chave para acionamento.
5. Capacitor eletrolítico de 1000 μF (25 V).
6. Transistor TIP122 para acionamento do paraquedas auxiliar (*drogue*).
7. Transistor TIP122 para acionamento do paraquedas principal.
8. Conector para o ignitor elétrico (*squib*) do *drogue*.
9. Conector para o ignitor elétrico (*squib*) do paraquedas principal.

10. Buzzer para comunicação sonora.
11. Botão (*push button*) para a interação com o usuário.
12. Resistor de $10\text{ k}\Omega$ para o botão.
13. Resistor de $470\ \Omega$ para o transistor do *drogue*.
14. Resistor de $470\ \Omega$ para o transistor do paraquedas.
15. Resistor de $330\ \Omega$ para o capacitor.
16. Diodo 1N4007 para proteção do circuito.
17. LED vermelho TH-5mm para comunicação visual.
18. Resistor de $330\ \Omega$ para o LED.

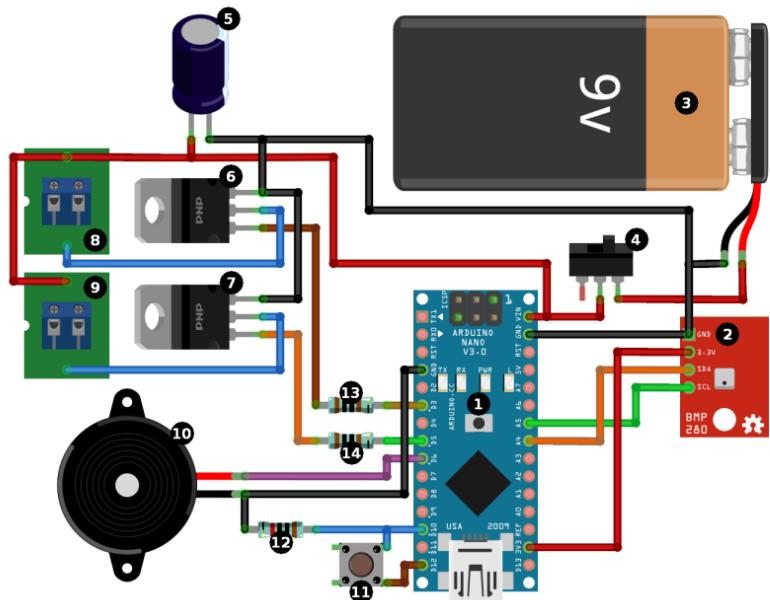


Figura 3: Ilustração do esquema elétrico do rRocket

Ignidores elétricos podem utilizar mais corrente do que a baterias 6LR61 conseguem fornecer. Além disso, o aumento da corrente elétrica faz com que a tensão nos terminais da bateria diminua. Para uma corrente suficientemente alta, é possível que a tensão seja tão baixa a ponto de desligar o Arduino. Para contornar esse problema, o rRocket utiliza uma capacitor para armazenar energia elétrica e realizar o acionamento dos ignidores. Em testes de bancada, observou-se que capacitores eletrolíticos de $1000\ \mu\text{F}$ carregados até ?? V são capazes de acionar ignidores com resistência de $1,6\ \Omega$ (Fig. ??). (**completar as informações**) Para carregar o capacitor sem sobreregar a bateria, utiliza-se um resistor como limitador de corrente. A escolha da resistência é descrita a seguir.

Os autores não encontraram na literatura uma indicação padronizada de faixas de corrente que as baterias 6LR61 suportam. Os *datasheets* normalmente apresentam gráficos de tensão da bateria *versus* o tempo para diversas cargas resistivas. Tipicamente a resistência de $270\ \Omega$ é utilizada para indicar alto consumo (33 mA), embora a bateria possa fornecer correntes mais altas (250 mA) por um tempo menor. Com base nestas informações, considera-se razoável que a bateria forneça corrente constante de 33 mA e 50 mA em alguns momentos de pico de uso.

A bateria deve alimentar diretamente o Arduino e o capacitor. De acordo com o site do fabricante¹, o Arduino Nano consome cerca de 19 mA. Assumindo uma corrente de pico de 50 mA na bateria, o resistor do capacitor foi dimensionado em $330\ \Omega$, o que leva ao consumo de pico para carga do capacitor seja de 27 mA ($9\text{ V}/330\ \Omega$).

O resistor de base dos transístores TIP122 foi dimensionado em $470\ \Omega$ para que a corrente no pino digital do Arduino seja da ordem de 10 mA. **Concluir a elaboração.**

Inserir foto do rRocket2022 e do rRocket-EZ montados.

4 Software

A estrutura do código-fonte do rRocket foi elaborada para refletir a estrutura do seu *hardware* e levar em conta a trajetória esperada para voos típicos de minifoguetes, bem como as situações adversas da Seç. 2.2.

A representação do *hardware* no código-fonte foi baseada no paradigma de Orientação a Objetos e implementada na linguagem C++. O dispositivo é representado através da classe *RecoverySystem*, que por sua vez, contém instâncias das classes *Barometer*, *Actuator*, *Memory*, *Button* e *HumanInterface*. A Fig. 4 apresenta um diagrama UML das classes do rRocket e seus métodos públicos. Cada classe possui tarefas bem definidas:

- *RecoverySystem*: responsável pela gestão do sistema de recuperação;
- *Barometer*: responsável pela leitura do sensor BMP280, registro da altura atual e do apogeu;
- *Actuator*: responsável pela implantação do paraquedas auxiliar (*drogue*) e do paraquedas principal;
- *Memory*: responsável pela gestão da memória EEPROM (gravação e leitura de dados);
- *Button*: responsável pela observação do estado do botão. O botão pode estar em um dos seguintes estados: solto (*released*), pressionado (*pressed*), pressionado e solto em seguida (*pressedAndReleased*) e pressionado longamente (*longPressed*).
- *HumanInterface*: responsável pela comunicação com o usuário através da porta serial (USB), LED (visual) e *buzzer* (sonoro).

Para realizar a gestão do sistema de recuperação, a classe *RecoverySystem* utiliza o conceito de *Máquina de Estados Finitos*. Neste conceito, o comportamento do dispositivo muda de acordo com o seu estado e o estado muda de acordo com eventos predeterminados.

Considerando a trajetória típica de voo de minifoguetes (Fig. 1), foram identificados cinco estados para o dispositivo:

- *readyToFly* ($A \rightarrow B$): o dispositivo está aguardando o lançamento;
- *flying* ($B \rightarrow E$): o dispositivo está em voo;
- *drogueChuteActive* ($E \rightarrow F$): o dispositivo está em queda com a implantação do paraquedas auxiliar;
- *parachuteActive* ($F \rightarrow G$): o dispositivo está em queda com a implantação do paraquedas principal;
- *recovered* ($G \rightarrow H$): o dispositivo foi recuperado.

A Figura 5 apresenta nos balões os estados possíveis para o sistema de recuperação. As setas entre os balões indicam os eventos que geram as mudanças de estado e o sentido da mudança.

Existem cinco eventos que geram mudanças de estado:

¹<https://store.arduino.cc/products/arduino-nano>

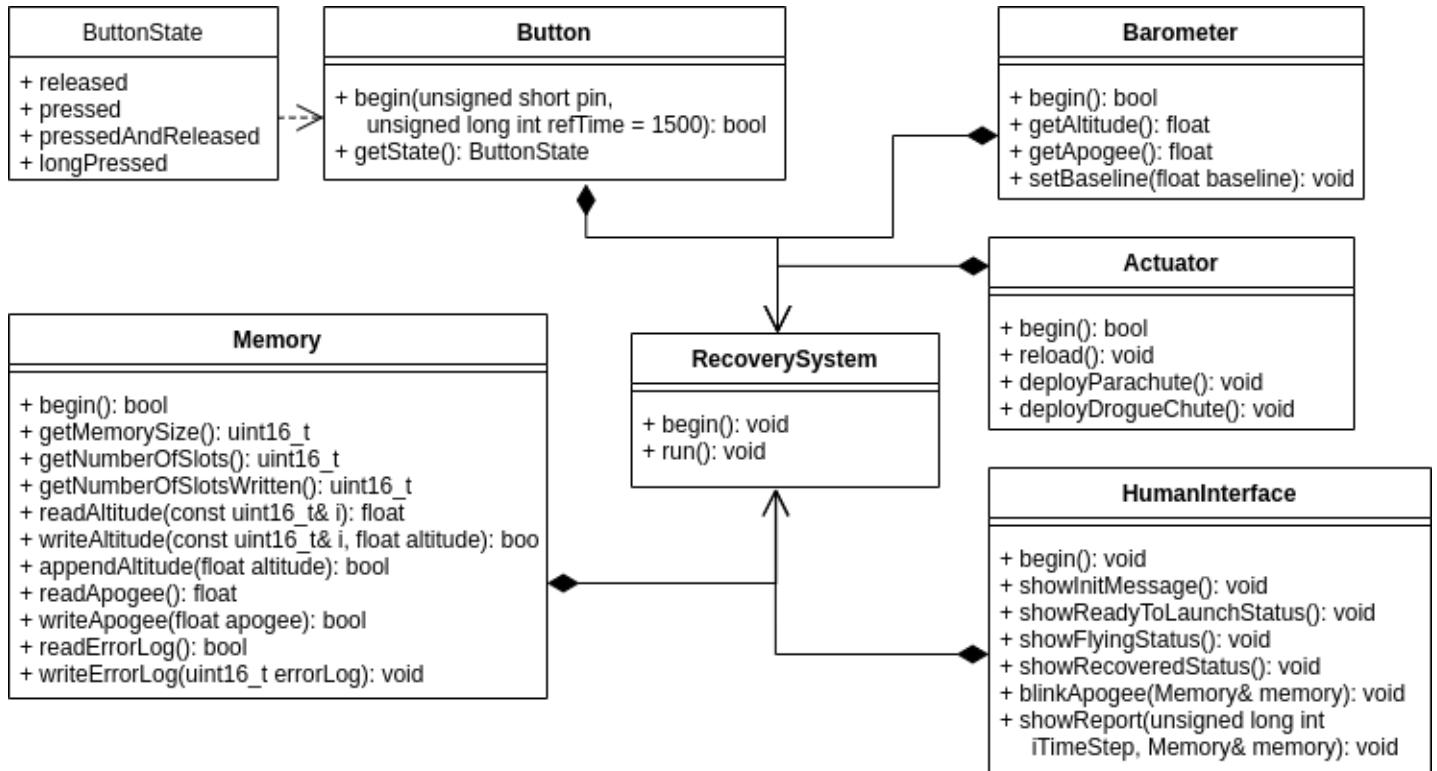


Figura 4: Diagrama UML das classes do rRocket.

- Decolagem: o critério de decolagem da Seç. 2.3 foi satisfeito.
- Queda: o critério de queda da Seç. 2.3 foi satisfeito.
- Altitude de abertura de paraquedas + capacitor carregado: o critério de implantação de paraquedas principal da Seç. 2.3 foi satisfeito e o capacitor para acionamento dos ignitores elétricos está carregado.
- Pouso: o critério de pouso da Seç. 2.3 foi satisfeito.
- Memória apagada: o usuário requisitou a limpeza da memória para o próximo voo.

Note na Figura 5 que o estado inicial do altímetro depende do estado da memória. Se já houver dados gravados, o altímetro inicia no estado *recovered*. Nesse estado, o altímetro fica sensível à decolagem ou à queda. Isso é necessário para lidar com reinicialização do altímetro durante o voo (situação adversa 8 - Seç. 2.2). **É fundamental que o usuário limpe a memória antes de cada voo.** Se o altímetro for lançado no estado *recovered*, o critério de queda pode ser satisfeito imediatamente após o lançamento, devido à situação adversa 5 (Seç. 2.2), causando a implantação prematura dos paraquedas. Caso a memória esteja limpa na inicialização, ou ainda, caso a memória seja apagada enquanto o estado é *recovered*, o estado do altímetro é alterado para *readyToFly*. Este estado é sensível apenas ao evento de decolagem, para evitar a situação adversa 5.

Caso o altímetro detecte decolagem no estado *readyToFly* ou caso detecte decolagem OU queda no estado *recovered*, o estado do altímetro é alterado para *flying*. Os dados do vetor h_k são gravados na memória permanente (EEPROM). Durante o voo, no estado *flying*, as condições de apogeu e queda são monitoradas. Se qualquer uma delas for satisfeita, o estado do altímetro é alterado para *drogueChuteActive*. Neste estado, o atuador inicia o primeiro ciclo de disparo do paraquedas auxiliar. O ciclo consiste em manter o ignitor elétrico ligado pelo intervalo de tempo T_{act} e desligado pelo intervalo de tempo $T_{recharge}$, para permitir que o capacitor seja recarregado. O usuário pode definir quantas vezes o ciclo deve ser repetido $N_{attempt}$. A redundância nos disparos é uma forma de mitigar a situação adversa 11. Caso a altitude de lançamento do paraquedas principal seja alcançada e o ciclo de disparo do paraquedas auxiliar tenha sido

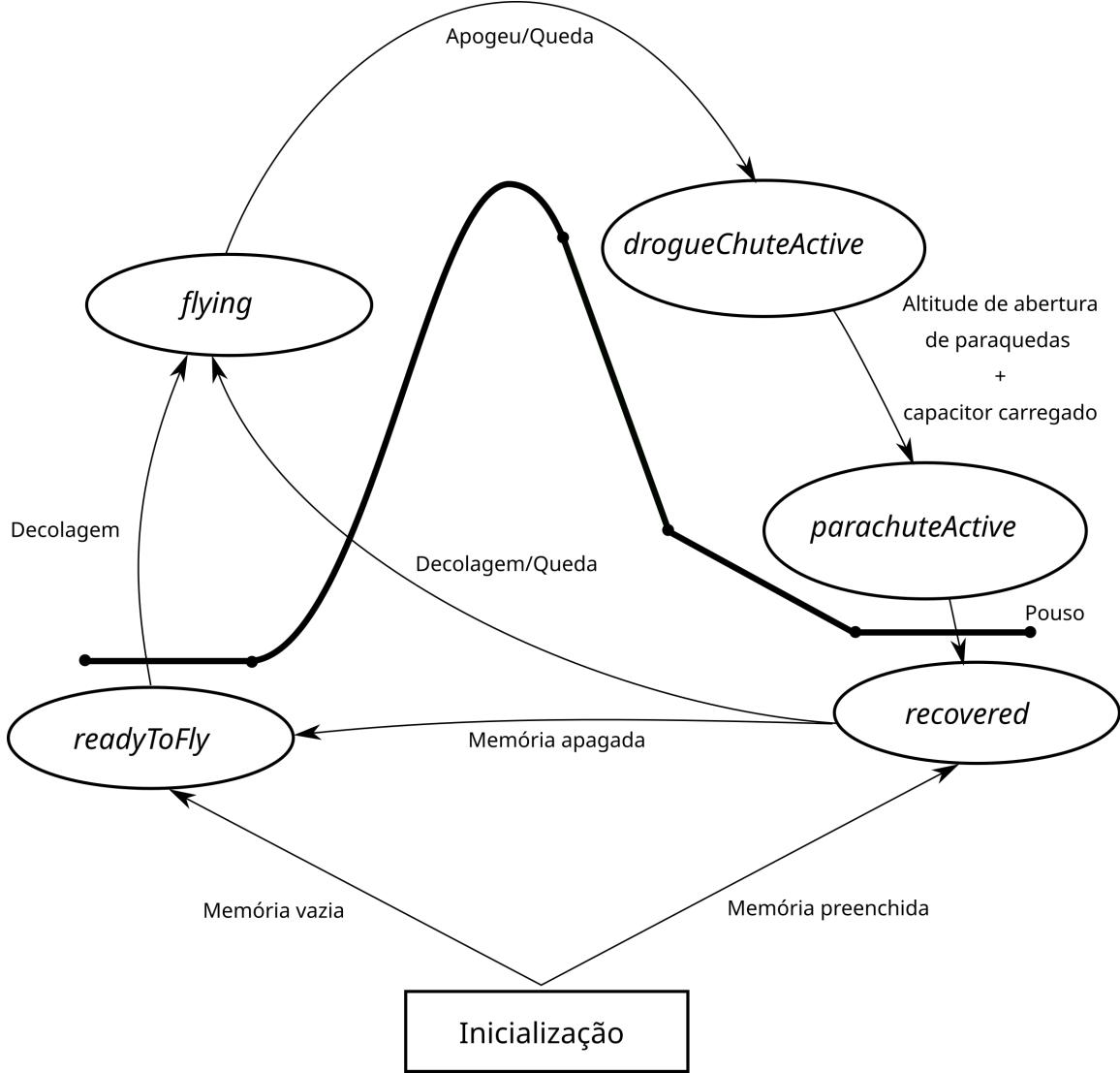


Figura 5: Diagrama de estados e eventos do rRocket.

concluído, o estado do altímetro é alterado para *parachuteActive*. Note que não é necessário que todas as tentativas de disparo do paraquedas auxiliar tenham sido executadas para a mudança de estado. Basta que um dos ciclos de disparo tenha sido concluído. No estado *parachuteActive*, o atuador inicia o ciclo de disparo do paraquedas principal. O ciclo é repetido até $N_{attempt}$ vezes. Caso o critério de pouso seja satisfeito, o estado do altímetro é alterado para *recovered*. Neste estado, o altímetro não registra mais os dados de voo na memória e observa o estado do botão. Se o botão for pressionado por menos de 3 segundos, o *buzzer*, o *LED* e a comunicação serial são utilizados para informar os dados do último voo. Caso pressionado por mais de 3 segundos, a memória é apagada.

No estado *readyToFly*, o altímetro liga o *LED* e aciona o *buzzer* a cada 1,5 segundos para indicar que o sistema está pronto para o lançamento. O usuário não deve lançar o minifoguete se esta condição não for satisfeita.

No estado *recovered*, ao pressionar o botão por menos de 3 segundos, o altímetro apresenta o relatório do último voo. O apogeu é apresentado através de acionamentos rápidos do *LED* e do *buzzer*. Cada piscada/*beep* curta corresponde a uma unidade. Uma piscada/*beep* longa indica o algarismo 0. O altímetro indica inicialmente a casa dos milhares (se necessário). Em seguida indica a casa das centenas (se necessário) e assim por diante. Cada casa decimal é separada por uma breve pausa. Após a apresentação do apogeu, os dados de tempo e altitude são enviados através da comunicação serial.

Todos os parâmetros de configuração estão em um único arquivo arquivo *Parameters.h*. Na Tab. 1

estão listados os parâmetros e seus significados.

Tabela 1: Parâmetros de configuração do rRocket

Variável	Valor	Significado
<i>softwareVersion</i>	rRocket v.1.5.6	Versão do <i>firmware</i>
<i>pinLed</i>	13	Número do pino para o LED
<i>pinBuzzer</i>	11	Número do pino para o <i>buzzer</i>
<i>pinButton</i>	10	Número do pino para o botão
<i>pinDrogueChute</i>	12	Número do pino para o paraquedas auxiliar
<i>pinParachute</i>	3	Número do pino para o paraquedas principal
<i>speedForLiftoffDetection</i>	30	Velocidade para detectar lançamento (\bar{v}_{liftoff}) - m/s
<i>speedForFallDetection</i>	30	Velocidade para detectar queda (\bar{v}_{fall}) - m/s
<i>speedForApogeeDetection</i>	7	Velocidade para detectar apogeu (\bar{v}_{apogee}) - m/s
<i>parachuteDeploymentAltitude</i>	200	Altura para disparar o paraquedas principal ($H_{\text{parachute}}$) - m
<i>displacementForLandingDetection</i>	3	Deslocamento para detectar pouso (D_{land}) - m
<i>maxNumberOfDeploymentAttempts</i>	3	Número máximo de tentativas de acionamento dos ignitores elétricos (N_{attempt})
<i>actuatorDischargeTime</i>	500	Tempo para manter o atuador ligado (T_{act}) - ms
<i>capacitorRechargeTime</i>	1000	Tempo para manter o atuador desligado e recarregar o capacitor (T_{recharge}) - ms

O altímetro também registra possíveis erros. A Tab. 2 lista os códigos de erro e seus significados.

Tabela 2: Códigos de erro do rRocket

Código	Significado
0	Nenhum erro detectado
1	Falha de inicialização do barômetro
2	Falha de inicialização do atuador
3	Altitude menor que o valor armazenável (16 bits)
4	Altitude maior que o valor armazanável (16 bits)
5	Voo iniciado com memória não vazia

5 Testes

Diversos testes foram planejados para avaliar os critérios de detecção de eventos de voo, o *software* de controle e o *hardware* do altímetro. Tais testes estão organizados em cinco categorias:

- A primeira visa avaliar os critérios de detecção de eventos de voo com base em trajetórias fabricadas e que levam em conta as adversidades mencionadas na Seç. 2.2.
- A segunda categoria visa avaliar os critérios de detecção de eventos de voo com base em trajetórias de voos reais, cujos dados foram obtidos com altímetros comerciais. Cabe destacar que o altímetro rRocket não foi utilizado nas duas primeiras categorias. Elas servem apenas ao propósito de testar os critérios de detecção de eventos de voo.
- A terceira categoria consiste em testes em bancada utilizando o altímetro no modo *debug*. Neste modo, o barômetro não lê a pressão, mas utiliza algumas das trajetórias fabricadas da categoria anterior. Além disso, o altímetro imprime na porta serial o seu *status* (tempo, altitude, velocidade e

estado). O objetivo desta categoria é avaliar o funcionamento do algoritmo para alguns dos casos da primeira categoria de testes e também avaliar a capacidade de acionamento dos ignitores elétricos. As atividades dos terminais do paraquedas auxiliar, do paraquedas principal e do capacitor são monitoradas.

- Na quarta categoria, o altímetro, em modo normal, é submetido a variações de pressão controladas. As atividades dos terminais do paraquedas auxiliar, do paraquedas principal e do capacitor são monitoradas.
- A quinta categoria consiste em voos reais com o uso do rRocket.

5.1 Critérios de detecção de eventos de voo - voos fabricados

Três funções são introduzidas para testar os critérios de detecção de eventos de voo: (i) $h_{\text{base}}(t)$, que modela uma trajetória com propulsão seguida de voo balístico, (ii) $h_{\text{noise}}(t)$ para adicionar o ruído com amplitude U_h e (iii) $h_{\text{pert}}(t)$, que representa perturbações nas medições de pressão.

A função base $h_{\text{base}}(t)$ é definida como

$$h_{\text{base}}(t) = \begin{cases} 0 & \text{se } t < t_B, \\ a_{\text{prop}}(t - t_B)^2/2 & \text{se } t_B \leq t \leq t_C, \\ h_C + v_C(t - t_C) - g(t - t_C)^2/2 & \text{se } t_C < t \leq t_G, \\ h_G & \text{se } t_G < t, \end{cases} \quad (9)$$

onde t_B é o instante de início da queima do propelente, $t_C = t_B + T_{\text{burn}}$, onde T_{burn} é o tempo de queima do propelente, a_{prop} é a aceleração constante durante a fase propulsada, h_G é a altitude de pouso do minifoguete. Os demais parâmetros são calculados como

$$h_C = a_{\text{prop}}(t_C - t_B)^2/2, \quad (10)$$

$$v_C = a_{\text{prop}}(t_C - t_B), \quad (11)$$

e

$$t_G = t_C + \frac{v_C + \sqrt{v_C^2 + 2g(h_C - h_G)}}{g}. \quad (12)$$

O apogeu ocorre no instante t_D , dado por

$$t_D = t_C + \frac{v_C}{g}. \quad (13)$$

O ruído $h_{\text{noise}}(t)$ é calculado como

$$h_{\text{noise}}(t) = U_h \cdot (2\mathcal{R} - 1), \quad (14)$$

onde \mathcal{R} é uma variável aleatória com distribuição uniforme entre 0 e 1.

Por fim, a função de perturbação $h_{\text{pert}}(t)$ é dada por

$$h_{\text{pert}}(t) = h_{\text{pert},1}(t) + h_{\text{pert},2}(t) + h_{\text{pert},3}(t), \quad (15)$$

onde

$$h_{\text{pert},i}(t) = \begin{cases} A \sin\left(2\pi \frac{t-t_{b,i}}{T_p}\right) & \text{se } t_{b,i} \leq t \leq t_{e,i}, \\ 0 & \text{caso contrário.} \end{cases}, \quad 1 \leq i \leq 3. \quad (16)$$

A representa a amplitude, T_p o período, $t_{b,i}$ e $t_{e,i}$ os instantes de início e fim, respectivamente, da i -ésima perturbação.

Tabela 3: Parâmetros para as funções de teste.

Parâmetro	Valor	Unidade
a_{prop}	2g	m/s ²
T_{burn}	3	s
t_B	10	s
h_G	40	m
U_h	1	m
A	50	m
T_p	1,2	s
$t_{b,1}$	$t_B/3$	s
$t_{e,1}$	$t_{b,1} + T_p$	s
$t_{b,2}$	t_B	s
$t_{e,2}$	0,8 t_B	s
$t_{b,3}$	1,2 t_D	s
$t_{e,3}$	t_G	s

Tabela 4: Parâmetros para os critérios de detecção de eventos de voo.

Parâmetro	Valor	Unidade
N	12	-
T	0,25	s
v_{liftoff}	30	m/s
v_{fall}	30	m/s
v_{apogee}	7	m/s
$H_{\text{parachute}}$	200	m
D_{land}	3	m

A Tab. 3 apresenta os valores padrão dos parâmetros das funções de teste.

Os parâmetros padrão para aplicação dos critérios de detecção de eventos de voo da Seç. 2.3 são dados na Tab. 4.

Para avaliar os critérios de detecção de eventos de voo, 14 voos fabricados foram considerados. Os objetivos de cada caso são discutidos a seguir. O resumo das configurações de cada caso são dados na Tab. 5.

- **Caso 1:** $h(t) = h_{\text{base}}$ a trajetória é uma função suave, como se pode observar na Fig. 7. A mesma figura apresenta os pontos onde os critérios de detecção de eventos foram satisfeitos e a velocidade calculada com o processo de suavização. Observe que a localização dos pontos é condizente com o que se espera, exceto pelo atraso na detecção do apogeu. O atraso é causado pelo procedimento de suavização da velocidade e pode ser reduzido aumentando-se o parâmetro v_{apogee} . Note, no entanto, que a diferença entre o apogeu detectado (259,7 m) e o real (264,6 m) é de apenas 4,9 m. O atraso causado pela suavização é compensado pelas vantagens obtidas nos casos a seguir.
- **Caso 2:** adiciona-se ruído à trajetória base, isto é, $h(t) = h_{\text{base}} + h_{\text{noise}}$. Como se pode observar na Fig. 8, os critérios de detecção de eventos de voo continuam funcionando adequadamente.
- **Caso 3:** adiciona-se ruído e perturbações à trajetória base, $h(t) = h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$ (Fig. 9). Há três perturbações: (i) uma durante o período de espera para lançamento, simulando rajadas laterais de vento ou variações de pressão durante a montagem do minifoguete, (ii) uma logo após o lançamento, simulando uma trajetória instável e (iii) uma após o apogeu, simulando uma queda

instável. Observe que a suavização da velocidade foi fundamental para a aplicação adequada dos critérios de detecção de eventos de voo.

- **Caso 4:** semelhante ao Caso 3, exceto pela perturbação em h ser mantida ao longo de todo o voo. Os critérios de detecção de eventos de voo produzem resultados condizentes com o esperado (Fig. 10).
- **Caso 5:** semelhante ao Caso 3, exceto pela aceleração de propulsão ser de apenas g , simulando assim, um minifoguete com baixa velocidade de lançamento. Note na Fig. 11 que o critério de implantação de paraquedas principal ocorre logo após o critério de detecção de apogeu, uma vez que o apogeu foi menor que os 200 m estipulados para o lançamento do paraquedas principal. Os demais critérios de detecção de eventos de voo produzem resultados condizentes com o esperado.
- **Caso 6:** semelhante ao Caso 3, exceto pela aceleração de propulsão ser de $6g$, simulando assim, um minifoguete com alta velocidade de lançamento. Os critérios de detecção de eventos de voo produzem resultados condizentes com o esperado (Fig. 12).
- **Caso 7:** semelhante ao Caso 3, exceto pela altitude de pouso ser menor que a de lançamento. Este caso, assim como o 3, serve para avaliar a situação adversa 7, da Seç. 2.2. Os critérios de detecção de eventos de voo produzem resultados condizentes com o esperado (Fig. 13).
- **Caso 8:** semelhante ao Caso 3, exceto pelo período da perturbação ser metade do original ($T_p = 0,6$ s). Os critérios de detecção de eventos de voo produzem resultados condizentes com o esperado (Fig. 14).
- **Caso 9:** semelhante aos Casos 3 e 8, exceto pelo período da perturbação ser o dobro do original ($T_p = 2,4$ s). Os critérios de detecção de eventos de voo **falham** na primeira perturbação (Fig. 15). Dos Casos 3, 8, 9, 10 e 11, observa-se que o procedimento de suavização é mais efetivo quando ao período utilizado no cálculo das médias ($\tau = NT/2 = 1,5$ s) é maior que o período da perturbação T_p , *i.e.*, $\tau > T_p$. Aumentar τ , no entanto, gera atraso na detecção de eventos. Os voos reais, analisados na próxima seção, apresentaram perturbações com períodos de oscilação menor ou igual a 0,8 s.
- **Caso 10:** semelhante ao Casos 3, exceto pelo número de períodos de observação ($N = 8$). Os critérios de detecção de eventos de voo **falham** na primeira perturbação (Fig. 16), pois, como comentado no item anterior, houve redução de τ .
- **Caso 11:** semelhante ao Casos 3, exceto pelo número de períodos de observação ($N = 16$). Os critérios de detecção de eventos de voo produzem resultados condizentes com o esperado (Fig. 17). Houve, como esperado, um atraso maior na detecção do apogeu.
- **Caso 12:** semelhante ao Casos 3, exceto pelo número de períodos de observação ($N = 24$) e período de leitura da pressão ($T = 0,125$ s). Os critérios de detecção de eventos de voo produzem resultados condizentes com o esperado e semelhantes ao do Caso 3, exceto por uma resolução maior na trajetória (Fig. 18).
- **Caso 13:** semelhante ao Casos 3, exceto pela aceleração de propulsão ($3g$) e início de leitura dos dados, simulando uma reinicialização do altímetro antes do apogeu. Os critérios de detecção de eventos de voo produzem resultados condizentes com o esperado (Fig. 19).
- **Caso 14:** semelhante ao Casos 3, exceto pela aceleração de propulsão ($3g$) e início de leitura dos dados, simulando uma reinicialização do altímetro após o apogeu. Os critérios de detecção de eventos de voo produzem resultados condizentes com o esperado (Fig. 20).

Tabela 5: Configurações dos casos avaliados.

Caso	Função $h(t)$	Parâmetros
1	h_{base}	Padrão
2	$h_{\text{base}} + h_{\text{noise}}$	Padrão
3	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão
4	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, exceto $t_{e,2} = 1, 2t_D$
5	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, exceto $a_{\text{prop}} = g$
6	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, exceto $a_{\text{prop}} = 6g$
7	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, exceto $h_G = -40$ m
8	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, exceto $T_p = 0, 6$ s
9	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, exceto $T_p = 2, 4$ s
10	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, exceto $N = 8$
11	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, exceto $N = 16$
12	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, exceto $N = 24$ e $T = 0, 125$ s
13	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, reinicialização em voo ($a_{\text{prop}} = 3g$, $t > 0, 5(t_B + t_D) + t_B$)
14	$h_{\text{base}} + h_{\text{noise}} + h_{\text{pert}}$	Padrão, reinicialização em voo ($a_{\text{prop}} = 3g$, $t > t_D$)

5.2 Critérios de detecção de eventos de voo - voos reais

Sete voos reais foram utilizados para testar os critérios de detecção de eventos de voo. Os voos são representativos de situações típicas de lançamentos:

- Lançamento 1: voo bem-sucedido com apogeu entre 200 e 500 m e com baixa velocidade de saída da rampa. Identificação: Netuno-F/Paraná-25/v2, LT 2 Dez 2019, apogeu = 285 m.
- Lançamento 2: voo bem-sucedido com apogeu entre 500 e 1000 m e com alta velocidade de saída da rampa. Identificação: Netuno-R-B/Paraná-7b, LT 7 Set 2017, apogeu = 794 m.
- Lançamento 3: voo bem-sucedido com apogeu acima de 1000 m e velocidade próxima à do som (Mach 0.9). Identificação: Urano/Paraná-15b, LT 7 Set 2018, apogeu = 1216 m.
- Lançamento 4: voo bem-sucedido com velocidade típica de saída da rampa. Identificação: Netuno-R-B/Paraná-26/v2, LT 2 Dez 2019, apogeu = 541 m.
- Lançamento 5: voo instável com rotação na direção vertical. Amplitude e frequência das oscilações: 125 m e 0,2 s. Identificação: Netuno-R-B/Paraná-7, LT 24 Jun 2017, apogeu = 271 m.
- Lançamento 6: voo instável lateral. Amplitude e frequência das oscilações: 85 m e 0,74 s. Identificação: Urano/Paraná-30, LT 9 Jan 2022, apogeu = 58 m.
- Lançamento 7: voo instável lateral. Amplitude e frequência das oscilações: 15 m e 0,8 s. Identificação: Plug/Paraná-24/v2, LT 15 Nov 2019, apogeu (visual) = 40 a 60 m.

Os gráficos das trajetórias e os pontos que representam os eventos de voo (considerando os parâmetros da Tab. 4) são apresentados nas Figs. 21-28. Os critérios de detecção de eventos de voo produziram resultados coerentes em todos os voos, exceto pelo Lançamento 7. Neste último caso, o lançamento não foi detectado pois $\bar{v}_y < \bar{v}_{\text{liftoff}} = 30$ m/s. O parâmetro \bar{v}_{liftoff} pode ser ajustado pelo usuário. Se $\bar{v}_{\text{liftoff}} = 15$ m/s, por exemplo, então os eventos de voo são detectados, como se pode observar na Fig. 28. Cabe notar que o altímetro comercial StratoLogger também não detectou o Lançamento 07. Além disso, não é recomendável um valor muito baixo para \bar{v}_{liftoff} , pois se o altímetro estiver ligado durante a montagem do minifoguete, variações de pressão podem causar detectar lançamento falso e causar o acionamento prematuro dos paraquedas (situação adversa 6).

5.3 Barômetro simulado

O objetivo deste conjunto de testes é avaliar o funcionamento do altímetro rRocket sob condições muito controladas. Para isso, o trecho de código responsável por retornar a leitura do barômetro real é substituída por um barômetro simulado, que retorna $h(t)$ de acordo com alguma trajetória fabricada.

Em todas as nove simulações monitorou-se o sinal elétrico nos terminais do capacitor e dos terminais dos paraquedas. Nas sete primeiras simulações, o altímetro ficou ligado à porta serial para que seu estado fosse monitorado. Para evitar problemas elétricos, nestas simulações o altímetro foi alimentado apenas através de cabo USB (a bateria 6LR61 não foi utilizada). Nas duas últimas simulações, a bateria foi ligada e o cabo USB desconectado do altímetro.

O resumo das simulações é apresentado na Tab. 6. As descrições completas das simulações e os resultados obtidos são apresentados a seguir:

- Simulação 1: utiliza a trajetória do Caso 3 da Seç. 5.1 que representa um voo com perturbações e ruídos. O altímetro é inicializado com a memória vazia. Os resultados da simulação são apresentados na Fig. 29. O eixo vertical da esquerda apresenta a altitude. O eixo vertical da direita representa o sinal elétrico (proporcional à tensão) medido no capacitor e nos terminais dos paraquedas. A trajetória registrada na memória EEPROM coincide com a trajetória lida em tempo real através da comunicação serial. Além disso, o altímetro detectou corretamente os eventos de voo. As letras ao lado das curvas indicam o estado do altímetro: R = *readyToFly*, F = *flying*, D = *drogueChuteActive*, P = *parachuteActive* e L = *recovered*. Note que a tensão no capacitor ficou praticamente constante ao longo de todo o voo. Isso ocorreu porque os terminais dos paraquedas ficaram abertos nesta simulação. Também é possível observar as três tentativas de acionamento do paraquedas auxiliar (*drogue*) e do paraquedas principal.
- Simulação 2: semelhante à Simulação 1, exceto pelo fato da memória não ser apagada antes do voo. Note na Fig. 30 que o altímetro é inicializado no estado *recovered*, como esperado, e que a memória contém os dados do último voo. Embora o voo ocorra a partir do estado *recovered*, o altímetro funciona corretamente, fazendo o acionamento dos paraquedas. O relatório de voo indica o erro 5, da tabela de erros, isto é “Voo iniciado com memória não vazia”.
- Simulação 3: semelhante Simulação 2, exceto pelo deslocamento do tempo em 16 s, simulando a reinicialização do altímetro em voo. Observe na Fig. 31 que o paraquedas auxiliar é acionado apenas uma vez, pois a condição de abertura de paraquedas já está satisfeita no instante em que o paraquedas auxiliar seria acionado pela segunda vez. Neste caso, a prioridade é acionar o paraquedas e, portanto, o altímetro altera seu estado de *drogueChuteActive* para *parachuteActive*. Em seguida, o paraquedas é acionado três vezes. O altímetro se comporta como esperado.
- Simulação 4: semelhante ao Caso 3, exceto por $a_{prop} = 10g$, $T_{burn} = 5$ s. O objetivo desta simulação é avaliar como a trajetória é registrada na memória quando o apogeu excede o limite superior (6000 m). Deve-se lembrar que, devido à restrição de memória EEPROM, decidiu-se armazenar a altitude como uma variável de 16 bits. Observe na Fig. 32 que a trajetória armazenada na memória do altímetro foi modulada ao intervalo de -500 a 6000 m. Desta forma é possível que o usuário consiga recuperar os dados do voo. O altímetro operou corretamente.
- Simulação 5: semelhante ao Caso 3, exceto por $h_G = -1600$. Esta trajetória simula a situação em que a altitude final da trajetória é menor que o limite inferior (-500 m) para armazenamento na memória do altímetro. Como se pode observar na Fig. 33, a altitude registrada na memória EEPROM foi modulada para permanecer no intervalo -500 a 6000 m. O altímetro operou corretamente.
- Simulação 6: simula a trajetória do Caso 5, isto é, um voo com apogeu inferior a 100 m e com baixa velocidade de saída da rampa. Como se pode observar na Fig. 34, o altímetro funcionou como esperado.

- Simulação 7: simula a trajetória do Caso 6, isto é, um voo com apogeu de cerca de 1800 m e alta velocidade de lançamento. Como se pode observar na Fig. 35, o altímetro funcionou como esperado.
- Simulação 8: simula a trajetória do Caso 3. O objetivo é avaliar a capacidade do altímetro em acionar quatro ignitores elétricos (dois ligados em paralelo aos terminais do paraquedas auxiliar e outros dois em paralelo aos terminais do paraquedas principal). Cada ignitor tem resistência média de $(1,6 \pm 0,1) \Omega$. Diferentemente da Simulação 1, nesta, o altímetro é alimentado por uma bateria 6LR61. No momento do teste, a tensão da bateria era de 8,54 V e a tensão no capacitor de 7,73 V. Como se pode observar na Fig. 36, há quedas de tensão abruptas (cerca de 1 ms) seguidas de recarga lenta (cerca de 1 s) no capacitor em todas as seis vezes em que os paraquedas são acionados, como esperado. Durante o acionamento do paraquedas auxiliar (*drogue*) apenas um ignitor foi acionado na primeira tentativa. O outro resistor foi acionado na segunda tentativa. Isso reforça a importância das várias tentativas de acionamento. Já para o paraquedas principal, os dois ignitores foram acionados na primeira tentativa.
- Simulação 9: simula a trajetória do Caso 5. O objetivo é avaliar a capacidade do altímetro em acionar dois ignitores elétricos em sequência (um ligado aos terminais do paraquedas auxiliar e outro aos terminais do paraquedas principal). Como se pode observar na Fig. 37, há quatro quedas de tensão no capacitor, uma para acionamento do *drogue* e três para acionamento do paraquedas principal. Os dois *squibs* foram ignitados corretamente na primeira tentativa.

Tabela 6: Resumo das simulações em bancada com o altímetro

Num.	Descrição	Terminais	
		Drogue	Paraquedas
Sim. 1	Caso 3	aberto	aberto
Sim. 2	Caso 3 + <i>recovered</i>	aberto	aberto
Sim. 3	Caso 3 + <i>recovered</i> + $t \geq 16$ s	aberto	aberto
Sim. 4	Caso 3, exceto ($a_{prop} = 10g$, $T_{burn} = 5$ s)	aberto	aberto
Sim. 5	Caso 3, exceto ($h_G = -1600$) m	aberto	aberto
Sim. 6	Caso 5	aberto	aberto
Sim. 7	Caso 6	aberto	aberto
Sim. 8	Caso 3	2 <i>squibs</i>	2 <i>squibs</i>
Sim. 9	Caso 5	1 <i>squib</i>	1 <i>squib</i>

5.4 Testes em bancada

Para avaliar o funcionamento do rRocket em modo normal de operação, produziu-se uma câmara (Fig. 6) onde é possível variar a pressão sob o altímetro. Esta câmara também é equipada com um barômetro e faz o monitoramento do sinal elétrico (proporcional à tensão) no capacitor e nos terminais dos paraquedas.

Sete experimentos foram realizados para avaliar o comportamento do altímetro em situações adversas. O resumo dos experimentos encontra-se na Tab. 7 e os detalhes de cada um são apresentados a seguir:

- Experimento 1: a variação de pressão é ajustada para simular um voo suave. A memória do altímetro foi apagada antes do experimento, de tal modo que o seu estado inicial é *readyToFly*. Na Fig. 38 é possível observar que a trajetória registrada pela instrumentação da câmara coincide com a registrada pelo altímetro (EEPROM) (eixo vertical à esquerda). Além disso, é possível observar na figura o sinal elétrico no capacitor e nos terminais dos paraquedas (eixo vertical à direita). O altímetro operou corretamente.

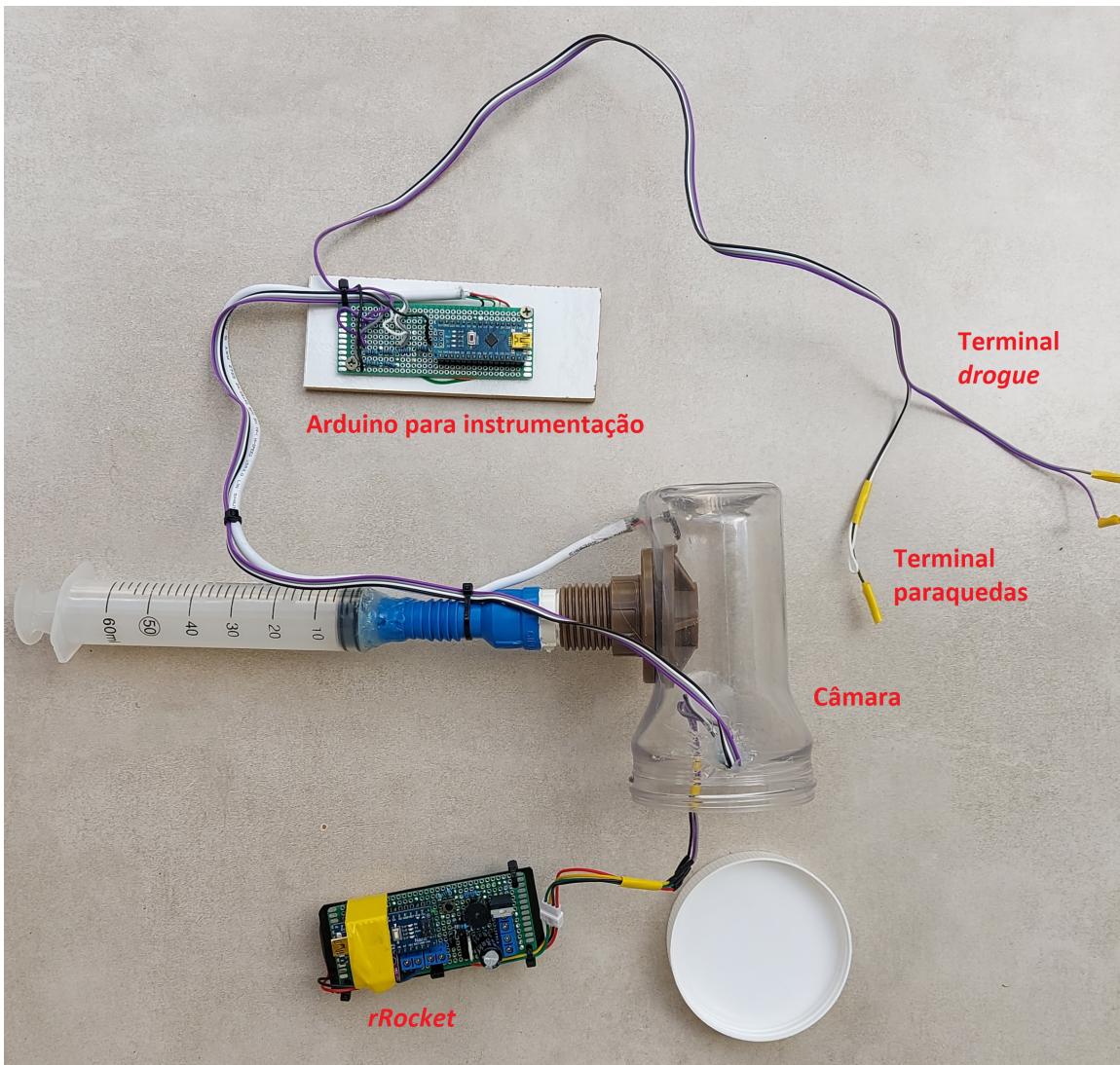


Figura 6: Aparato para instrumentação do rRocket em testes de bancada.

- Experimento 2: a variação de pressão é ajustada para simular um voo suave, porém, a memória do altímetro não foi apagada antes do experimento, de tal modo que o seu estado inicial é *recovered*. Com se pode observar na Fig. 39, o altímetro operou corretamente. O relatório de voo registrou o código de erro 5, ou seja, “Voo iniciado com memória não vazia”, como esperado.
- Experimento 3: a variação de pressão aplicada simula um voo instável. Há um decréscimo brusco em h logo após o lançamento seguido de oscilações. Com se pode observar na Fig. 40, o altímetro operou corretamente.
- Experimento 4: com a memória do altímetro parcialmente preenchida, de tal modo que seu estado inicial é *recovered*, aplicou-se um aumento de pressão para simular uma reinicialização do altímetro em voo. Os resultados são apresentados na Fig. 41. O altímetro detectou a queda e acionou os paraquedas.
- Experimento 5: a pressão foi reduzida abruptamente para simular um voo com alta velocidade de lançamento. Os resultados são apresentados na Fig. 42.
- Experimento 6: a pressão foi variada de tal modo que o apogeu fosse menor que 200 m. O objetivo é avaliar o acionamento dos paraquedas. Note na Fig. 43, que o altímetro fez uma tentativa de acionamento do *drogue* e, em seguida, fez três tentativas de acionamento do paraquedas, como esperado.

- Experimento 7: a pressão foi controlada de modo a simular o Lançamento 3, onde se observa uma redução em h imediatamente após a decolagem. Os dados do experimento são apresentados na Fig. 44. Mais uma vez, o altímetro se comportou corretamente.

Tabela 7: Resumo dos experimentos em bancada com o altímetro

Num.	Descrição	Terminais	
		Drogue	Paraquedas
Exp. 1	Voo suave	aberto	aberto
Exp. 2	Voo suave + <i>recovered</i>	aberto	aberto
Exp. 3	Voo instável	aberto	aberto
Exp. 4	Queda	aberto	aberto
Exp. 5	Alta velocidade de lançamento	aberto	aberto
Exp. 6	Apogeu < 200 m	aberto	aberto
Exp. 7	Falsa queda na decolagem	aberto	aberto

5.5 Testes em voo

Descrever os testes em voos reais com o uso do rRocket.

Referências

- [1] PerfectFlite. *StratoLoggerCF Users Manual*. Andover, NH.

A Figuras

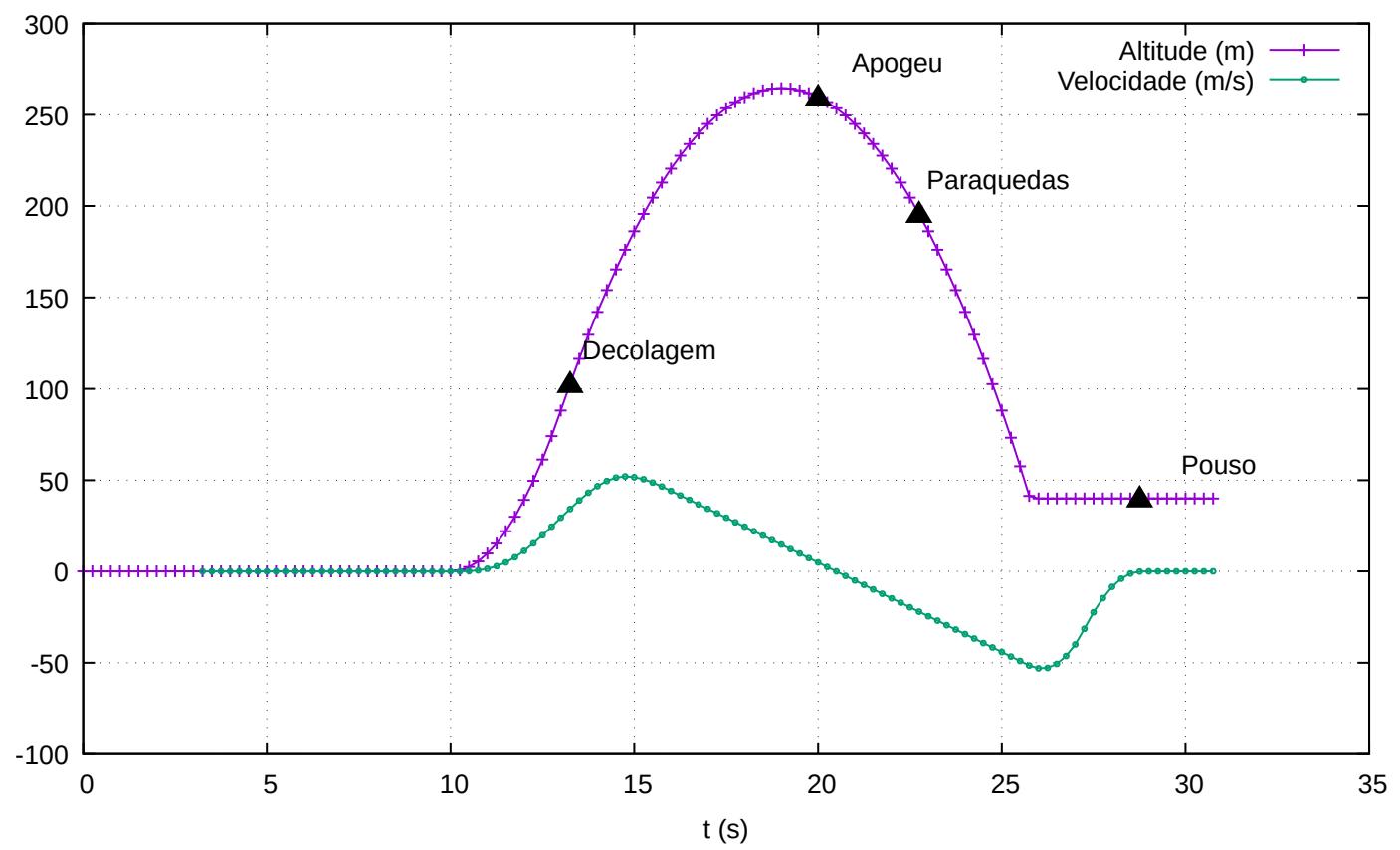


Figura 7: Aplicação dos critérios de detecção de eventos de voo. Caso 1

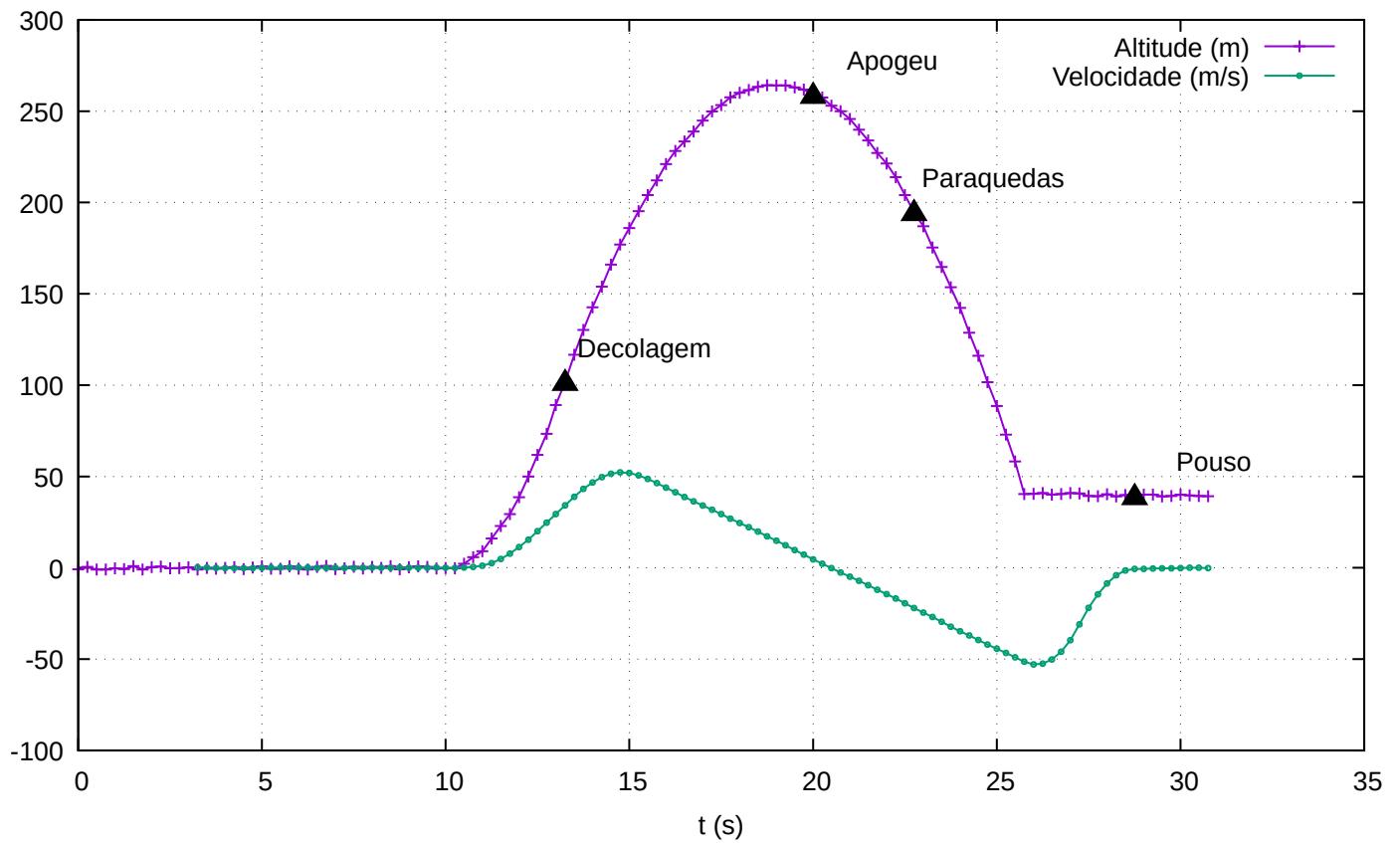


Figura 8: Aplicação dos critérios de detecção de eventos de voo. Caso 2

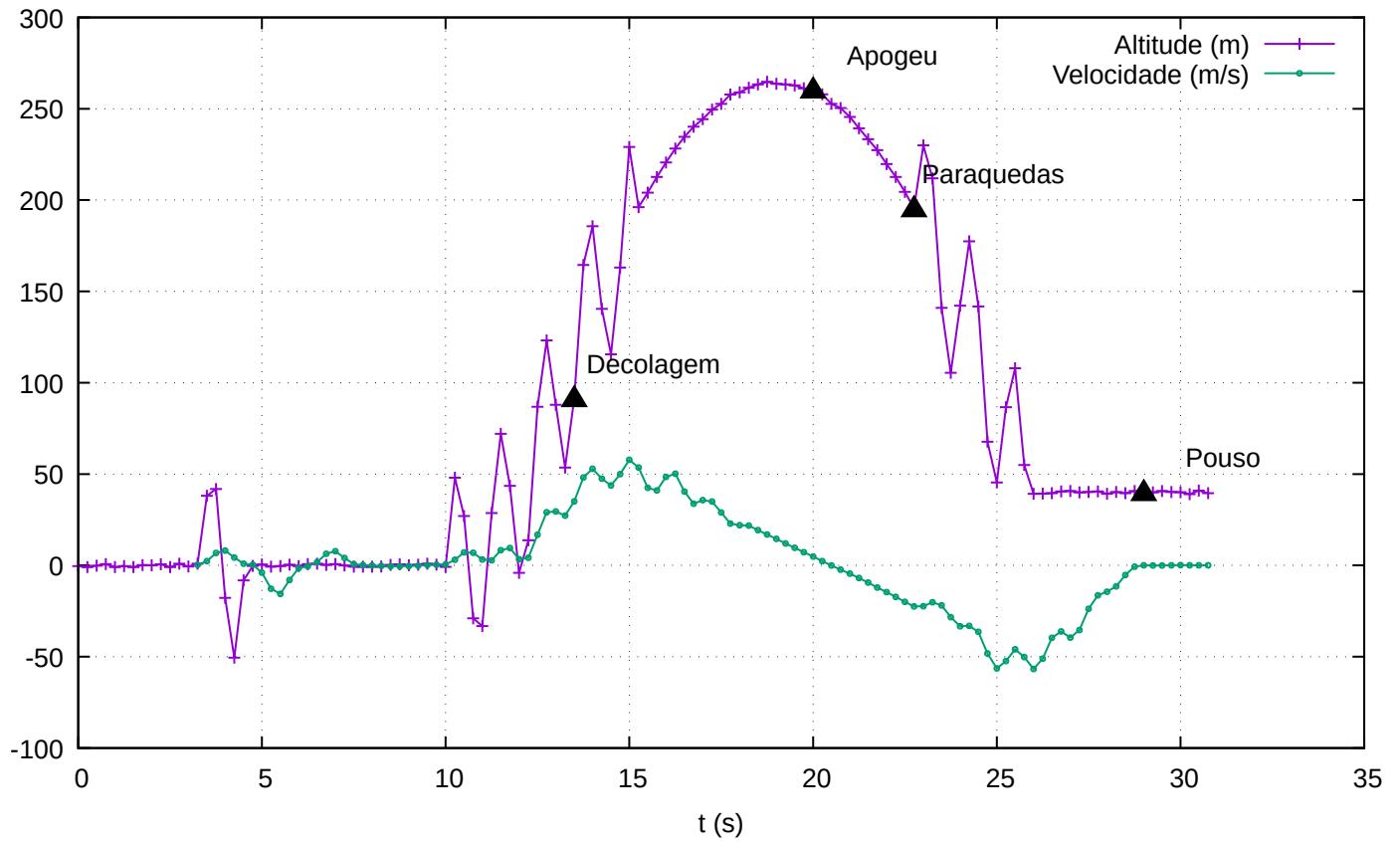


Figura 9: Aplicação dos critérios de detecção de eventos de voo. Caso 3

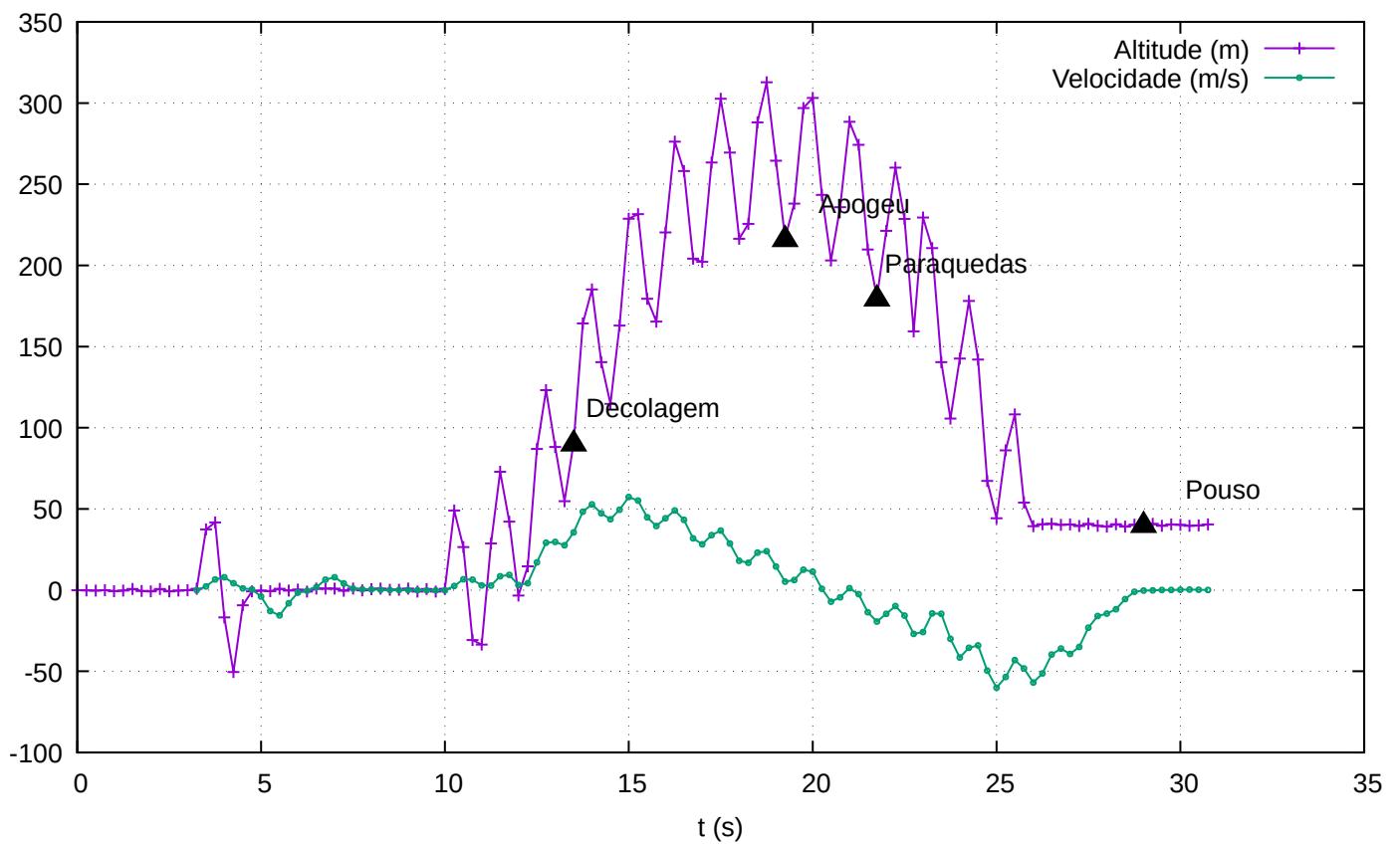


Figura 10: Aplicação dos critérios de detecção de eventos de voo. Caso 4

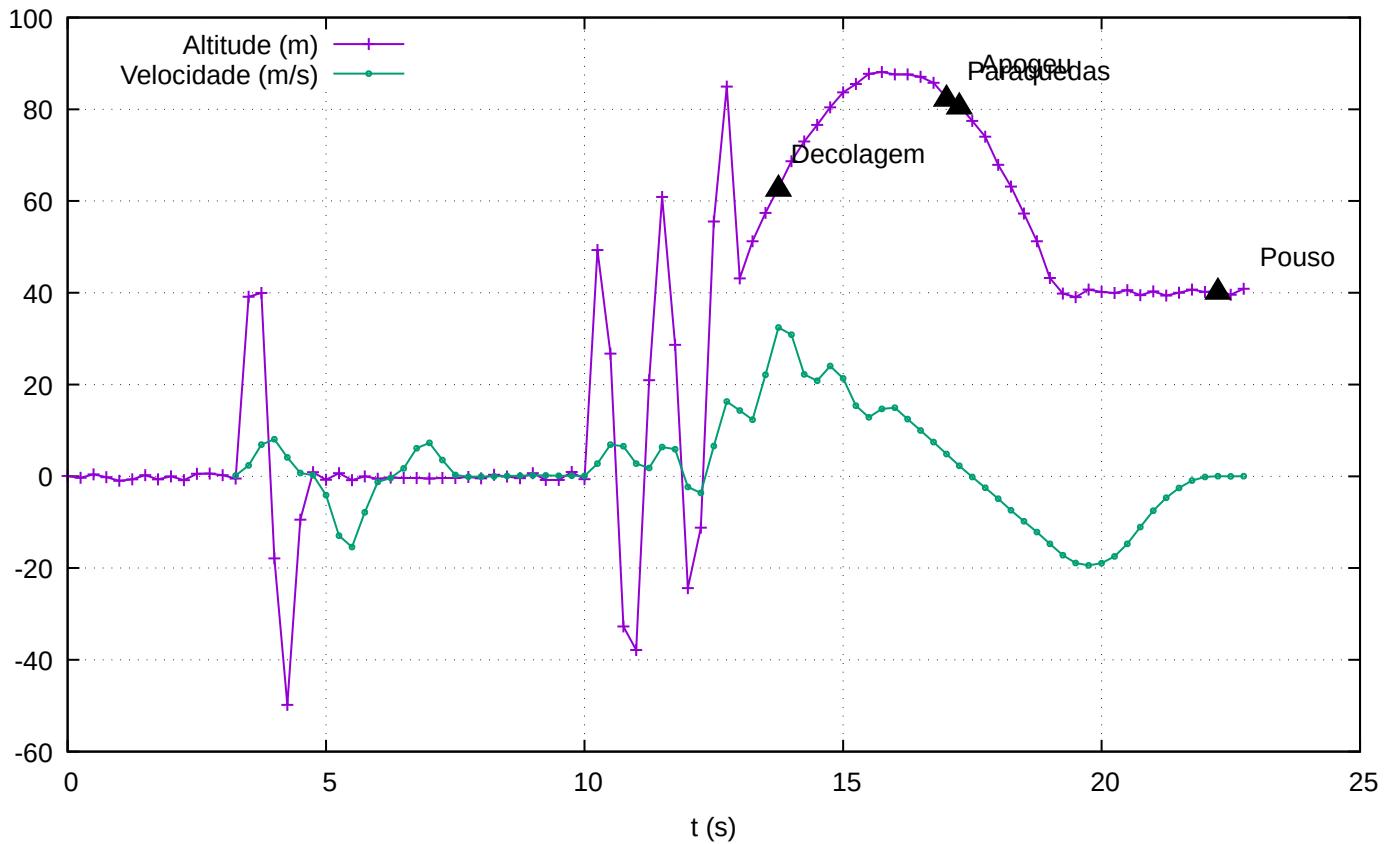


Figura 11: Aplicação dos critérios de detecção de eventos de voo. Caso 5

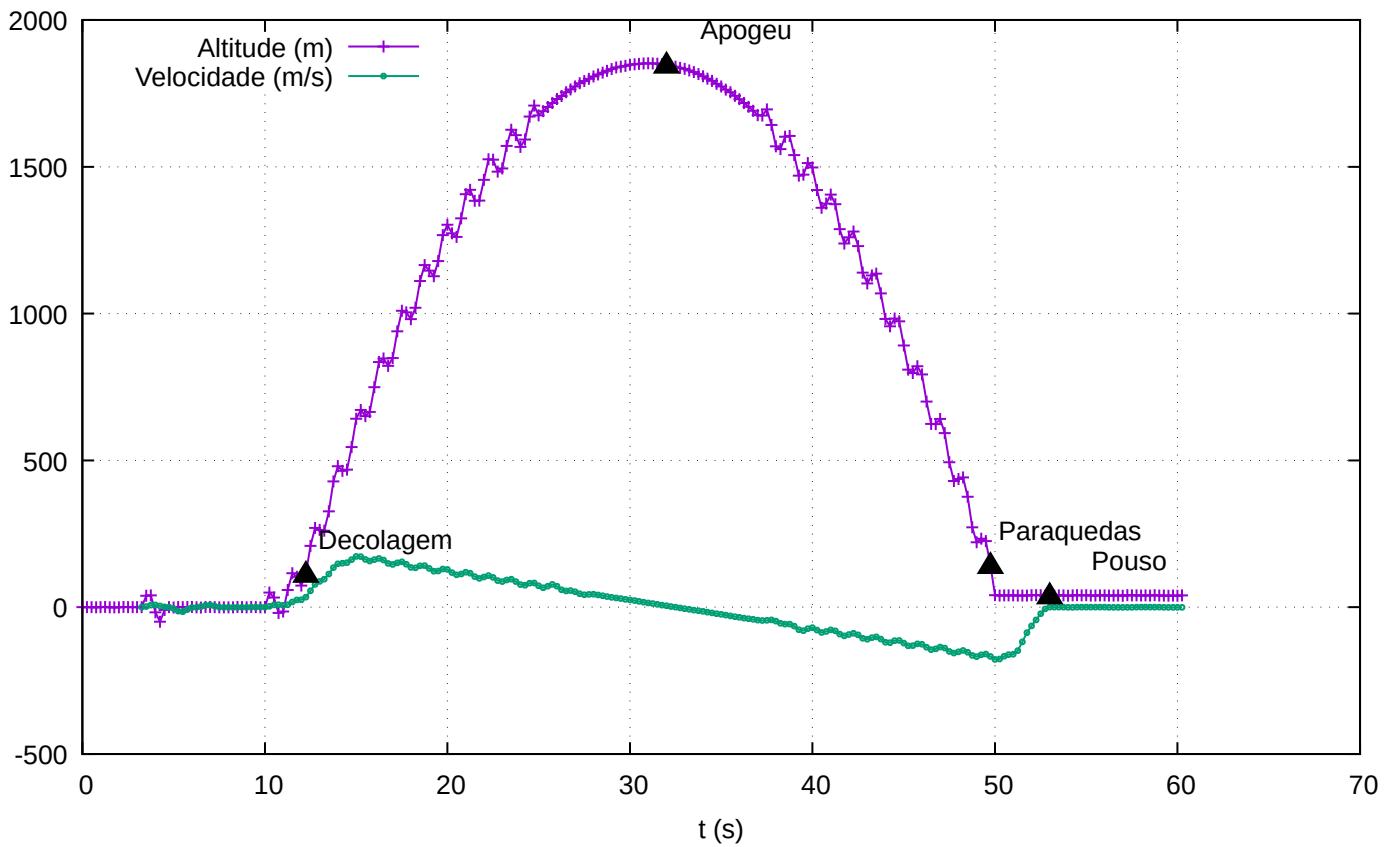


Figura 12: Aplicação dos critérios de detecção de eventos de voo. Caso 6

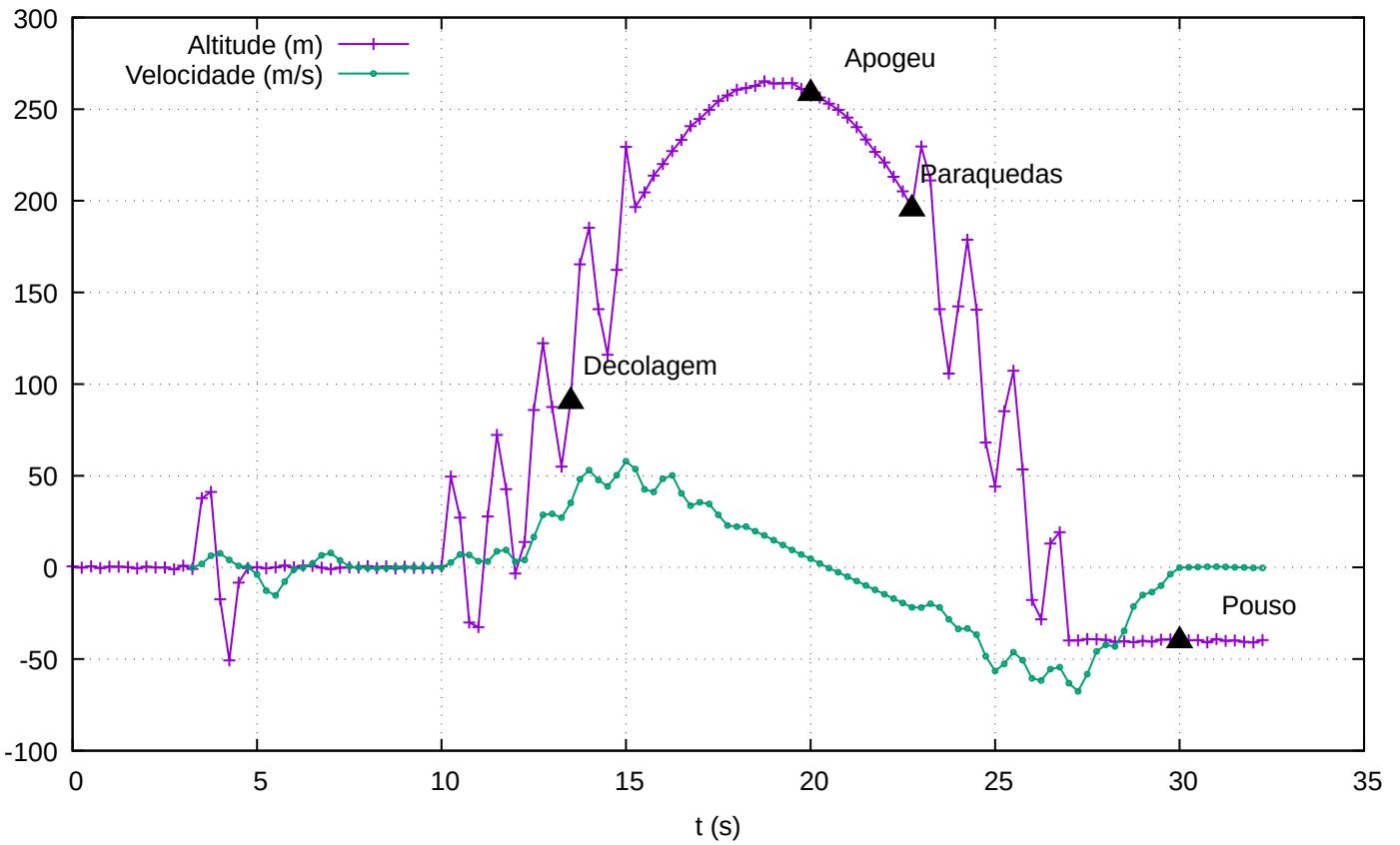


Figura 13: Aplicação dos critérios de detecção de eventos de voo. Caso 7

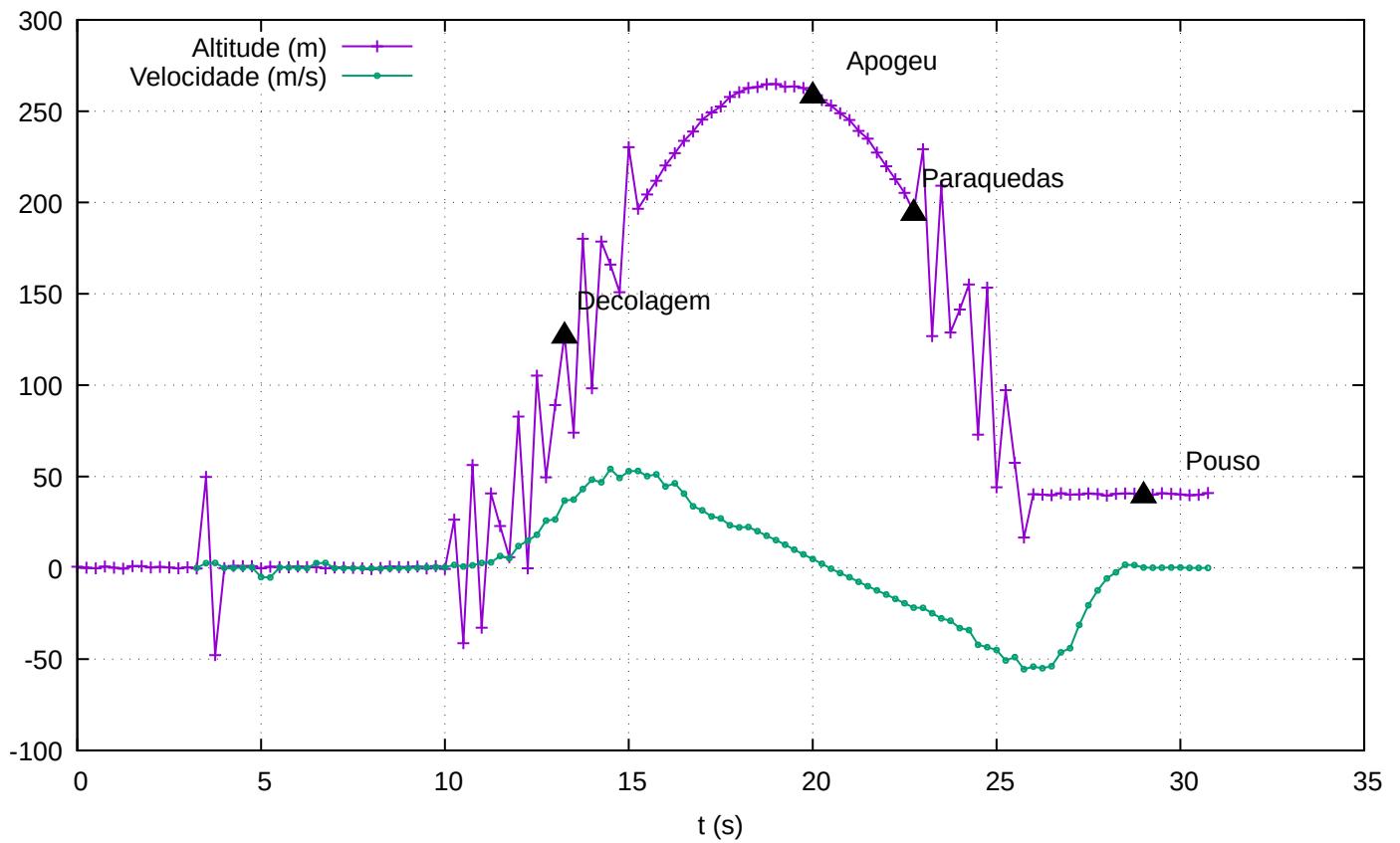


Figura 14: Aplicação dos critérios de detecção de eventos de voo. Caso 8

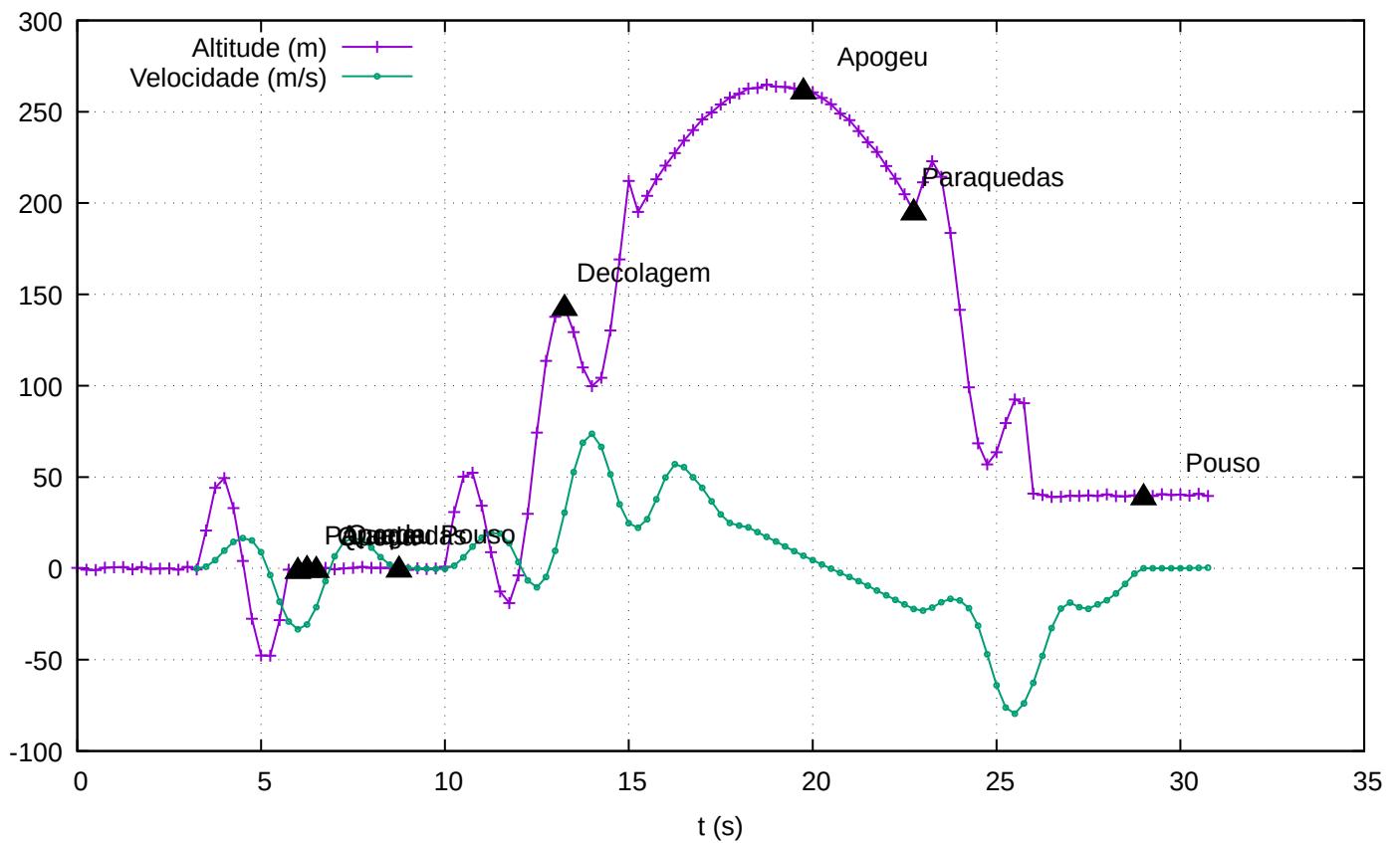


Figura 15: Aplicação dos critérios de detecção de eventos de voo. Caso 9

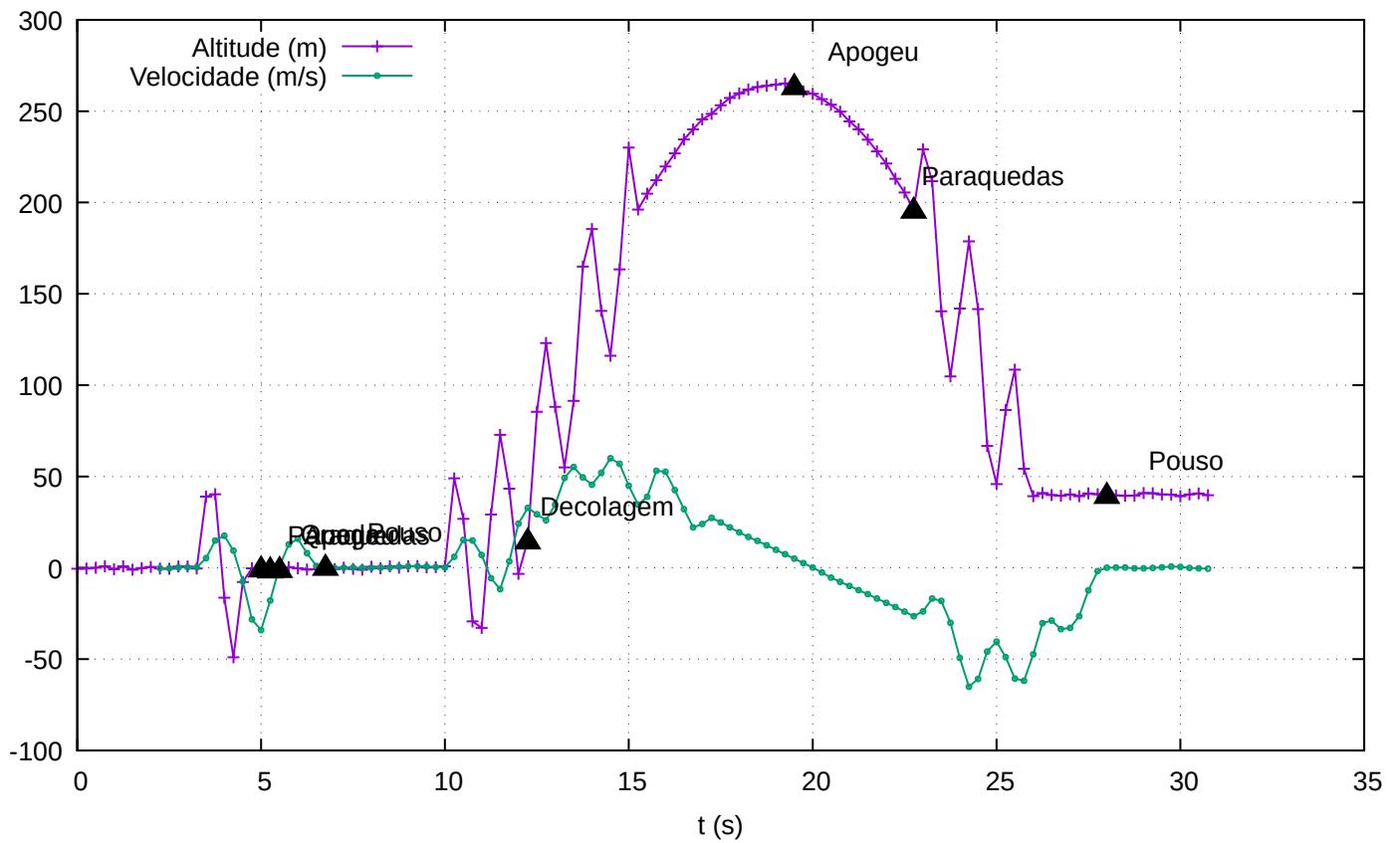


Figura 16: Aplicação dos critérios de detecção de eventos de voo. Caso 10

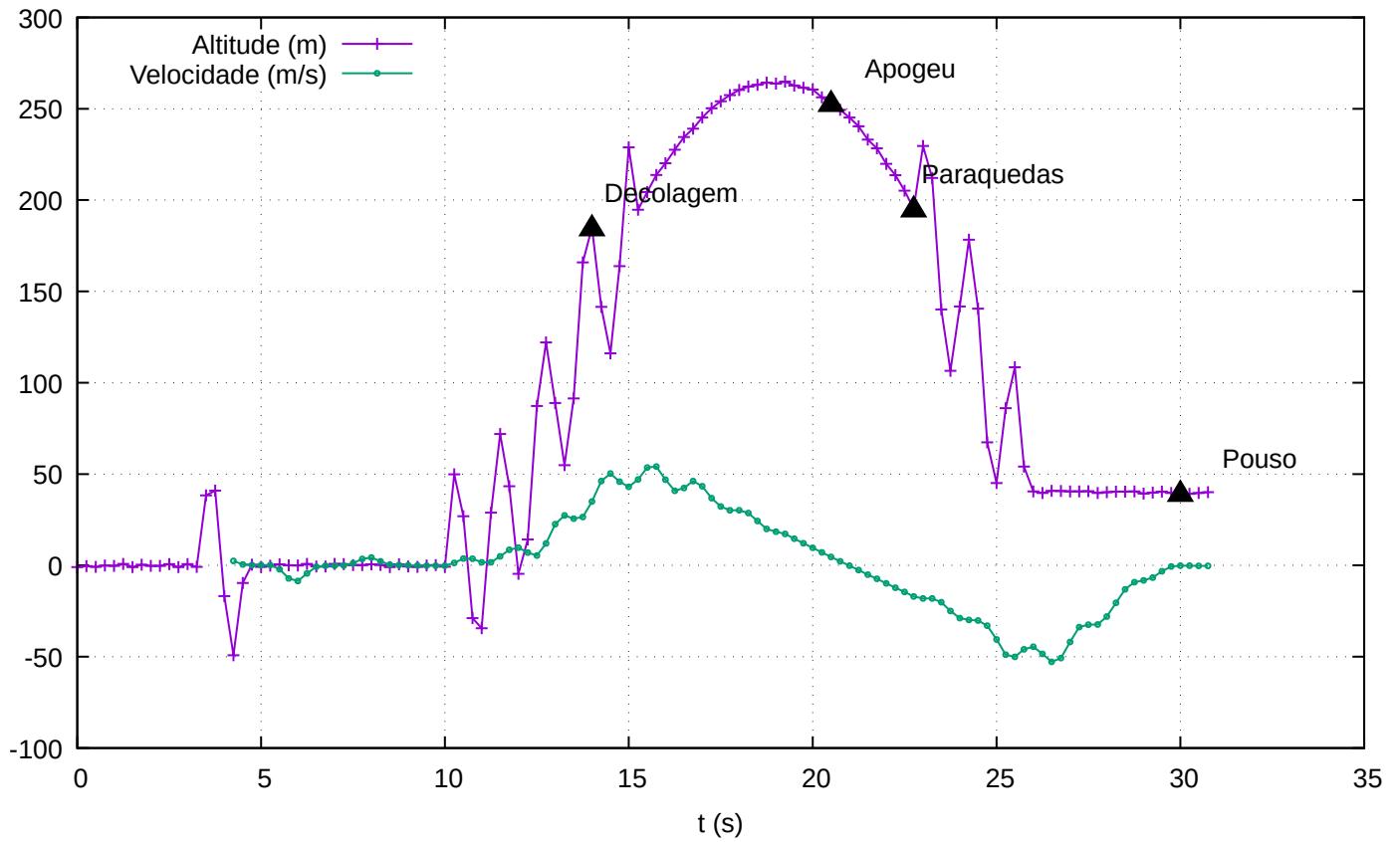


Figura 17: Aplicação dos critérios de detecção de eventos de voo. Caso 11

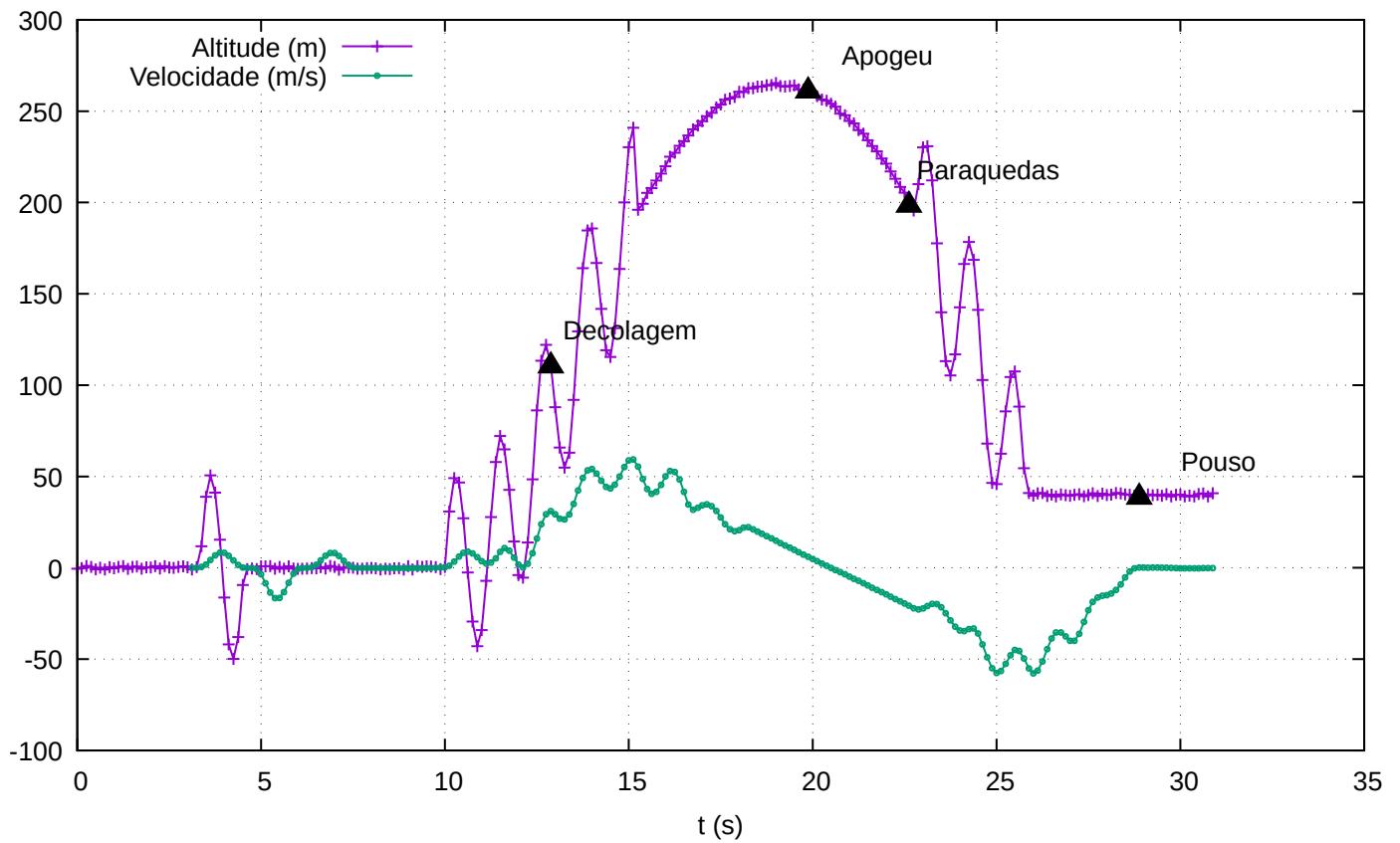


Figura 18: Aplicação dos critérios de detecção de eventos de voo. Caso 12

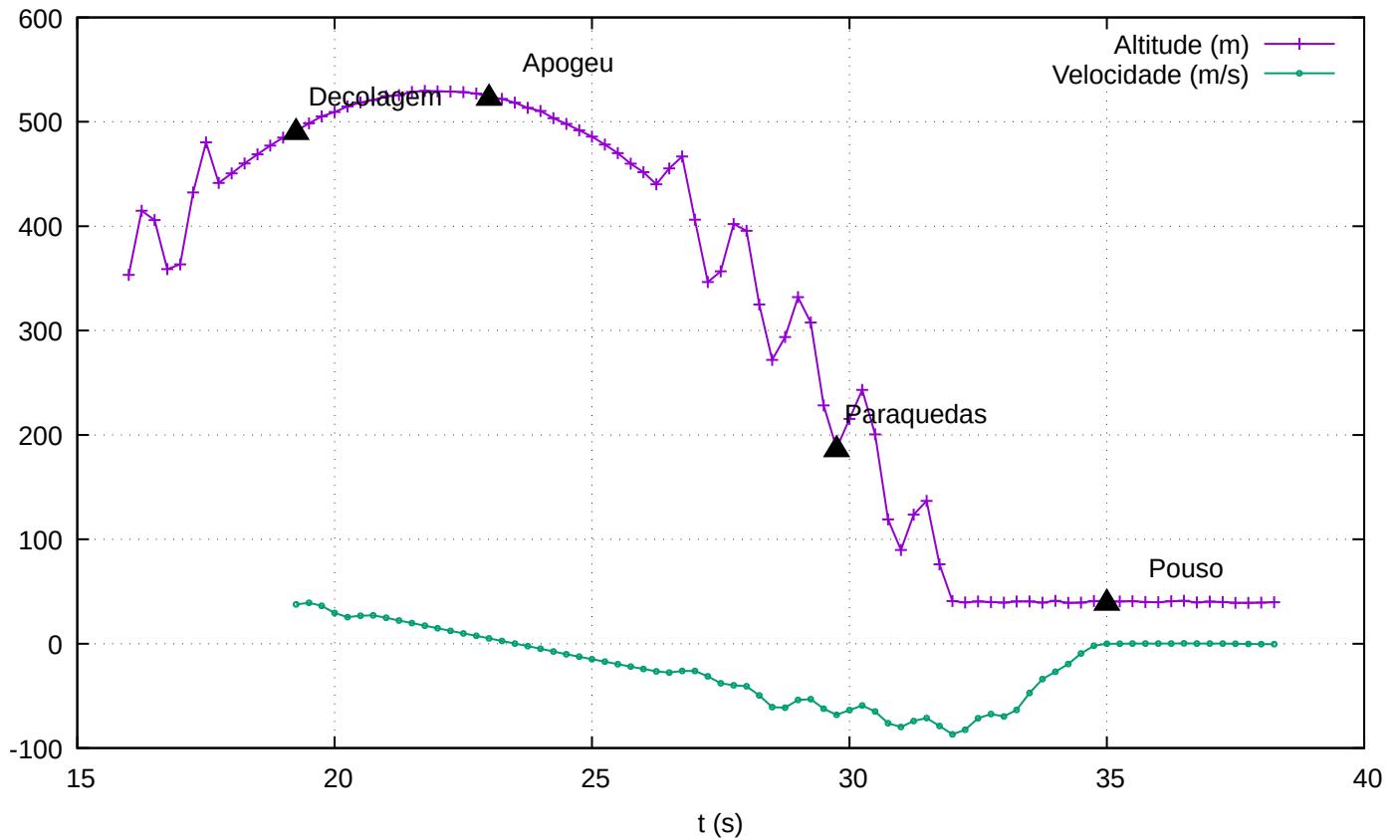


Figura 19: Aplicação dos critérios de detecção de eventos de voo. Caso 13

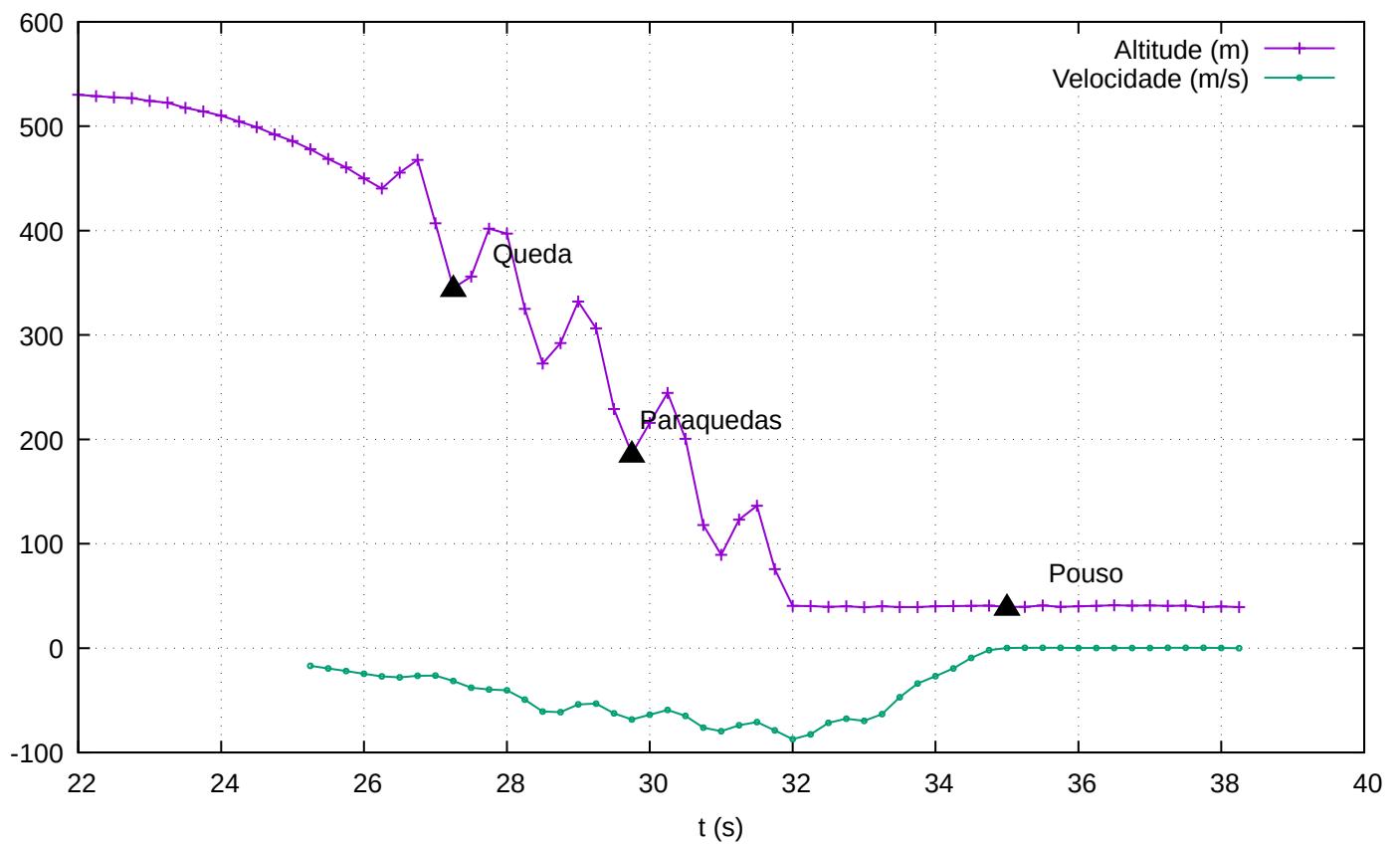


Figura 20: Aplicação dos critérios de detecção de eventos de voo. Caso 14

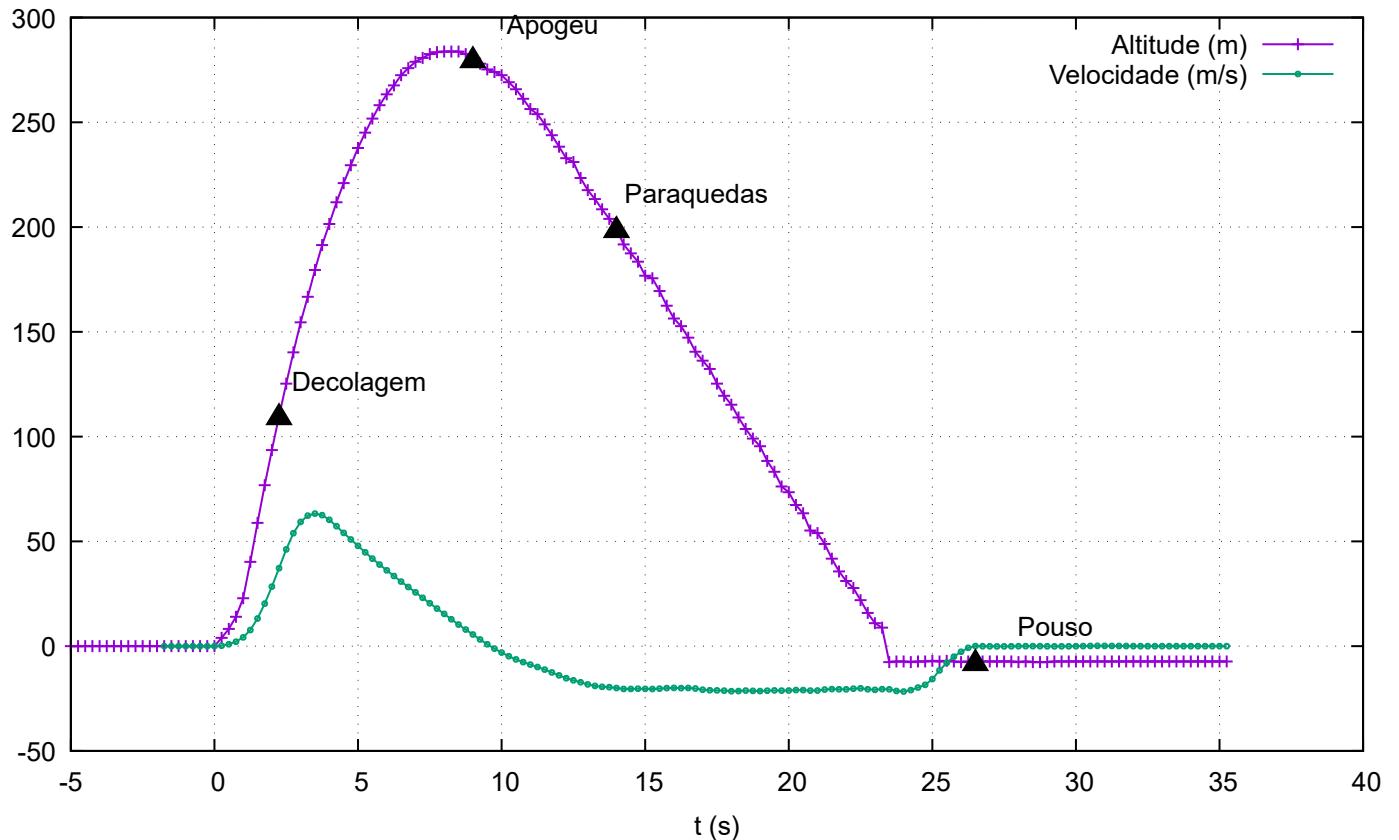


Figura 21: Aplicação dos critérios de detecção de eventos de voo para dados de voos reais. Lançamento 1

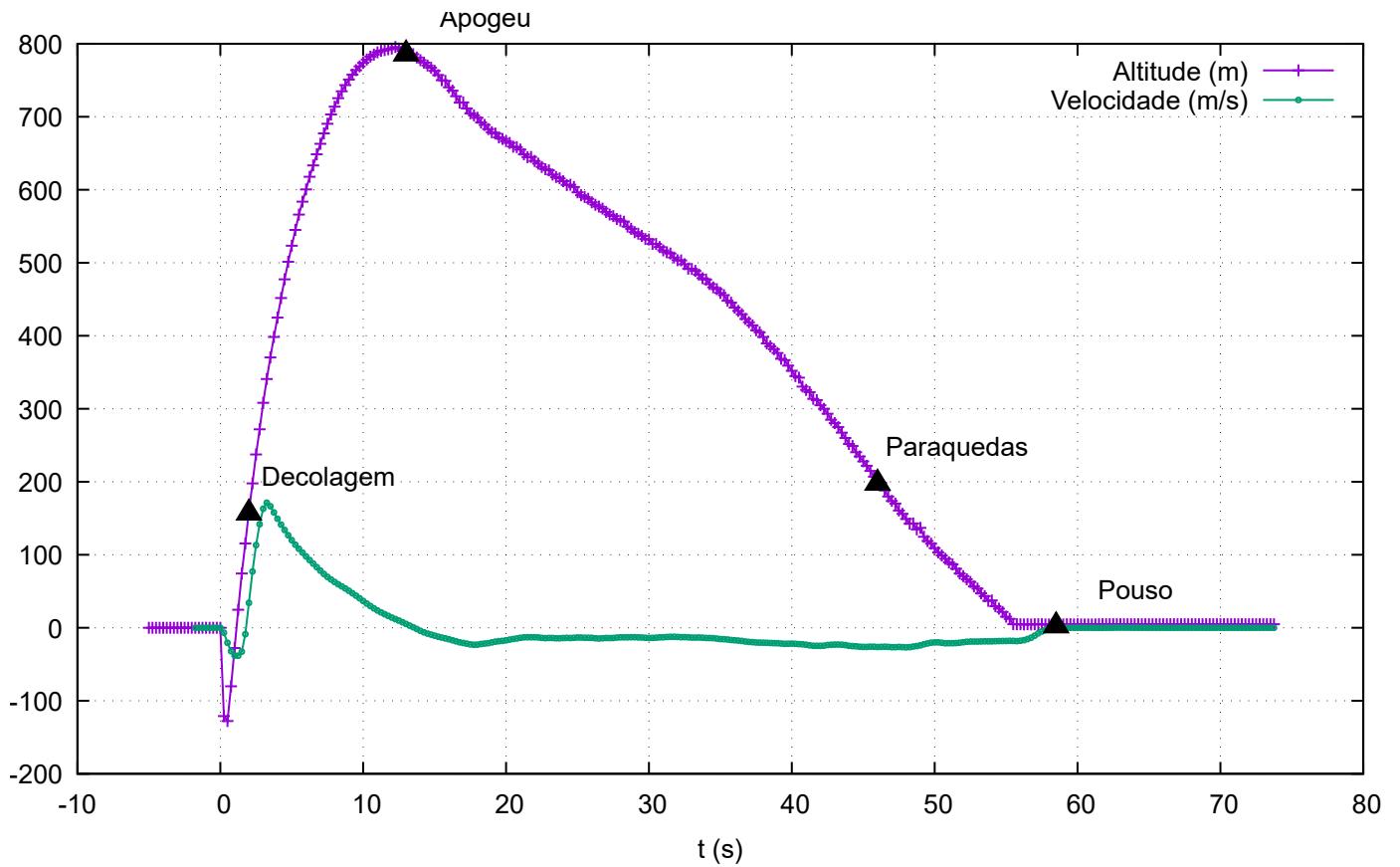


Figura 22: Aplicação dos critérios de detecção de eventos de voo para dados de voos reais. Lançamento 2

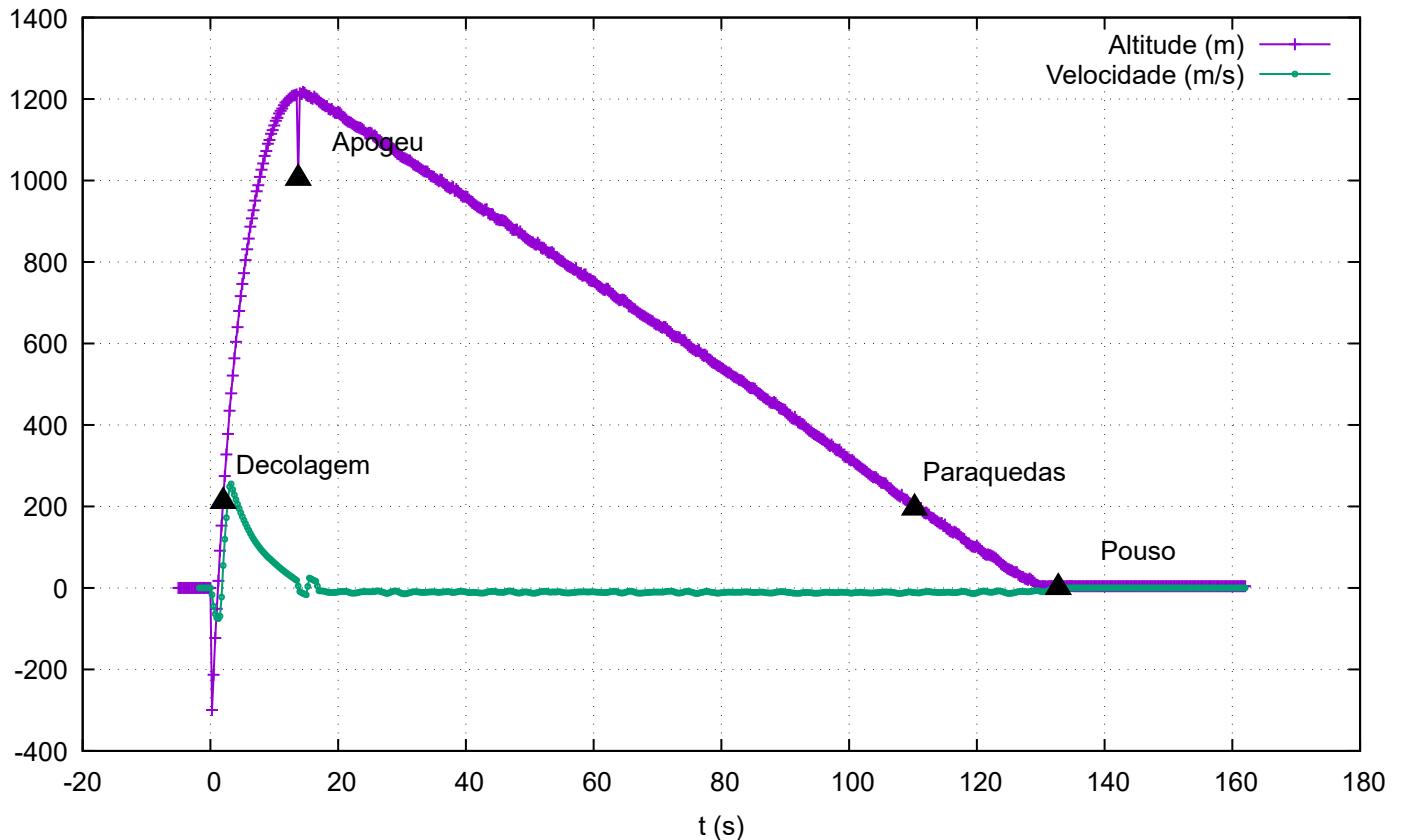


Figura 23: Aplicação dos critérios de detecção de eventos de voo para dados de voos reais. Lançamento 3

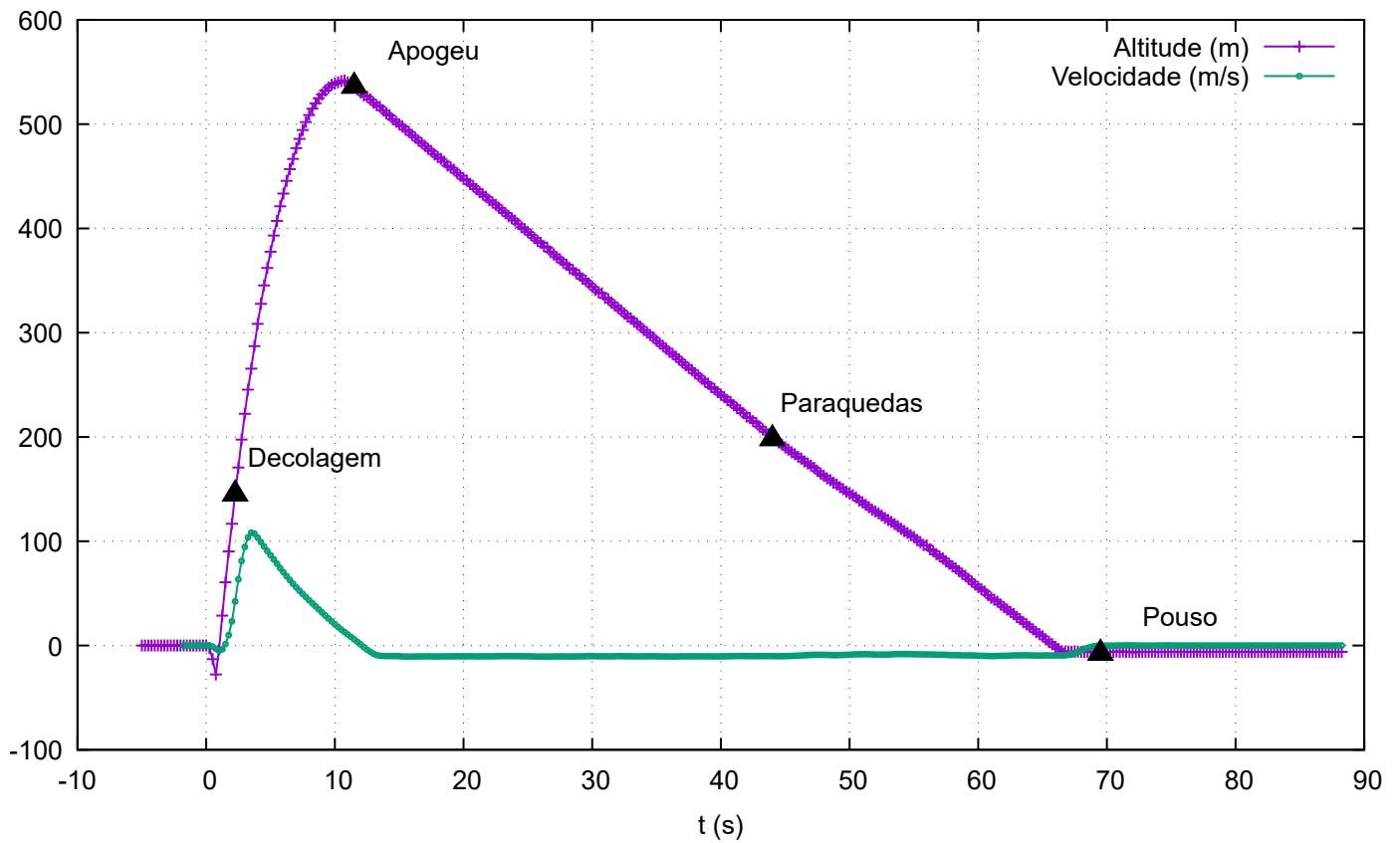


Figura 24: Aplicação dos critérios de detecção de eventos de voo para dados de voos reais. Lançamento 4

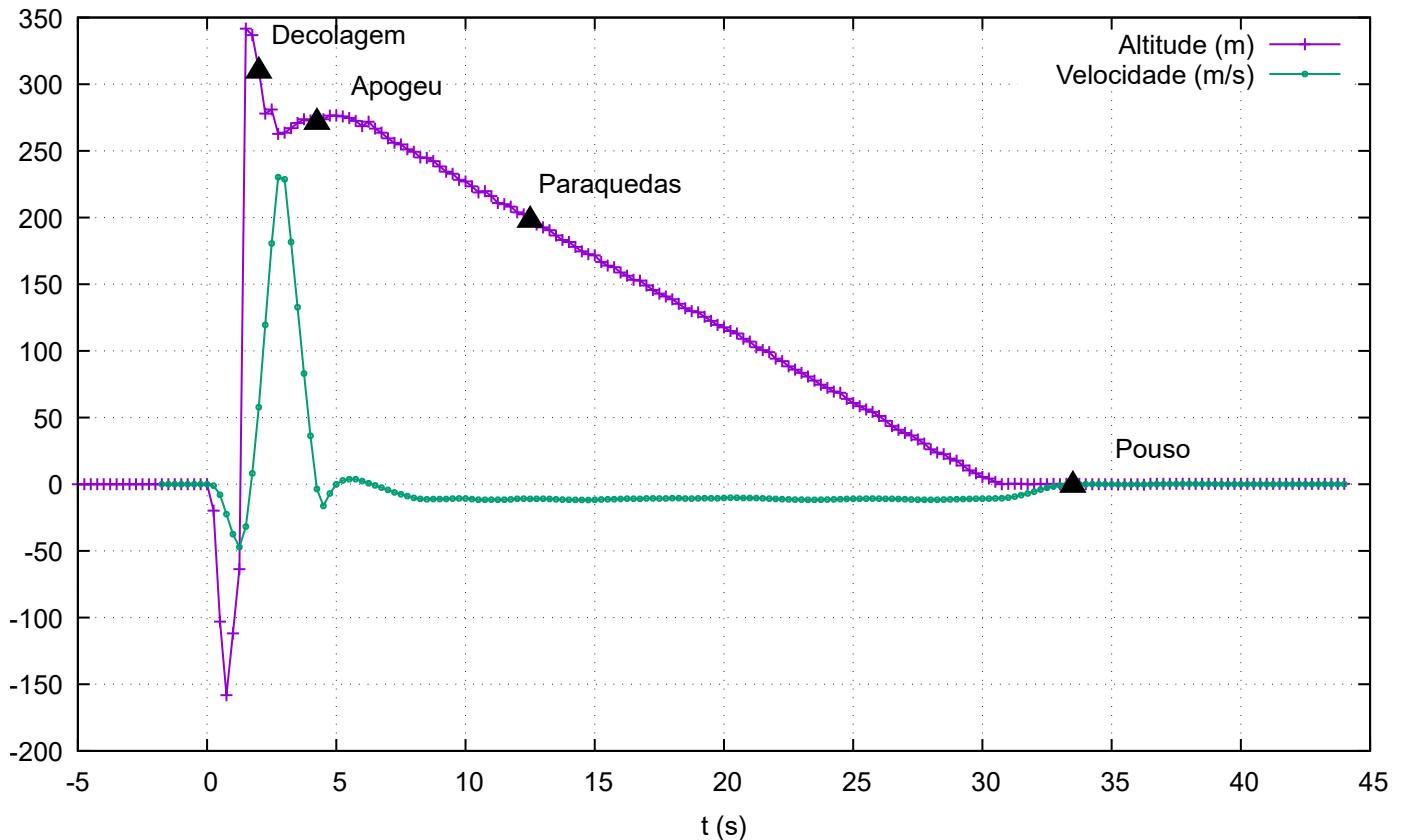


Figura 25: Aplicação dos critérios de detecção de eventos de voo para dados de voos reais. Lançamento 5

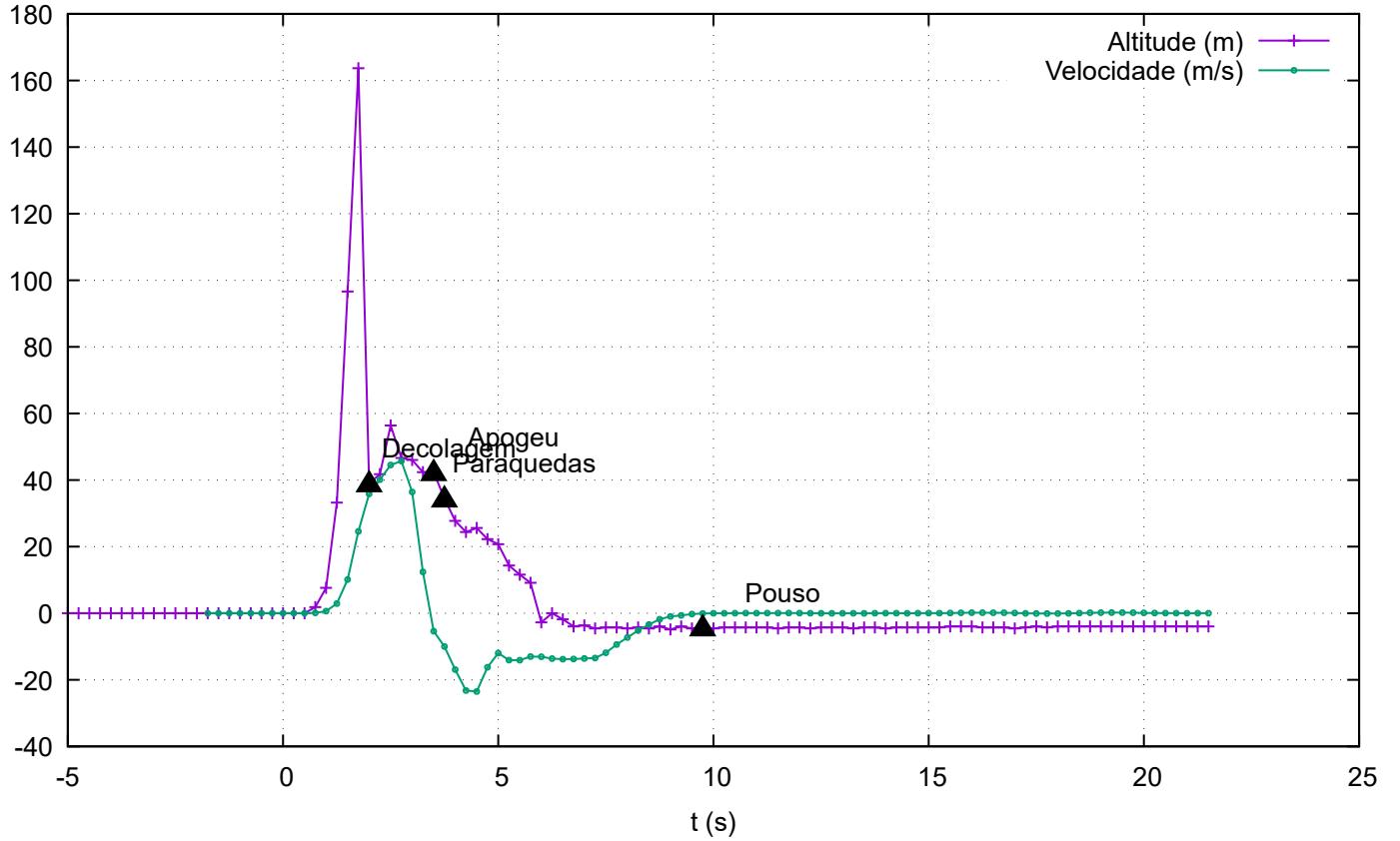


Figura 26: Aplicação dos critérios de detecção de eventos de voo para dados de voos reais. Lançamento 6

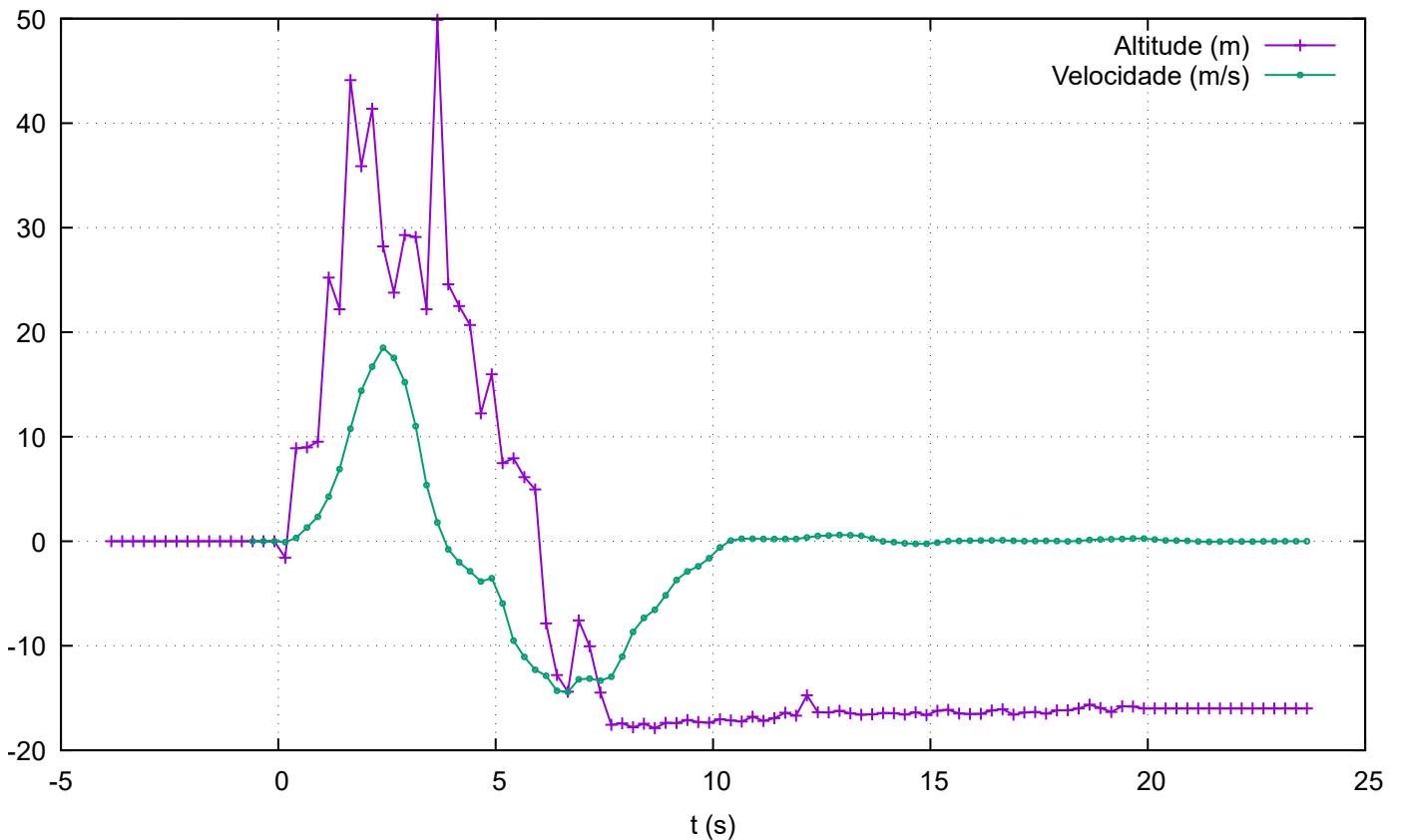


Figura 27: Aplicação dos critérios de detecção de eventos de voo para dados de voos reais. Lançamento 7 ($\bar{v}_{\text{lift-off}} = 30 \text{ m/s}$)

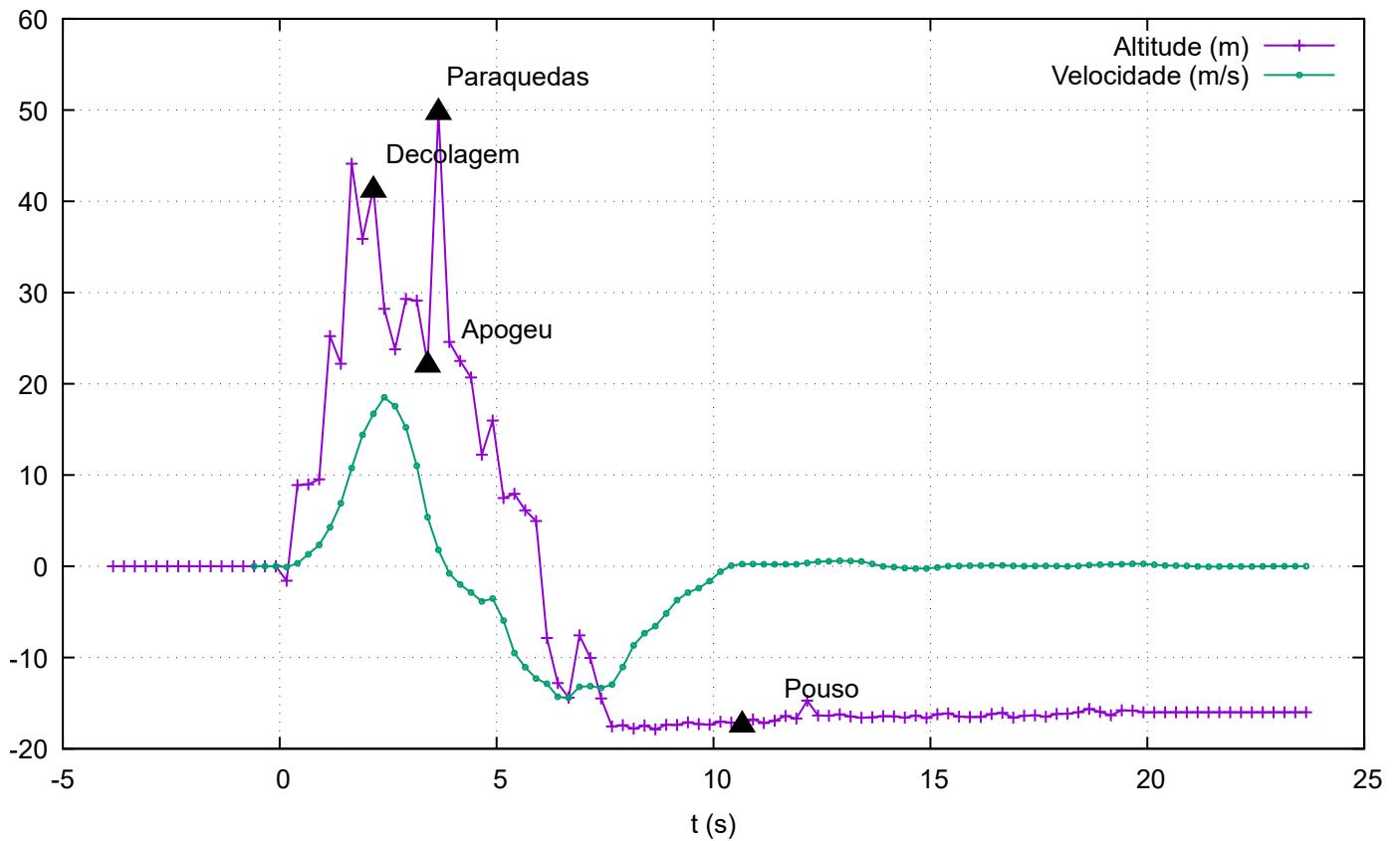


Figura 28: Aplicação dos critérios de detecção de eventos de voo para dados de voos reais. Lançamento 7 ($\bar{v}_{\text{lift-off}} = 15 \text{ m/s}$)

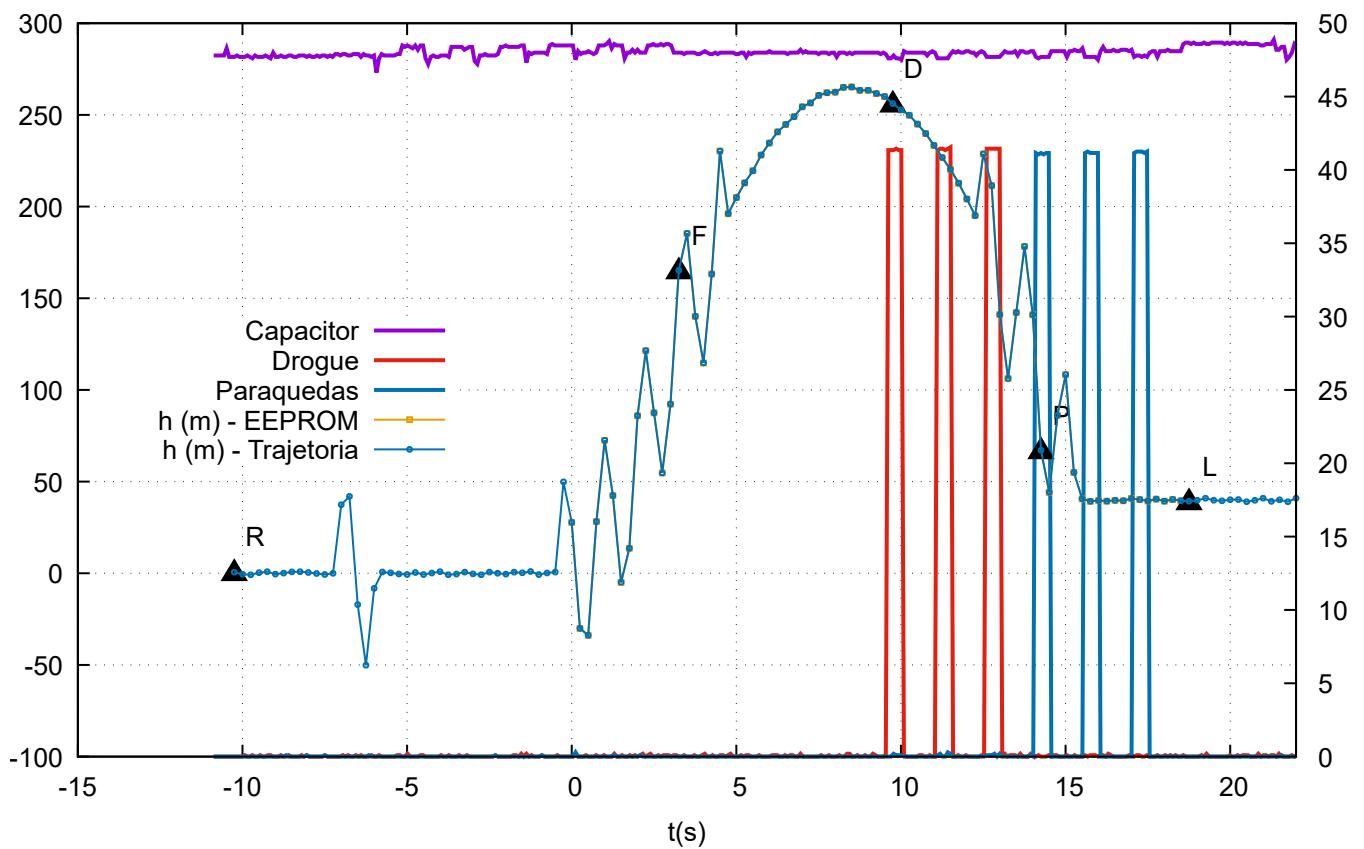


Figura 29: Testes com barômetro simulado. Simulação 1

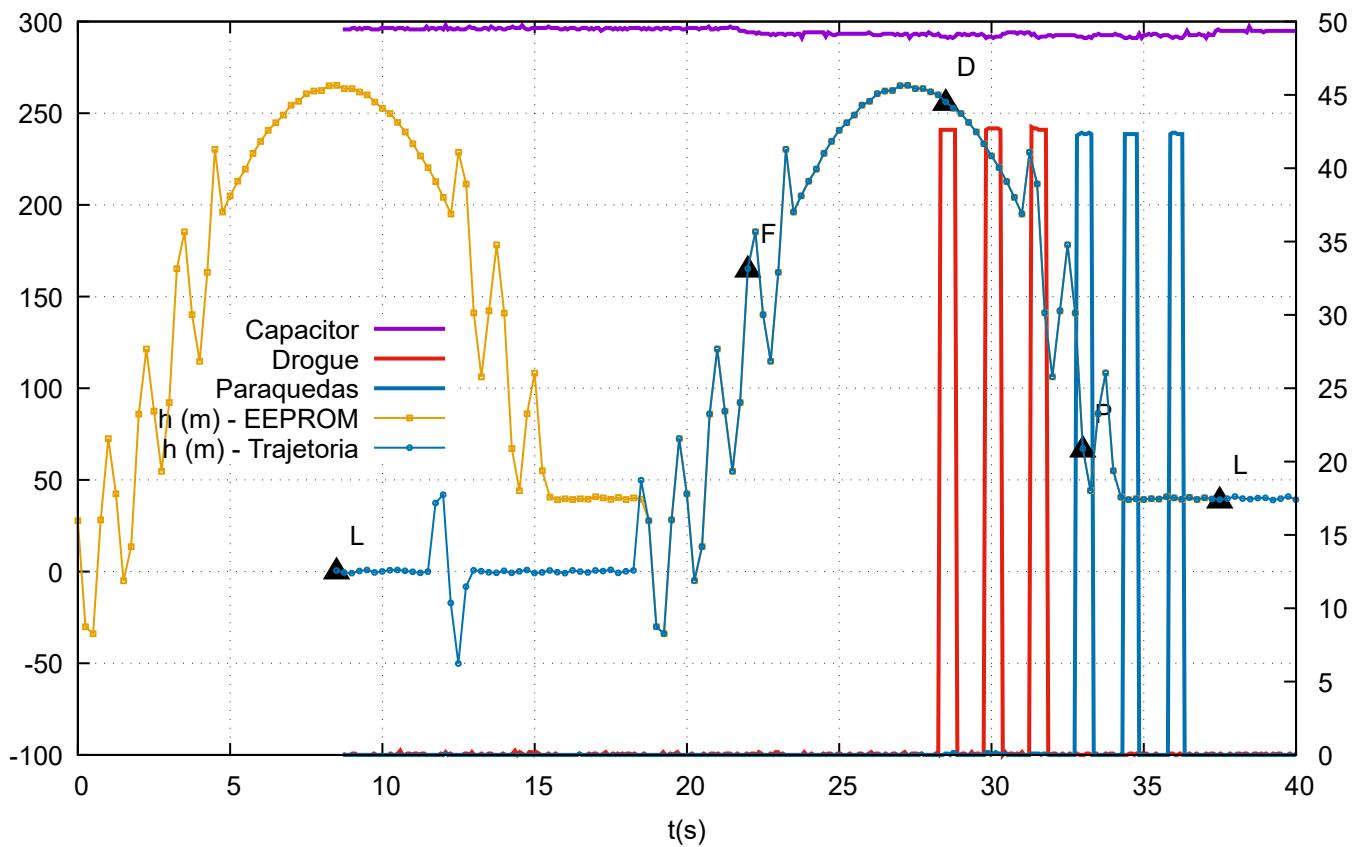


Figura 30: Teste com barômetro simulado. Simulação 2

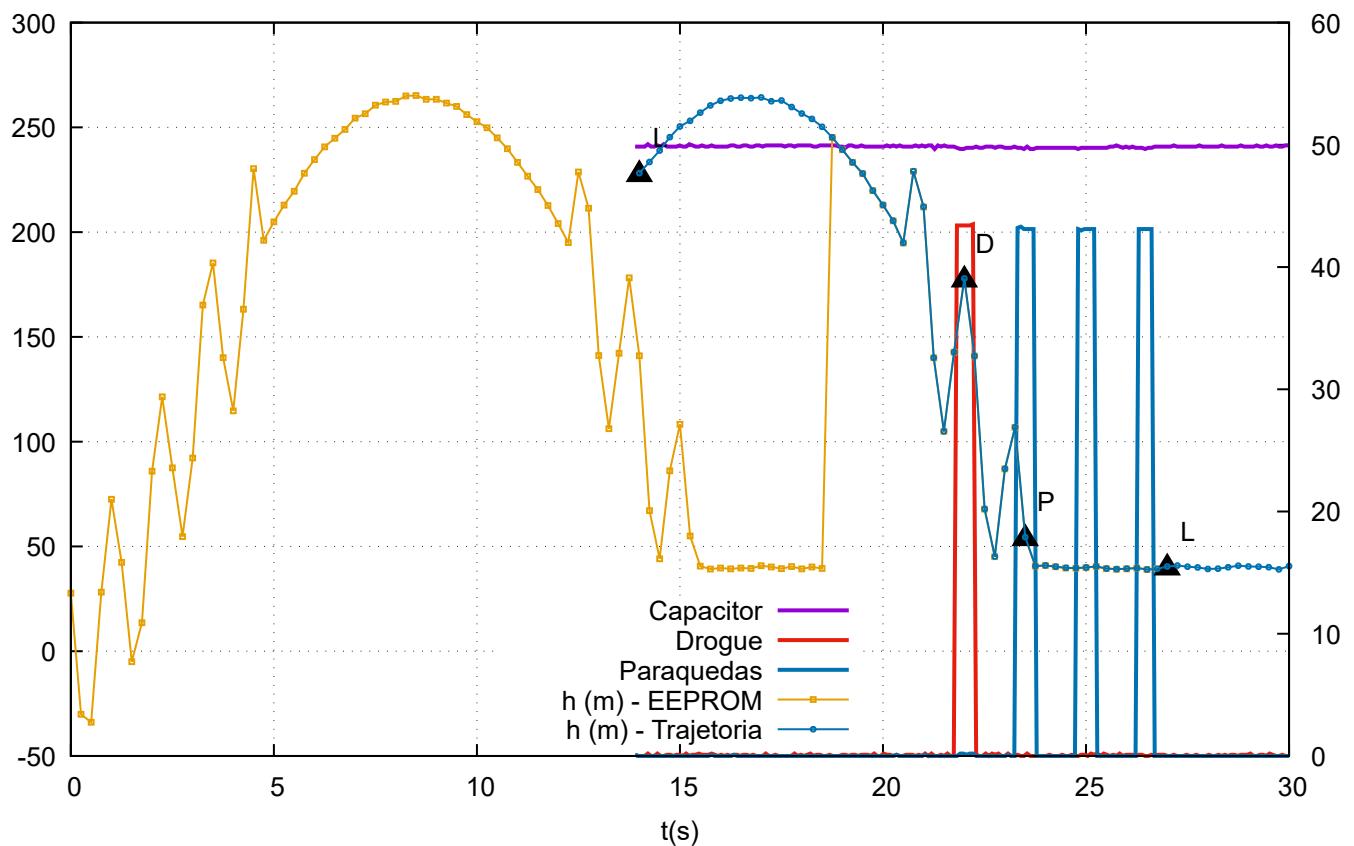


Figura 31: Teste com barômetro simulado. Simulação 3

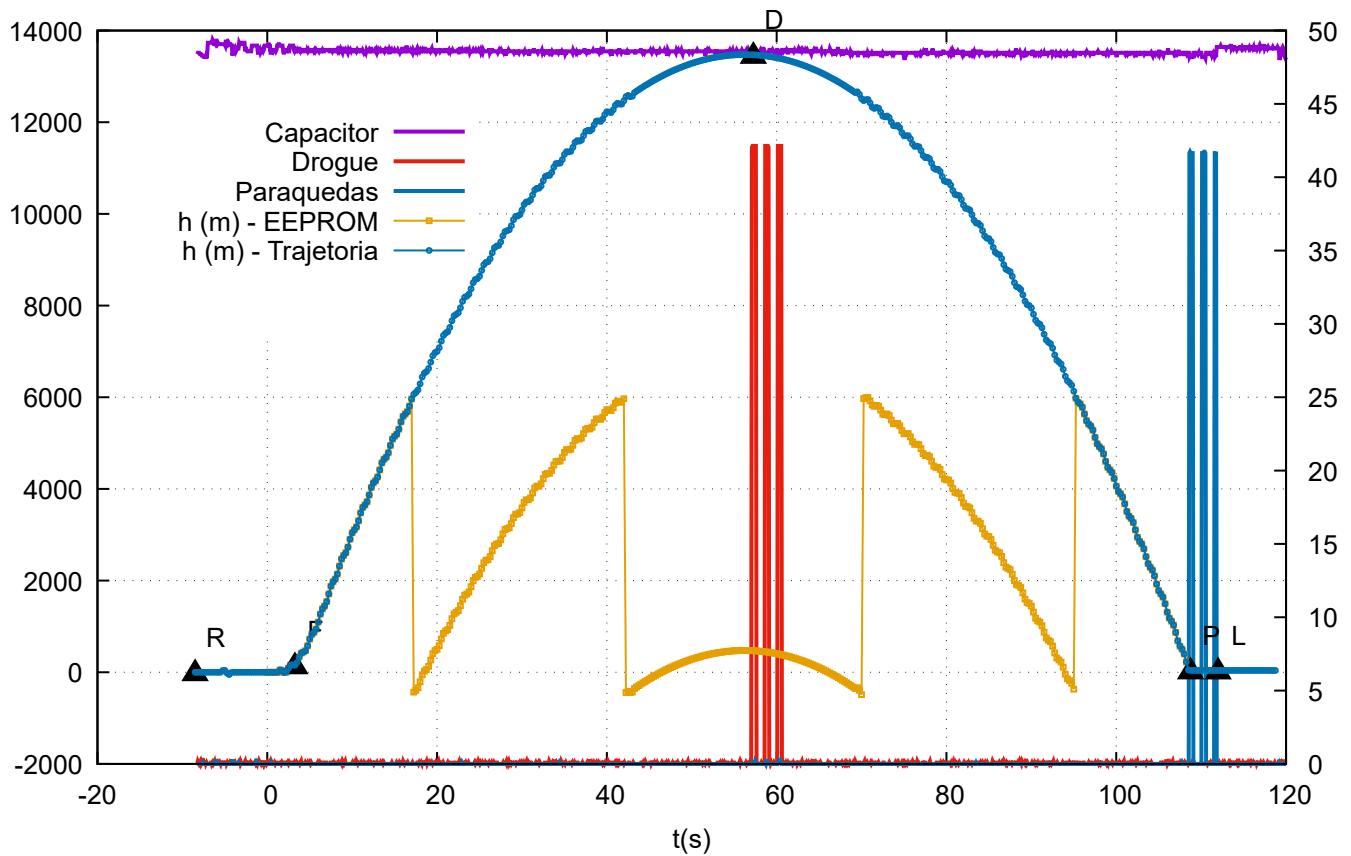


Figura 32: Teste com barômetro simulado. Simulação 4

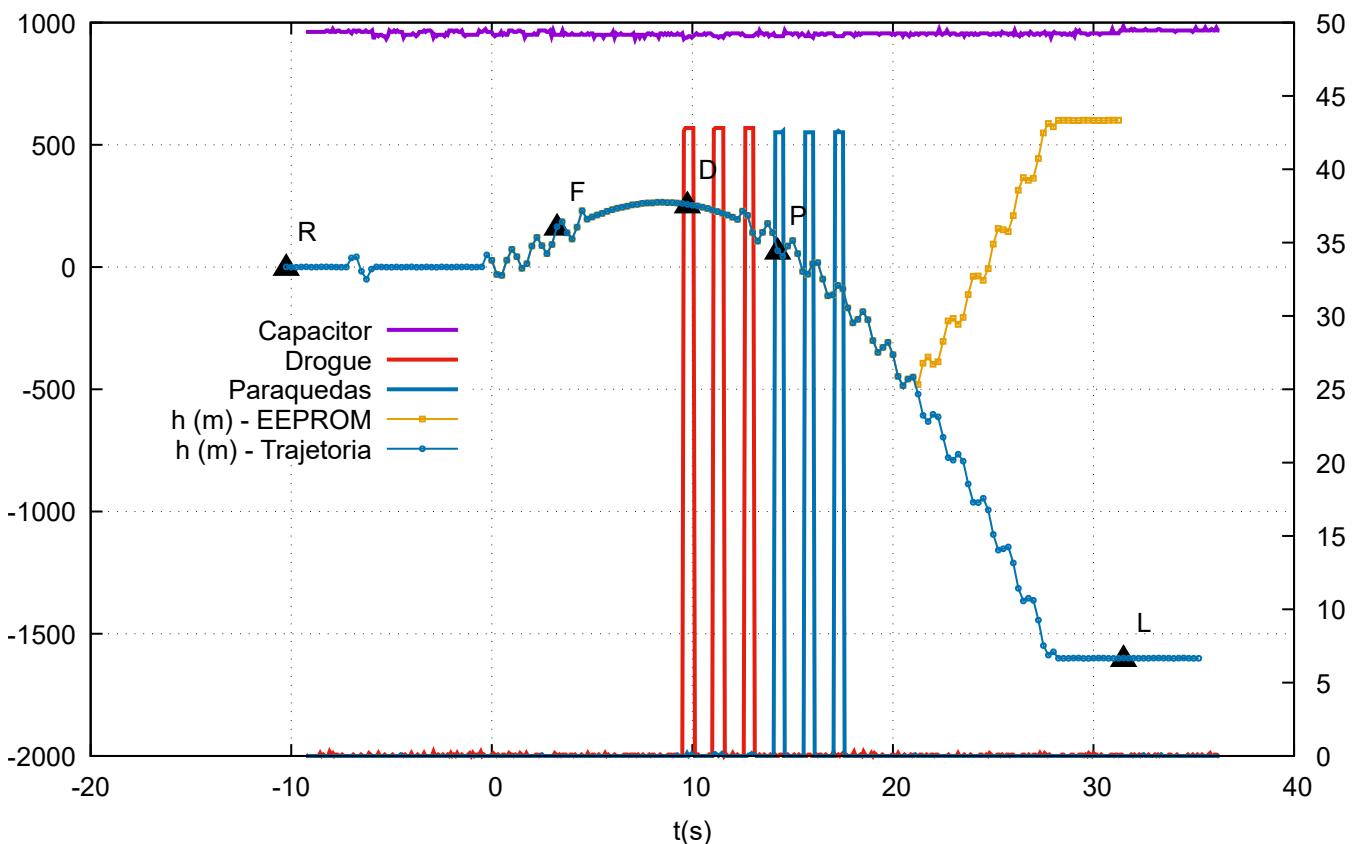


Figura 33: Teste com barômetro simulado. Simulação 5

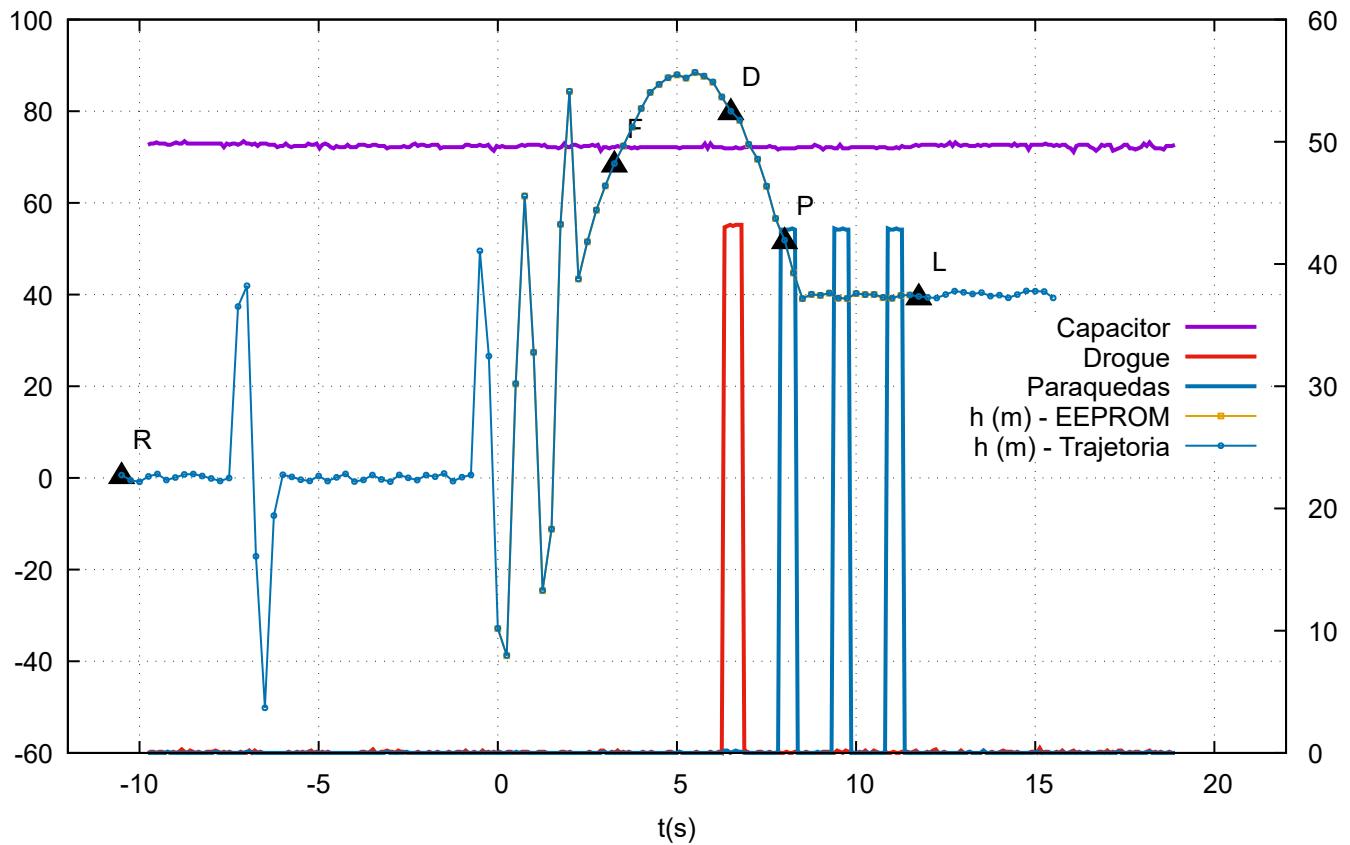


Figura 34: Teste com barômetro simulado. Simulação 6

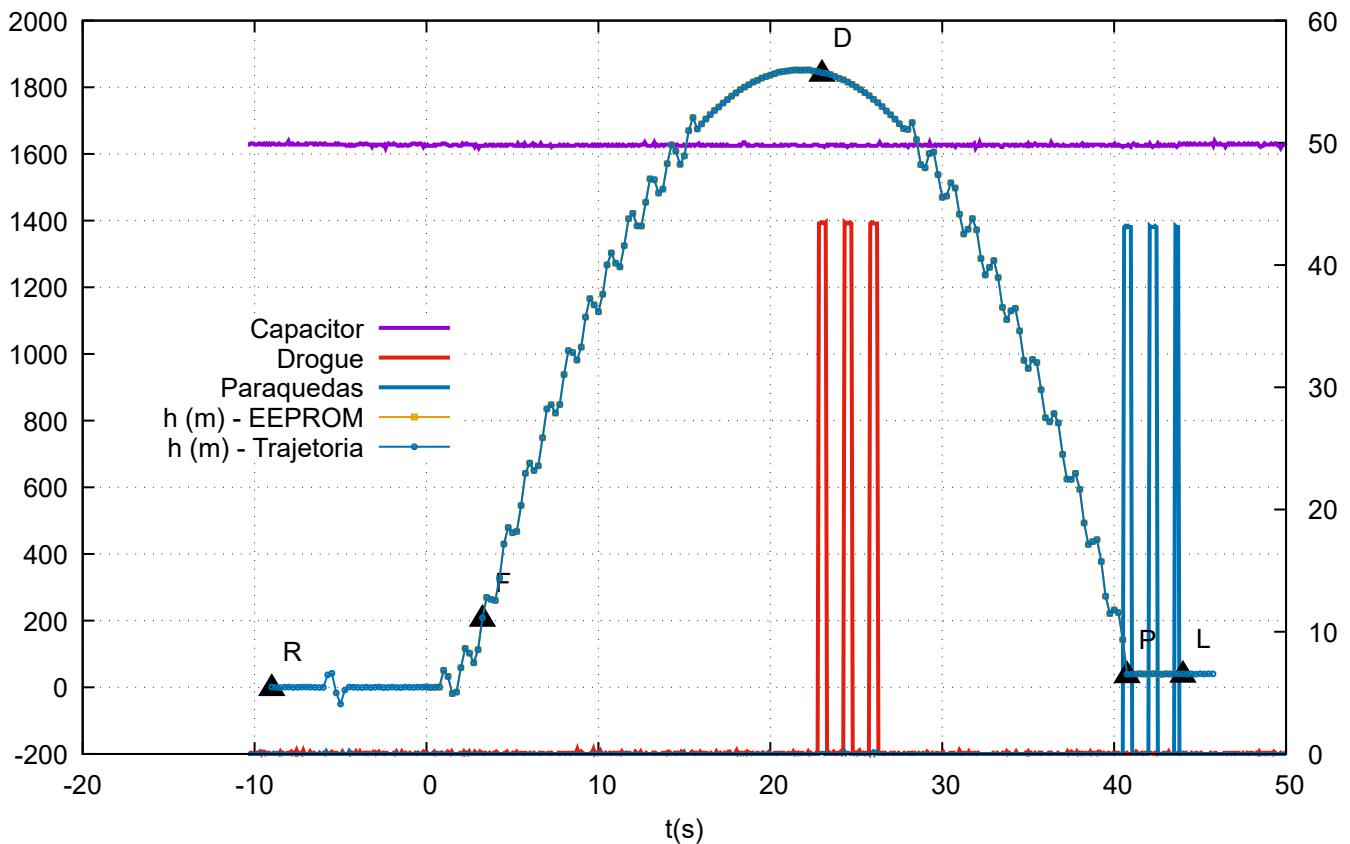


Figura 35: Teste com barômetro simulado. Simulação 7

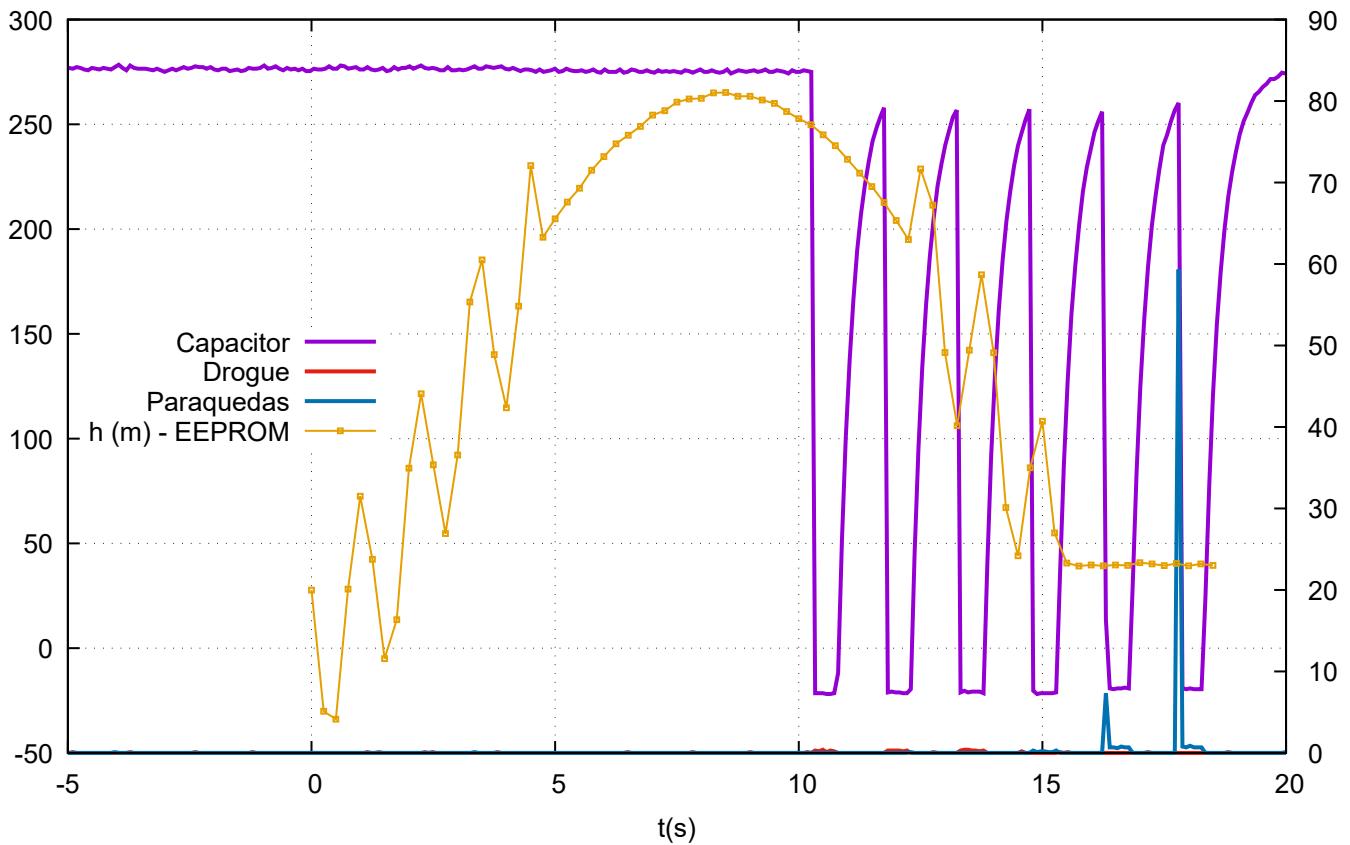


Figura 36: Teste com barômetro simulado. Simulação 8

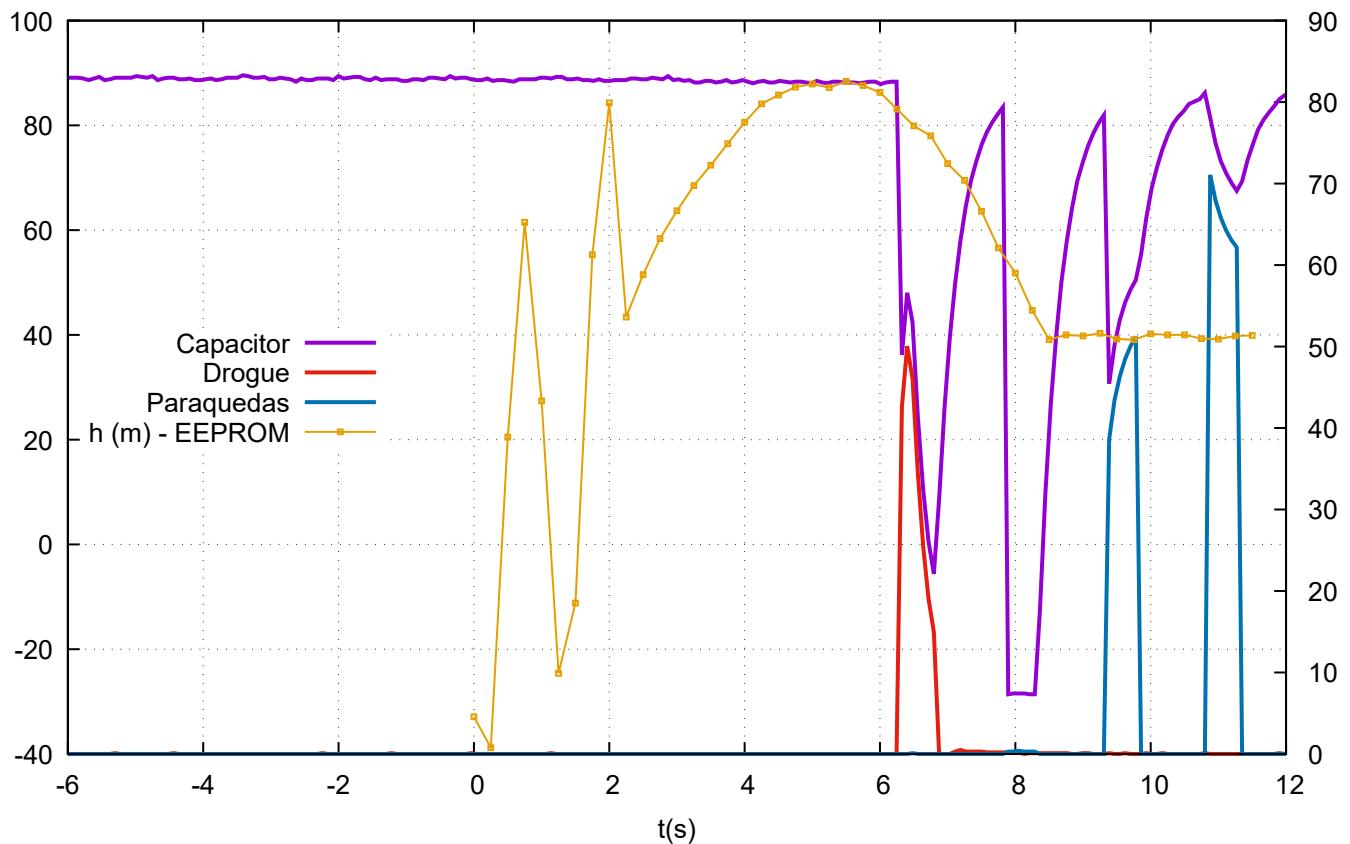


Figura 37: Teste com barômetro simulado. Simulação 9

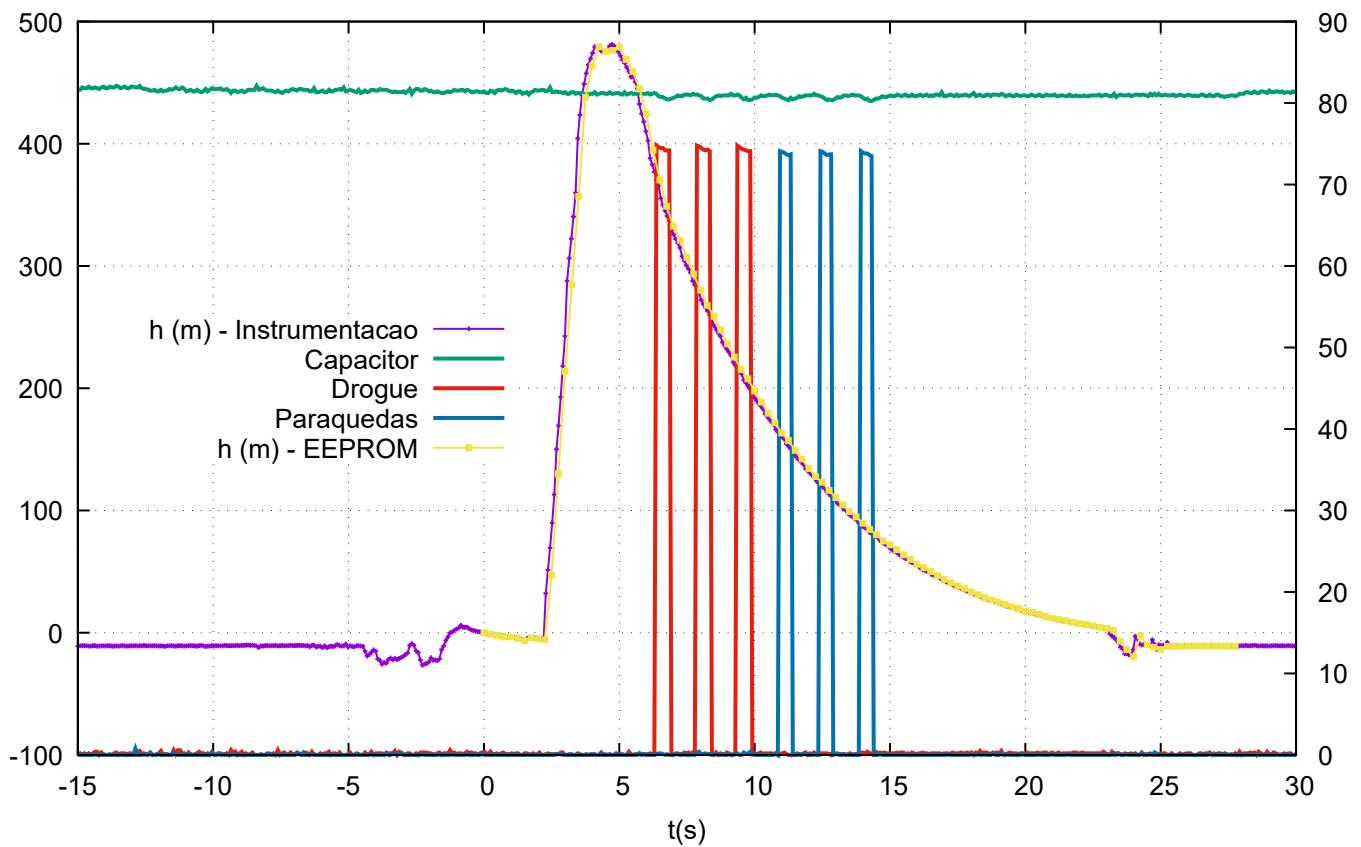


Figura 38: Experimento de bancada 1

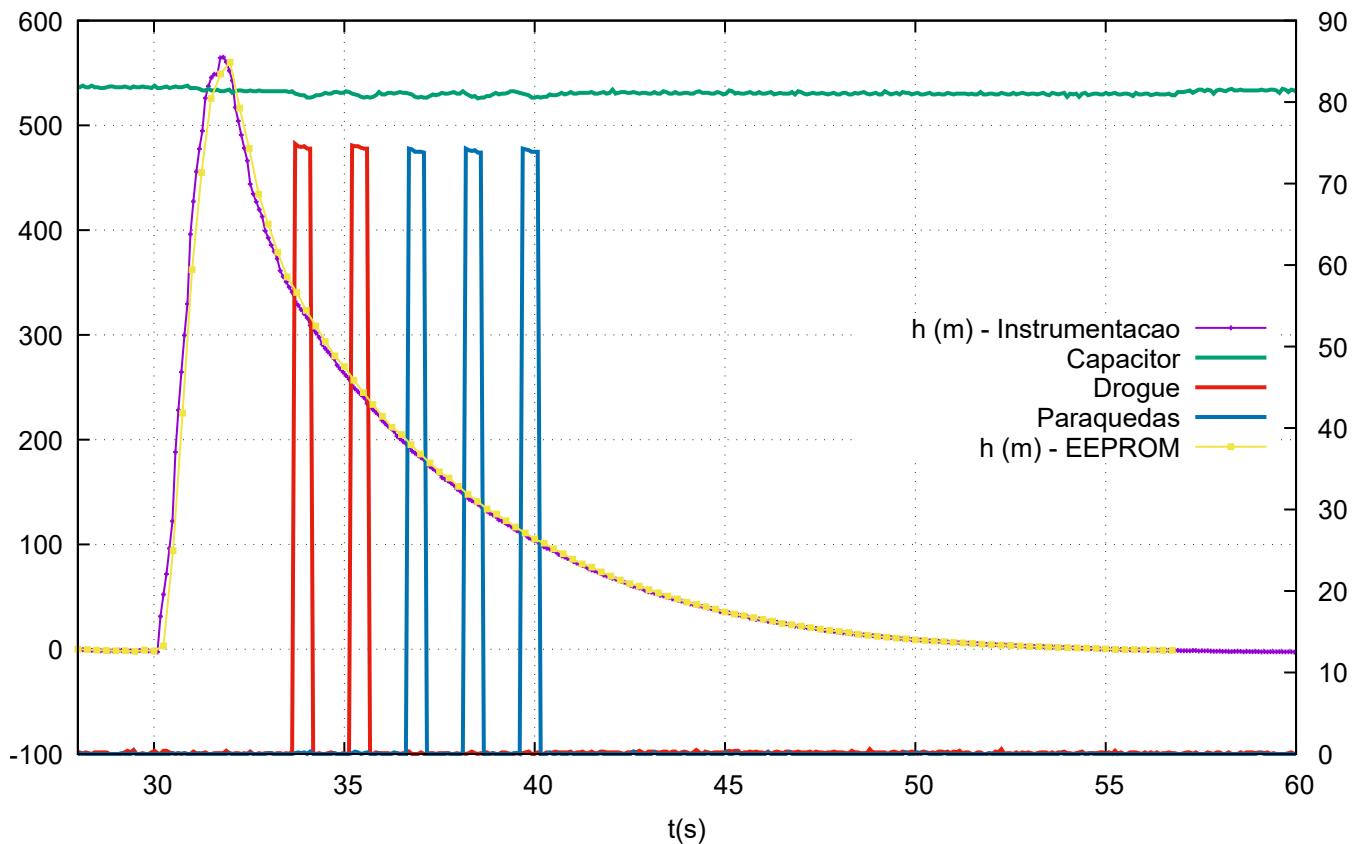


Figura 39: Experimento de bancada 2

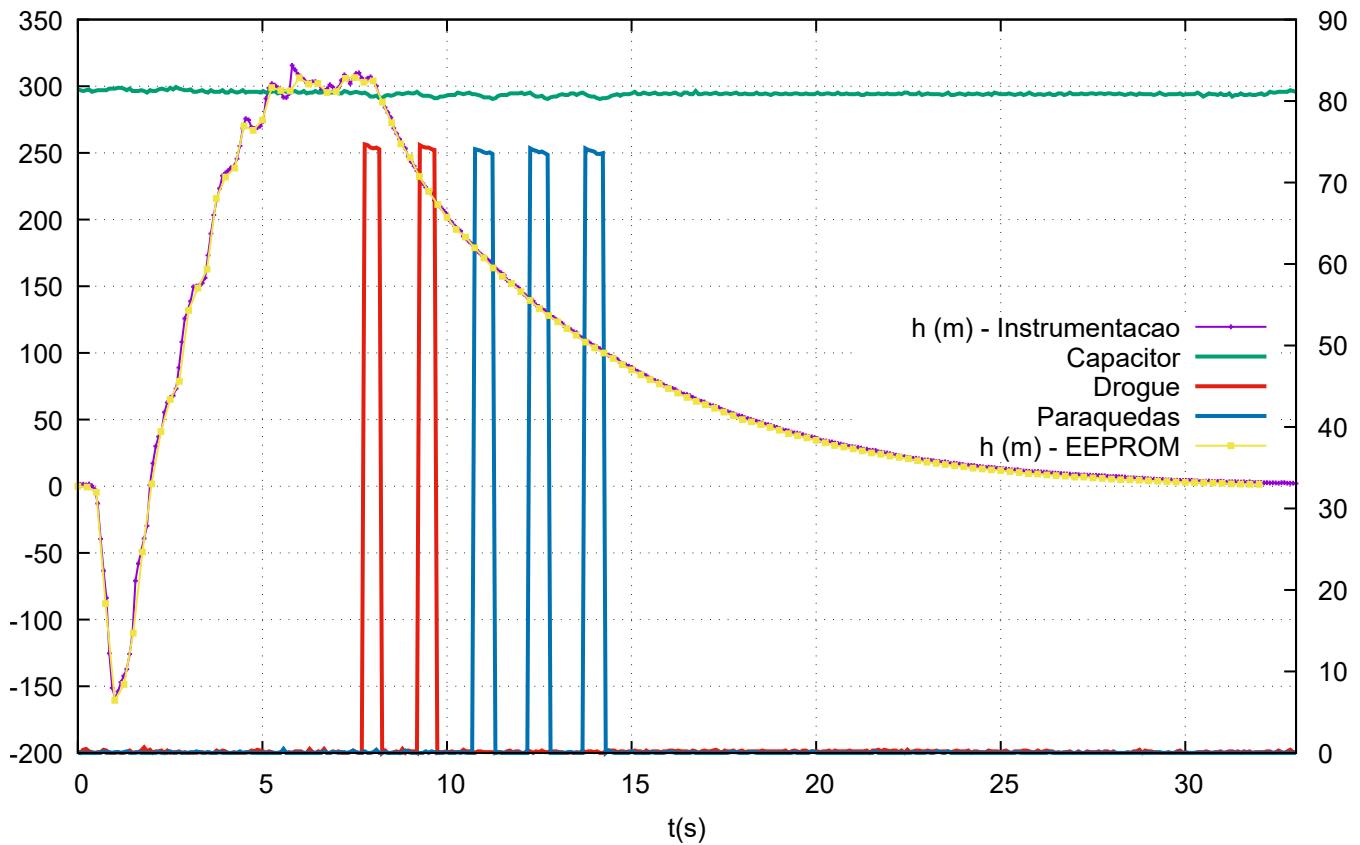


Figura 40: Experimento de bancada 3

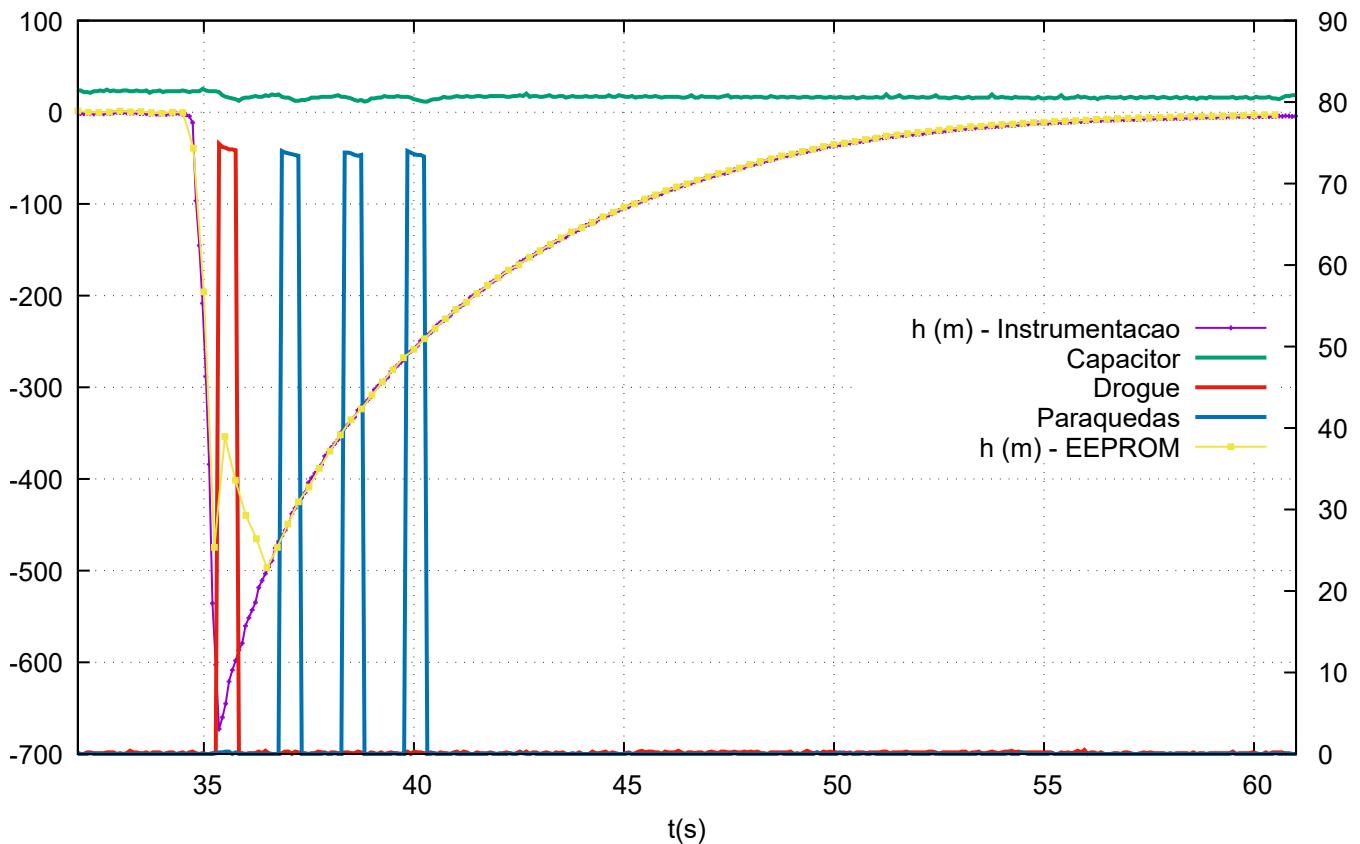


Figura 41: Experimento de bancada 4

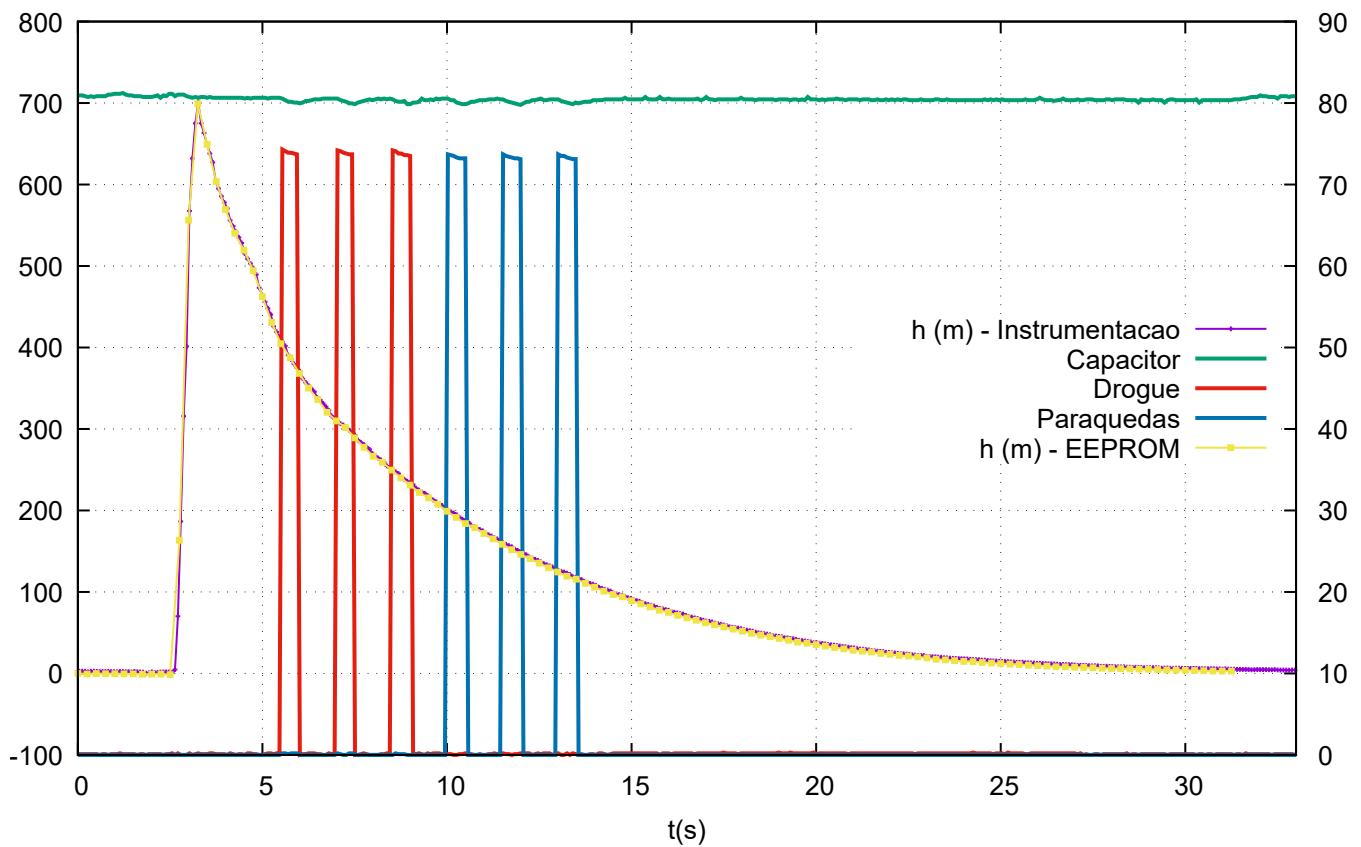


Figura 42: Experimento de bancada 5

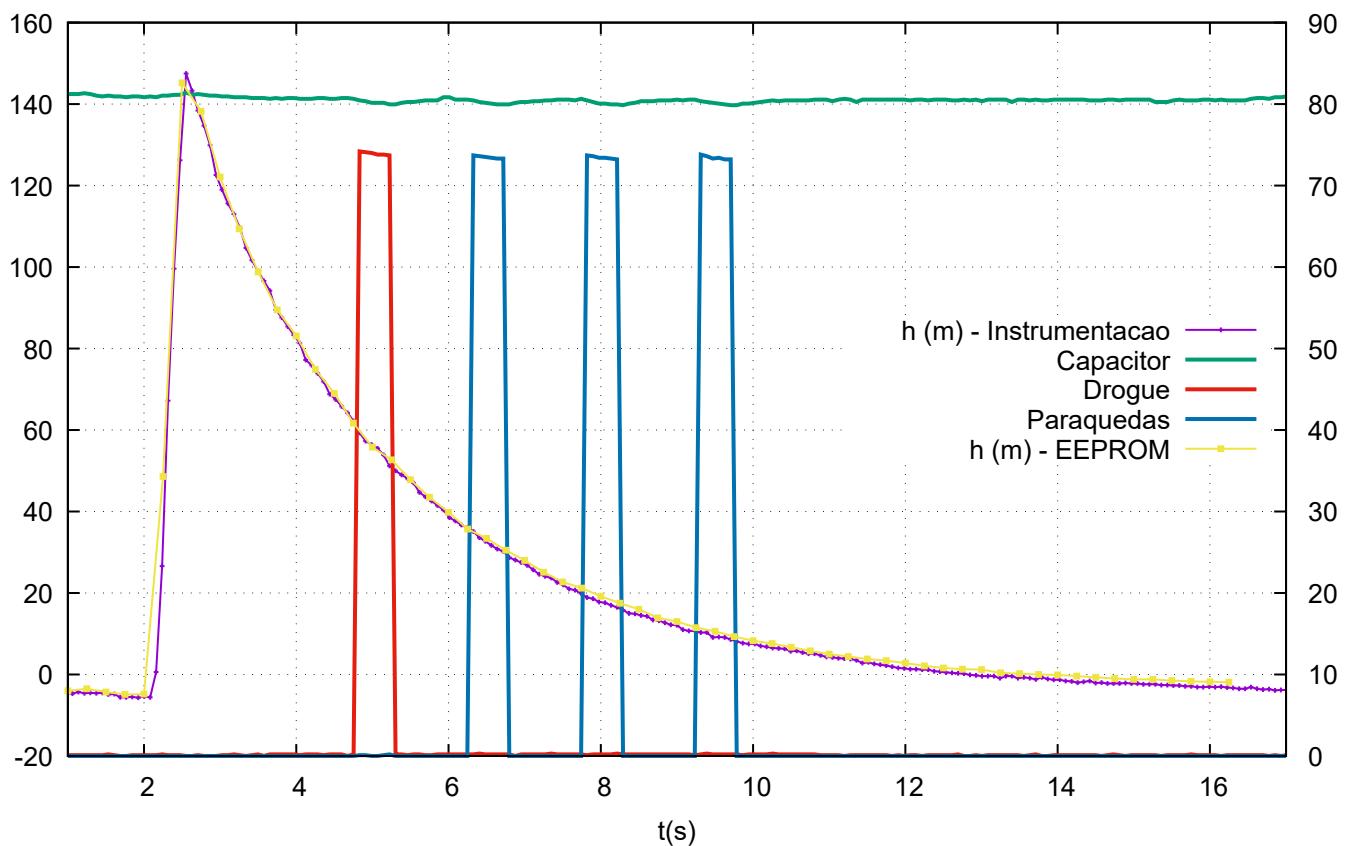


Figura 43: Experimento de bancada 6

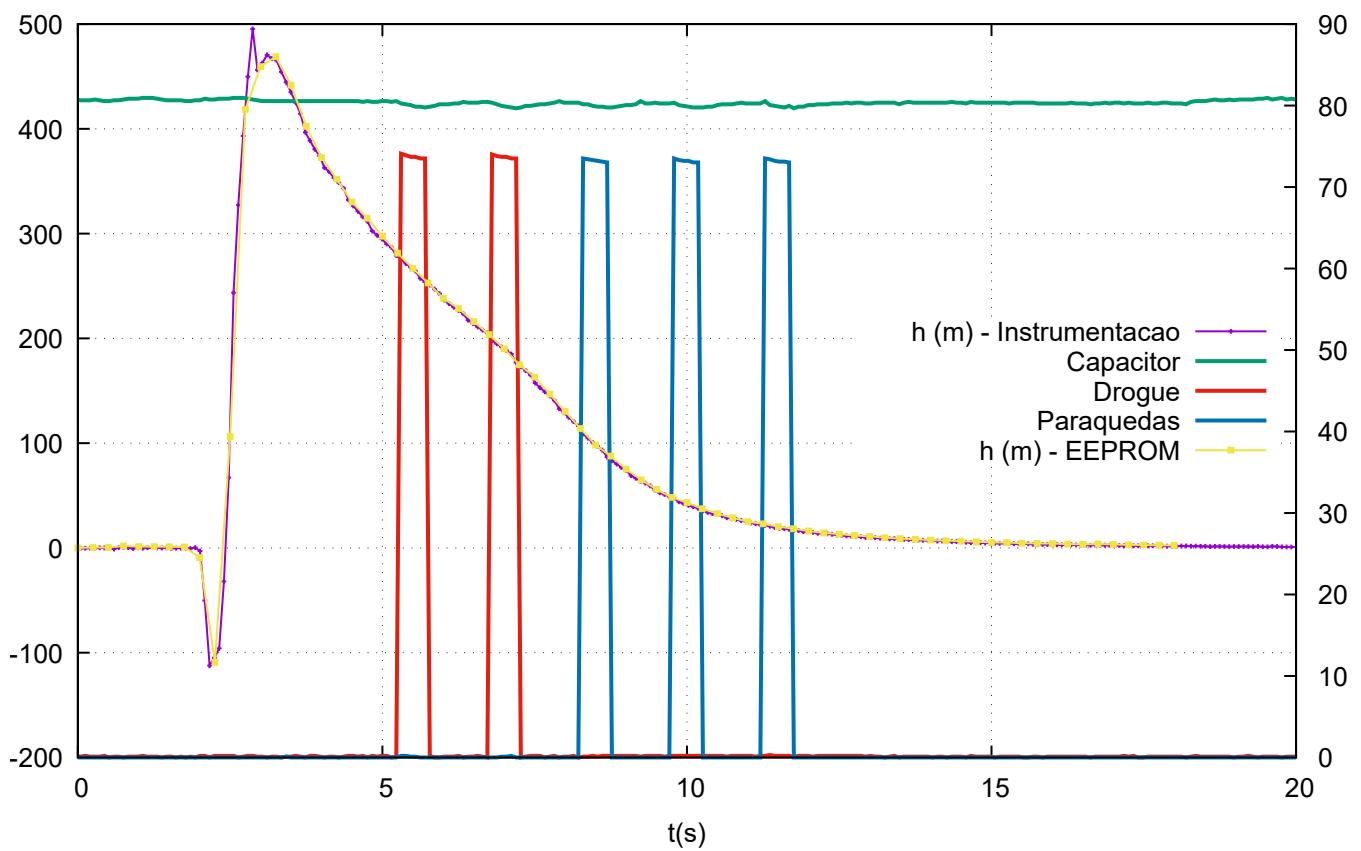


Figura 44: Experimento de bancada 7