

PROJETO FINAL INTEGRADO

Teoria de Controle 2 e Microcontroladores (2023.2)

Alunos: Jonas e Vinícius Bauer

Para realizar o projeto final realizamos um controle discreto utilizando um microcontrolador aplicando a uma ventoinha. Fizemos o uso do kit de desenvolvimento PICGENIOS efetuamos um procedimento para poder ter a obtenção da função de transferência dessa planta em questão. O procedimento envolve a leitura da velocidade da ventoinha ao aplicarmos um degrau, considerando um tempo de amostragem de 100ms. Isso possibilita a obtenção das velocidades em um vetor, permitindo a identificação do seu sistema para a análise da nossa planta. Levando em consideração que estamos considerando como unidade de velocidade pás/100ms obtivemos esses dados e anexamos ao nosso trabalho.

Curva obtida para um período de amostragem $T_s = 100ms$:

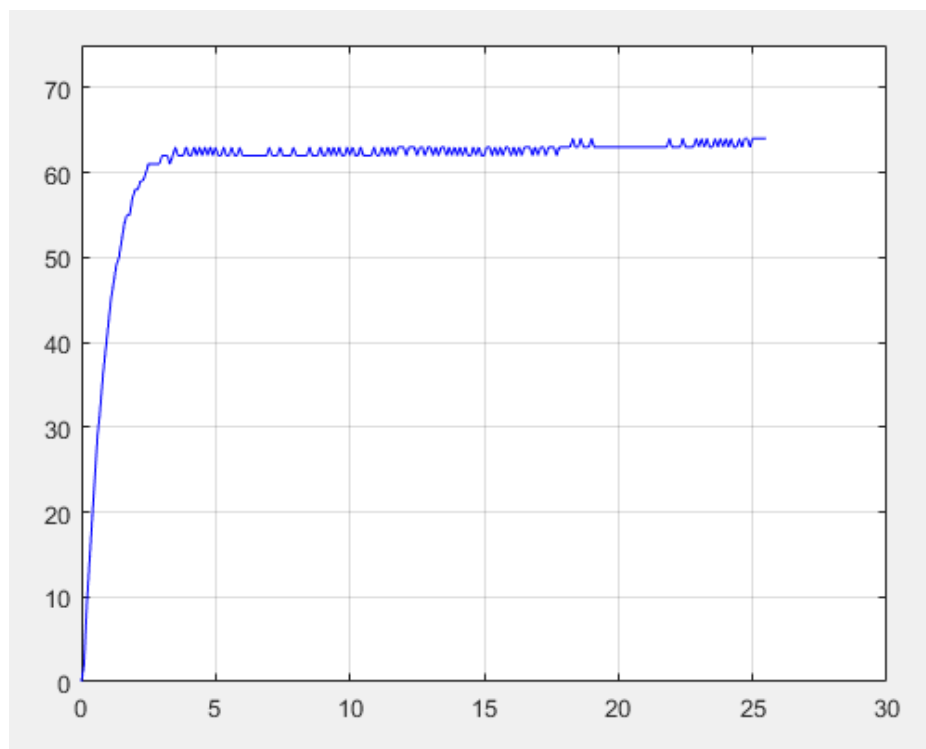


Figura 1 - Plotagem da curva de velocidade.

A partir da imagem anterior é possível determinar a função de transferência de primeira

ordem $G_s = \frac{12.6}{0.85s + 1}$, conforme a imagem abaixo:

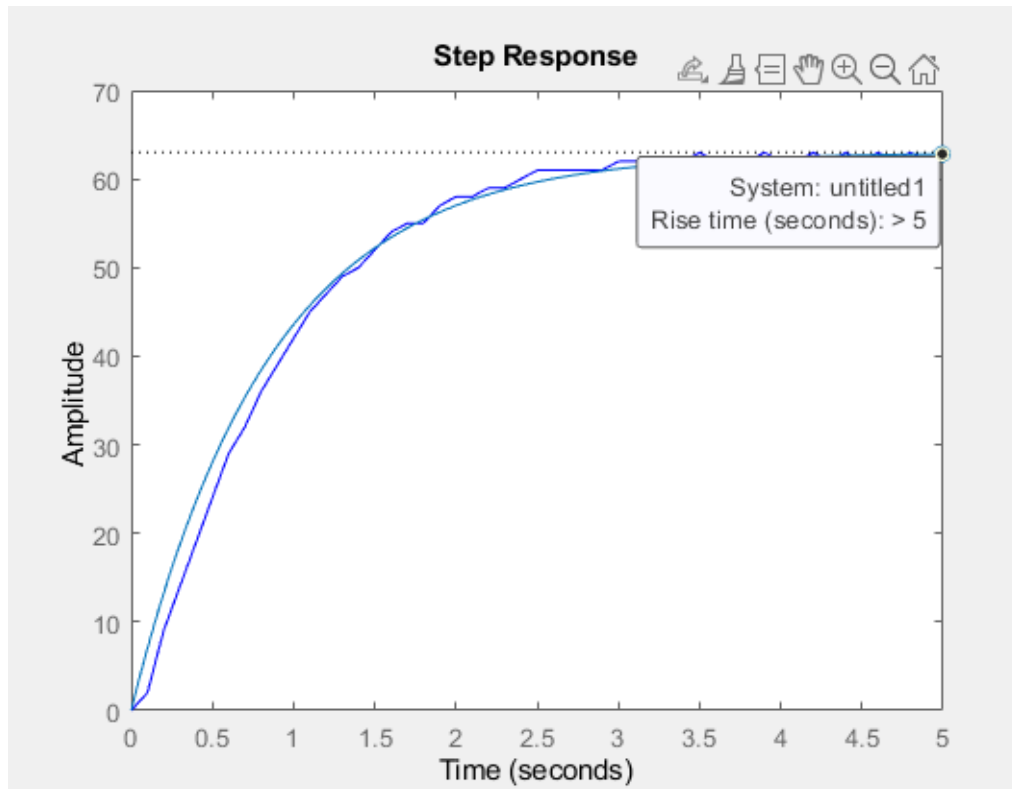


Figura 2 - Comparativo função de transferência.

Com a função de transferência obtida, é possível determinar a função discretizada utilizando o método ZOH para o período de amostragem $T_s = 100ms$:

$$G_z = \frac{1.398}{z - 0.889}$$

Segue abaixo o comparativo das três curvas.

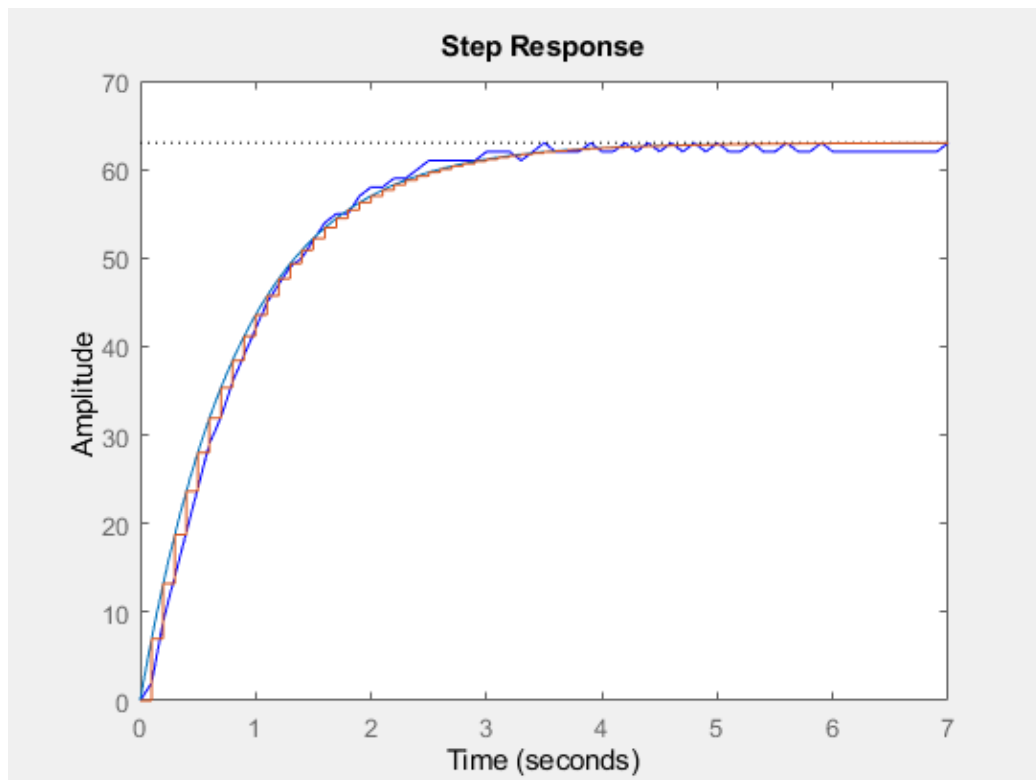


Figura 3 - Comparativo função de transferência discreta.

Para o projeto dos controladores foi testado um controlador Proporcional-integral (PI), o qual caracteriza-se por ter um Zero ajustável e um Polo fixo em $Z=1$. Para este controlador foi possível obter um Settling Time de 2.73 segundos e um Overshoot de 0.942%, conforme a imagem abaixo

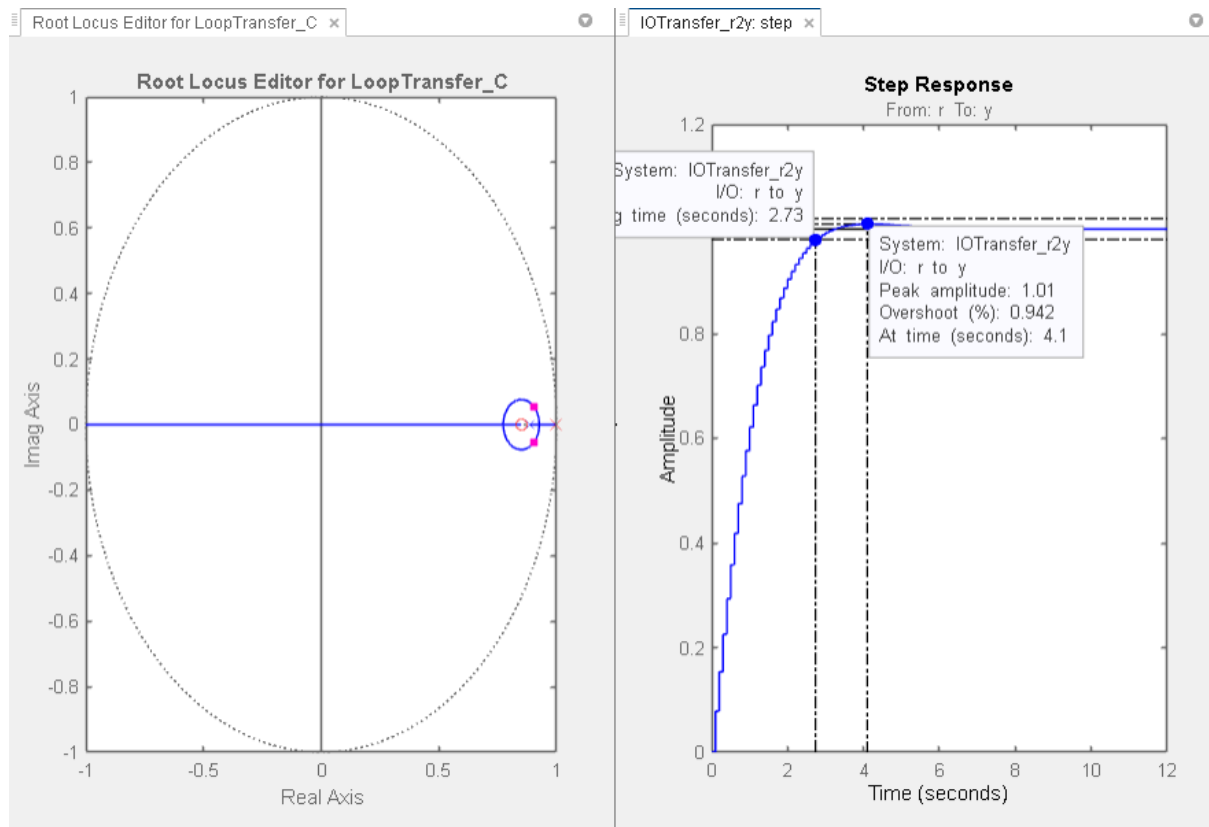


Figura 4 - Parâmetros de desempenho PI.

Com isso, é necessário realizar a equação à diferença para integrar o controlador no projeto.

$$C(z) = \frac{U(z)}{E(z)} = \frac{0.056(z - 0.85)}{z - 1}$$

$$U(z) \cdot (z - 1) = 0.056E(z) \cdot (z - 0.85)$$

Multiplicando por z^{-1} , obtém-se:

$$U(z) - U(z) \cdot z^{-1} = 0.056[E(z) - 0.85E(z) \cdot z^{-1}]$$

$$u_k = u_{k-1} + 0.056(e_k - 0.85e_{k-1})$$

Também foi projetado um controlador PID, o qual caracteriza-se por ter dois Zeros ajustáveis e dois Polos fixos em $Z=1$ e $Z=0$. Para este controlador foi possível obter um Settling Time de 5.39 segundos e um Overshoot de 0.478%, conforme a imagem abaixo

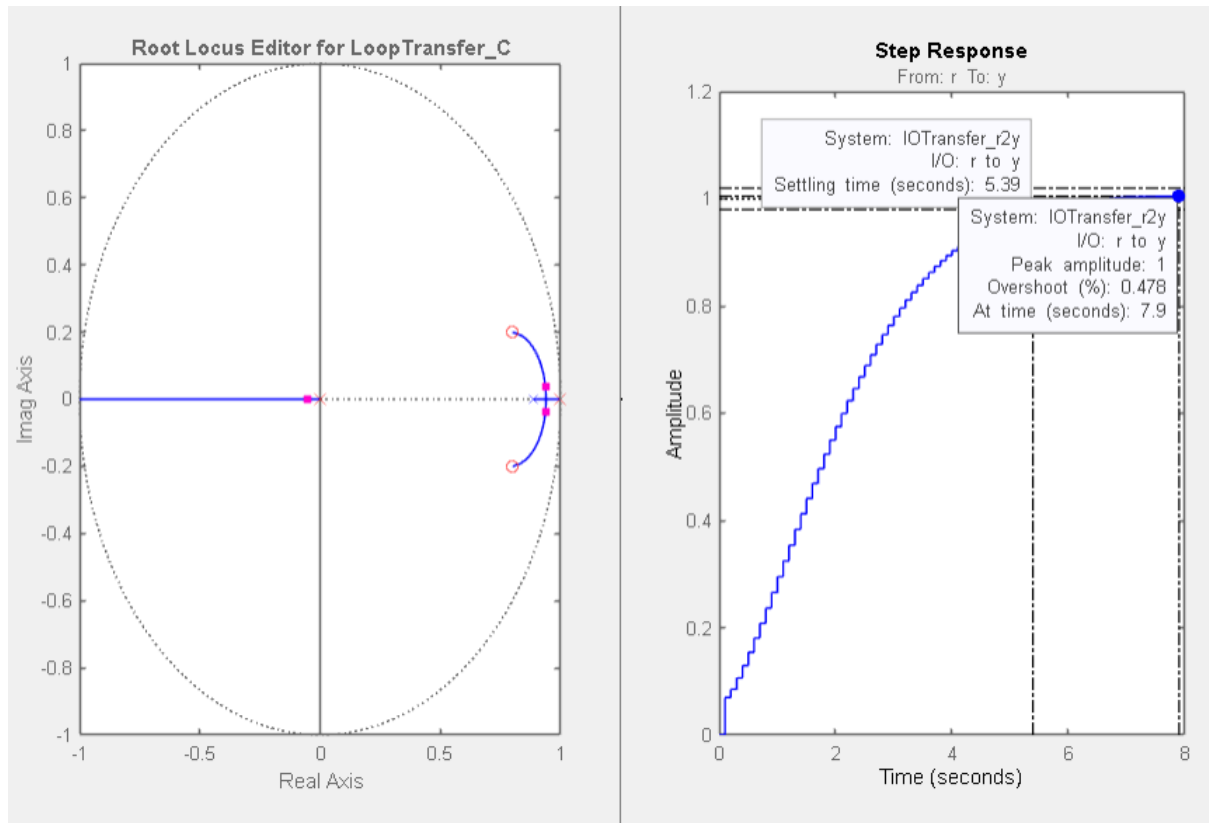


Figura 5 - Parâmetros de desempenho PID.

Equação a diferença:

$$C(z) = \frac{U(z)}{E(z)} = 0.05 \frac{z^2 - 1.6z + 0.68}{z(z - 1)}$$

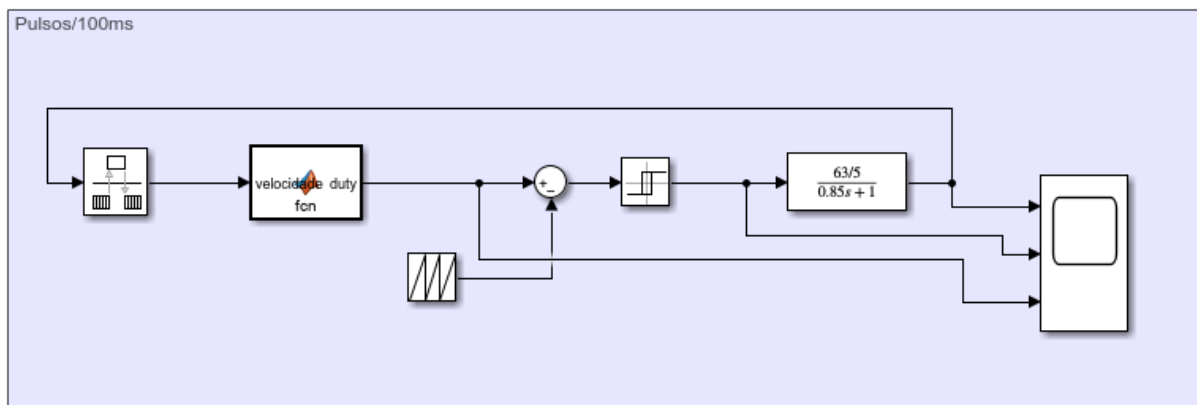
$$U(z) \cdot z(z - 1) = 0.05E(z) \cdot (z^2 - 1.6z + 0.68)$$

Multiplicando por z^{-2} , obtém-se:

$$U(z) - U(z) \cdot z^{-1} = 0.05[E(z) - 1.6E(z) \cdot z^{-1} + 0.68z^{-2}]$$

$$u_k = u_{k-1} + 0.05(e_k - 1.6e_{k-1} + 0.68e_{k-2})$$

Para realizar a validação desse modelo de controle utilizamos blocos para simular através da ferramenta Simulink:



Utilizamos o seguinte script para efetuar o modelo de controle discreto em nossa planta e escolhendo nossa referência:

```
function duty = fcn(velocidade)

persistent ek_1 ek_2 uk_1;

if( isempty(ek_1) ) ek_1=0; end
if( isempty(ek_2) ) ek_2=0; end
if( isempty(uk_1) ) uk_1=0; end

% Erro
vref=40
ek = vref-velocidade

% Controle
% PI
uk = uk_1 + 0.056*(ek - 0.85*ek_1)
% PID
%uk = uk_1 + 0.05*(ek - 1.6*ek_1 + 0.68*ek_2)

% Atraso das variáveis
uk_1=uk
ek_2=ek_1
ek_1=ek

duty = uk*799/5;
```

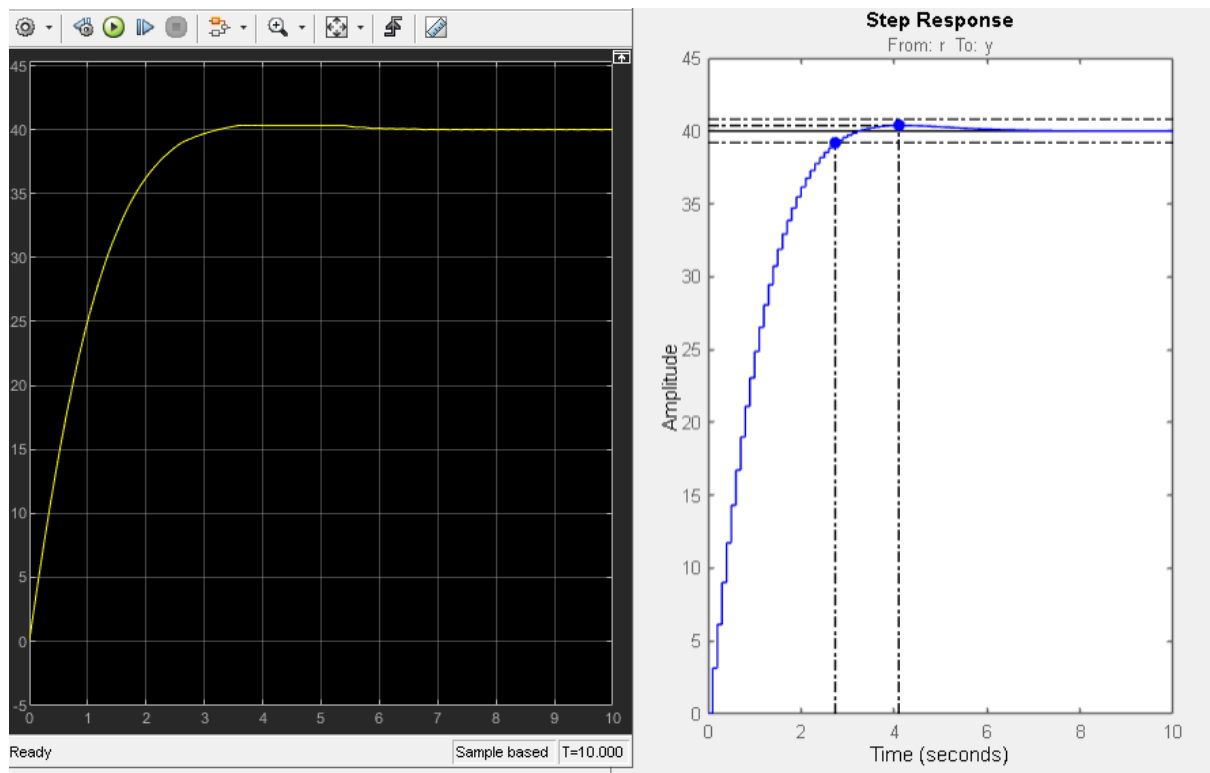


Figura 7 - Comparativo do modelo no Simulink e Sisotool (PI).

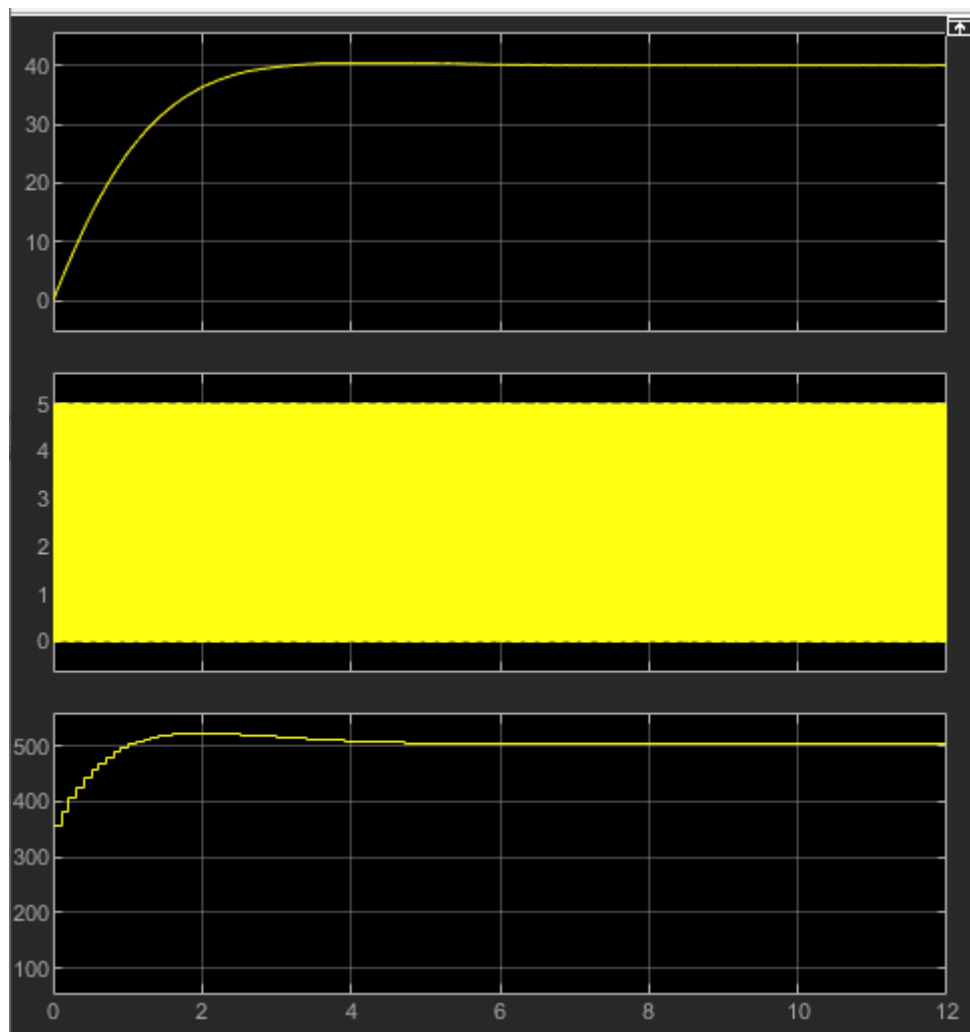


Figura 8 - Respostas ao controle PI: Velocidade, PWM e u_k .

Observando a resposta, podemos verificar a validação do controle proporcional-integral (PI) ao comparar o modelo do controlador, elaborado por meio da ferramenta Sisotool, com a implementação do controlador por meio de um bloco de código no Simulink, no qual normalizamos o esforço de controle para simulação. Além disso, constatamos que o esforço de controle é compatível com a referência estabelecida pelo sistema de controle, e o mesmo estabiliza o esforço quando atingimos o valor predefinido pelo script.

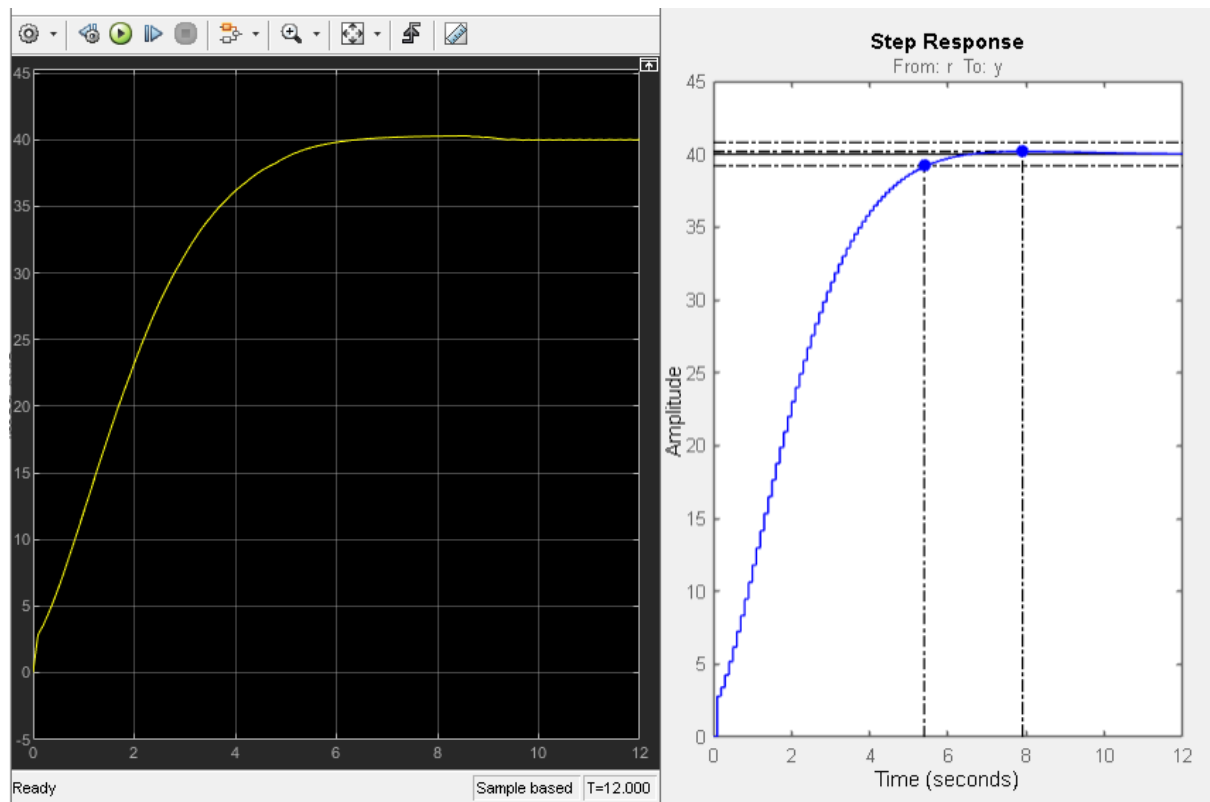


Figura 9 - Comparativo modelo no Simulink e Sisotool (PID).

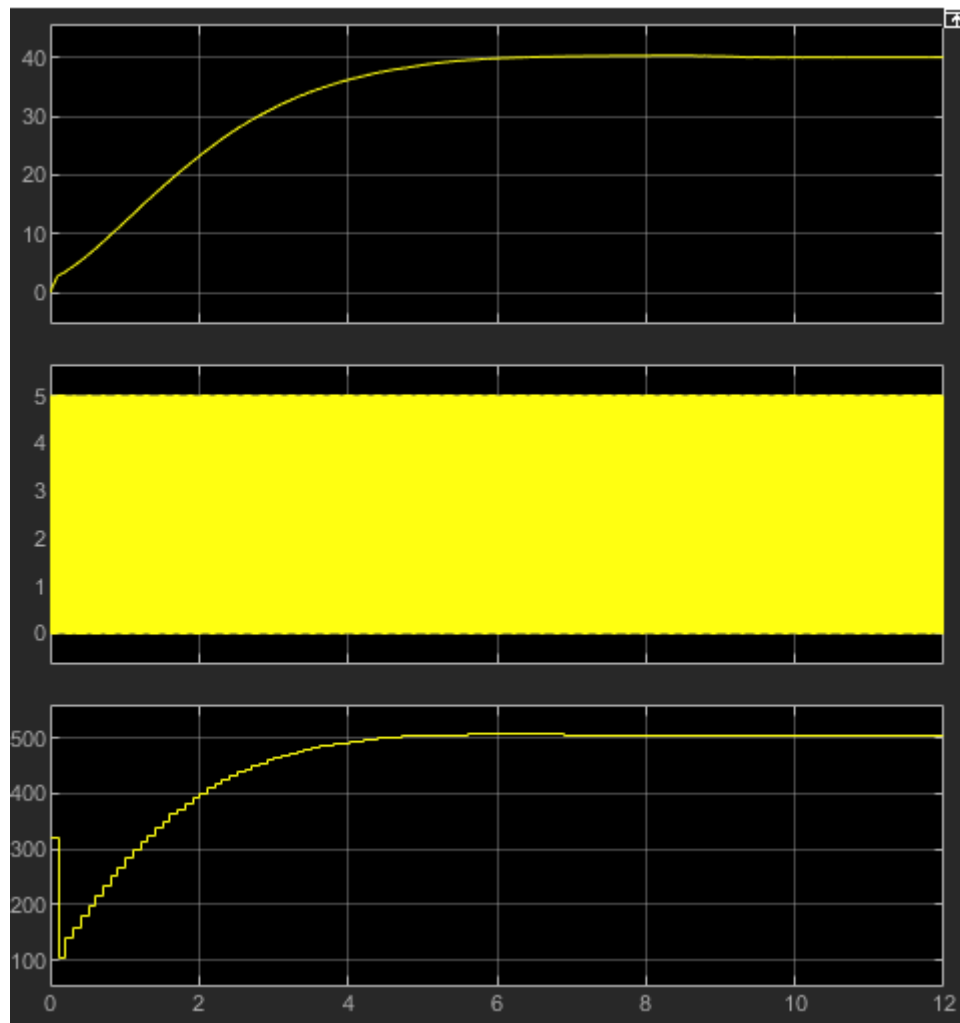


Figura 8 - Respostas ao controle PID: Velocidade, PWM e u_k .

Ao analisar as respostas, podemos também realizar validação do controle proporcional-integral-derivativo (PID) ao comparar o modelo do controlador, elaborado por meio da ferramenta Sisotool, com a implementação do controlador por meio de um bloco de código no Simulink. Dessa forma, ao verificarmos as respostas possuímos um modelo satisfatório de controle discreto que podemos implementar em um microcontrolador com a finalidade de realizar o controle de velocidade da ventoinha que estará atuando.