



Prj1CDig - Controle PI Anti-Windup de Processo Térmico

Objetivo:

Projetar um controlador PI anti-windup discreto utilizando o LGR. Verificar em simulação e experimentalmente o efeito da saturação do atuador sobre o atendimento das especificações de projeto. Observar a habilidade do controlador PI para rejeitar perturbações constantes e senoidais.

Visão Geral

Prj1CDig

1. Providenciar o material necessário:
Protoboard, ESP32, 2 x LM35, 2 x BC548, Cooler pequeno, 3 x Resistores 50 Ω 1/4 W, R e C filtro Passa-Baixas
2. Medir a resposta em Malha Aberta para sinal PRBS, tipicamente 8h de coleta (prbs.m)
3. Identificar os parâmetros do processo: K_p , T_z , T_{p1} , T_{p2} e T_d (procest.m)
4. Projetar um controlador PI contínuo no LGR, simular no simulink, considerando saturação 0-12V e Anti-Windup.
5. Escolher uma taxa de amostragem adequada. Implementar o controlador PI Anti-Windup digital no ESP32.
6. Verificar se as especificações do projeto estão sendo atendidas. Voltar ao passo 4 ou 5, caso necessário.
7. Gravar video de 5 a 7 minutos, apresentando o projeto de forma objetiva.

Prj2CDig

8. Simular o observador de perturbações senoidais contínuo com realimentação de estados e canal I-Anti-Windup.
9. Implementar o Prj2CDig no ESP32.
10. Verificar se as especificações do projeto estão sendo atendidas. Voltar ao passo 4 ou 8, caso necessário.
11. Gravar video de 5 a 7 minutos, apresentando o projeto Prj2CDig de forma objetiva.
12. Apresentação presencial.

1. O processo térmico com perturbação

O processo térmico utilizado no laboratório de controle digital é constituído de um sensor de temperatura LM35, um conjunto de resistores e transistores acionados via PWM por um microcontrolador ESP32 com conversor A/D, conforme mostrado na figura 4. Um amplificador operacional pode ser utilizado para ajustar a faixa dinâmica do sensor de temperatura à faixa de leitura do ADC (LM35 $10\text{mV}/^{\circ}\text{C} \Rightarrow 100^{\circ}\text{C} \rightarrow 1\text{ V}$. ESP32 trabalha com $3,3\text{V}$). Um pequeno ventilador (cooler de processador, e.g.) permite gerar perturbações constantes e/ou senoidais, resfriando o processo. É importante fixar o ventilador (fita isolante, cola etc.) em relação ao LM35 para que não ocorram variações significativas dos parâmetros do processo. A temperatura ambiente deve ser medida “longe” da perturbação. Filtros passa-baixa, conforme o data-sheet do LM35, reduzem significativamente o ruído de medida.

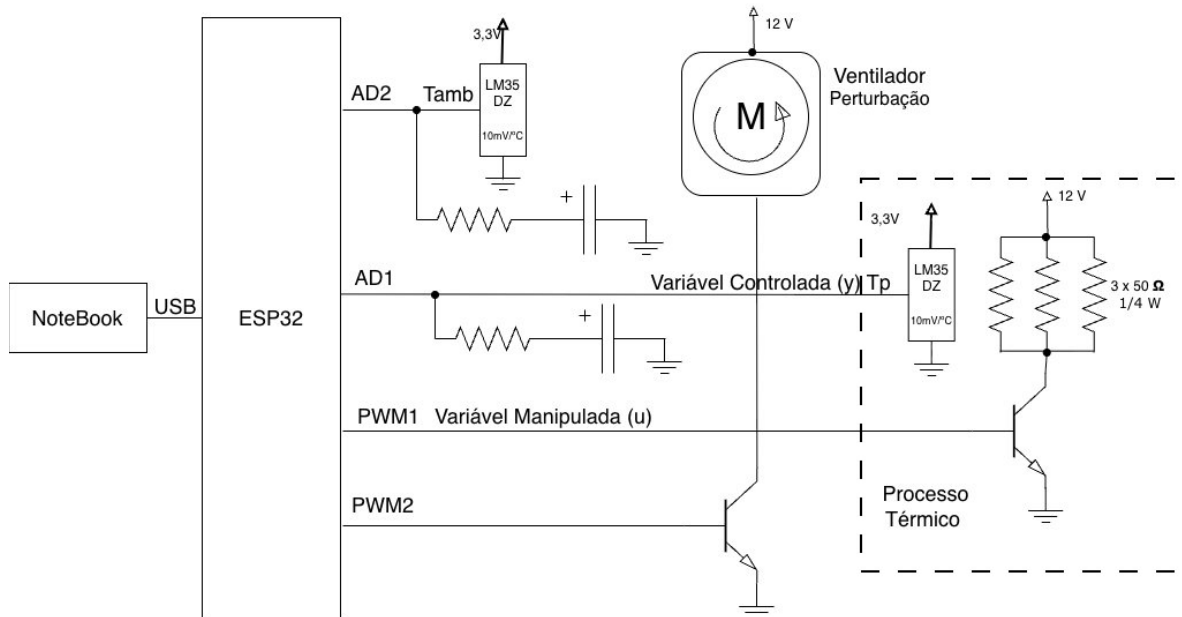


Figura 1 – Diagrama esquemático Prj1CDig.

Considere o modelo de aquecimento PWM do processo térmico, figura 2. Este modelo foi obtido utilizando-se uma aproximação a parâmetros concentrados de princípios fundamentais de fenômenos de transporte de calor. [Bauchspiess, A. et al. First-Principles Structured Identification For Predictive Hvac Control. In: XII Congresso Latino-Americano de Controle Automático (XII CLCA), 2006, Salvador, 2006. p. 42-47]. Tal modelo é conhecido por “caixa cinza”, pois utiliza algum conhecimento das leis físicas (princípios fundamentais) que regem o processo. É importante ressaltar que a função de transferência que representa um processo físico não é único. Outro modelo conveniente é a forma modal, obtida por expansão em frações parciais. Além de inúmeras realizações no espaço de estado, que permitem contemplar condições iniciais.

O sensor LM35 e os resistores que o cercam apresentam uma capacitância térmica própria, isto é, armazenam energia térmica no conjunto das massas envolvidas. A temperatura T é função de um fluxo de calor de aquecimento produzido pelo circuito de PWM e de um fluxo de calor de resfriamento produzido pelo meio ambiente e pelo ventilador. A temperatura do meio ambiente, T_a , deve ser medida ao longo do experimento. Variações de T_a e perturbação W , constantes são mitigadas pelo canal-I do controlador PI. As perturbações de resfriamento senoidais, produzidas pelo ventilador, serão estimadas e compensadas no Prj2Cig.

O princípio fundamental de transferência de calor associado à condução e à convecção (as duas formas estão presentes no experimento) são $q_p = K_p (T - T_p)$, $q_w = K_w (T - T_w)$ e $q_a = K_a (T - T_a)$, para o acionamento PWM, perturbação e Meio Ambiente, respectivamente. Note que para cada montagem, cada grupo, as constantes de tempo, os ganhos e a temperatura ambiente serão ligeiramente diferentes, além do ruído, que é estocástico num processo a parâmetros distribuídos. A modelagem a parâmetros concentrados, é suficiente, para os propósitos acadêmicos deste experimento.

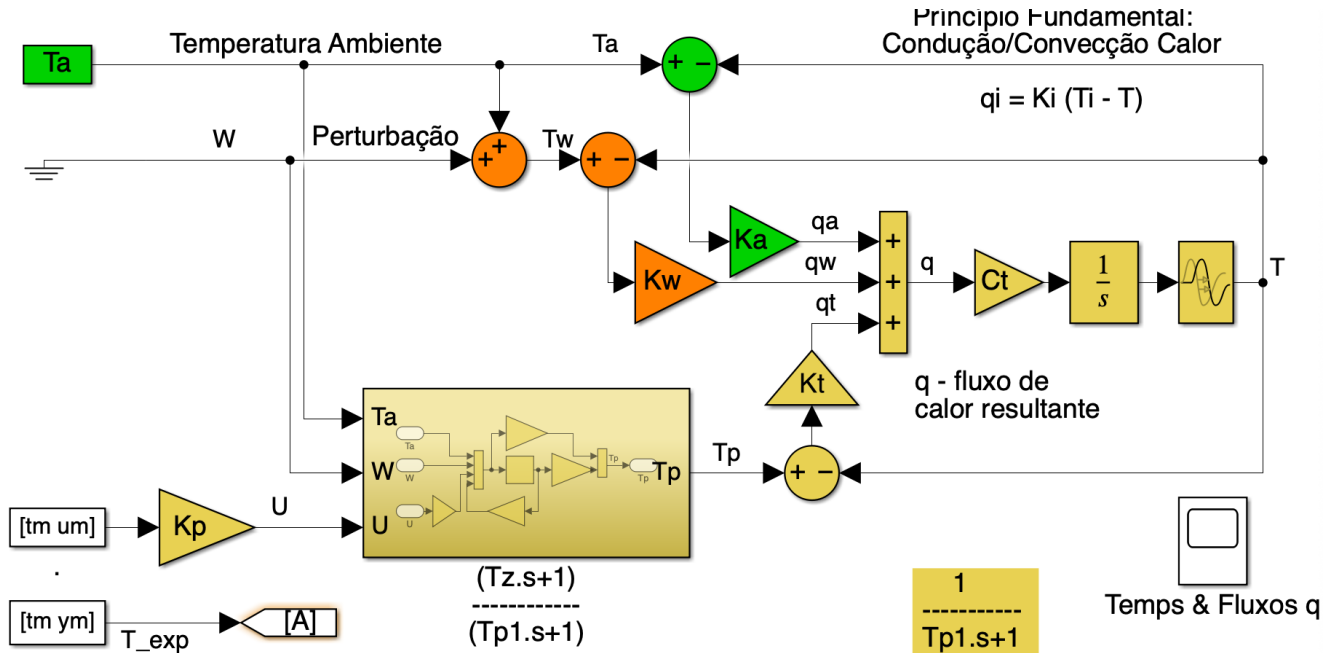


Figura 2 – Modelo princípios fundamentais do processo. Capacitância térmica e fluxos de calor do ambiente, q_a , aquecimento PWM, q_p , e perturbação, q_w , permitem a “entender” o processo de transferência de calor. (Termico_FP.slx).

A obtenção dos parâmetros do processo é feita, em malha aberta, com um sinal de entrada pseudo-aleatória, para ponto de operação de $u(t)$, em torno do qual o modelo é válido, ver Figura 3. PRBS é fortemente recomendado para que não haja correlação do sinal de entrada com os parâmetros identificados. O ideal é que a excursão do sinal do sinal de temperatura aconteça entre 40° e 90°C. O modelo que melhor explica a relação entre temperatura e sinal do PWM utiliza dois modos, duas constantes de tempo: polo rápido associado ao aquecimento PWM, polo lento associado a T_a .

$$\frac{Y(s)}{U(s)} = \frac{K_p (T_z s + 1) e^{-T_d s}}{(T_{p1} s + 1)(T_{p2} s + 1)}$$

Procedimento de obtenção do Modelo

- Taxa de amostragem uniforme, $T_s = 1$ s.
- P.O. Ponto de operação.
- Aplicar o sinal $U(t)$ obtido de `prbs.m/idinput()` ao processo. Salvar $Y(t)$ (`u=idinput(31500,'PRBS',[0 .01],[512-128 512+128])`: identification toolbox).

```
> data=iddata(yid(1:len_id), uid(1:len_id), Ts);
> kk=delayest(data);
> G=procest(data,'P2DZ','InputDelay',kk); h=get(G);
> Kp=h.Kp; Tz=h.Tz; Tp1=h.Tp1; Tp2=h.Tp2; Td=h.Td;
```

Valores típicos P2DZ

```
Kp = 0.09078;
Tp1=352.96493;
Tp2= 48.86313;
Tz =249.22561;
Td = 4.66250;
```

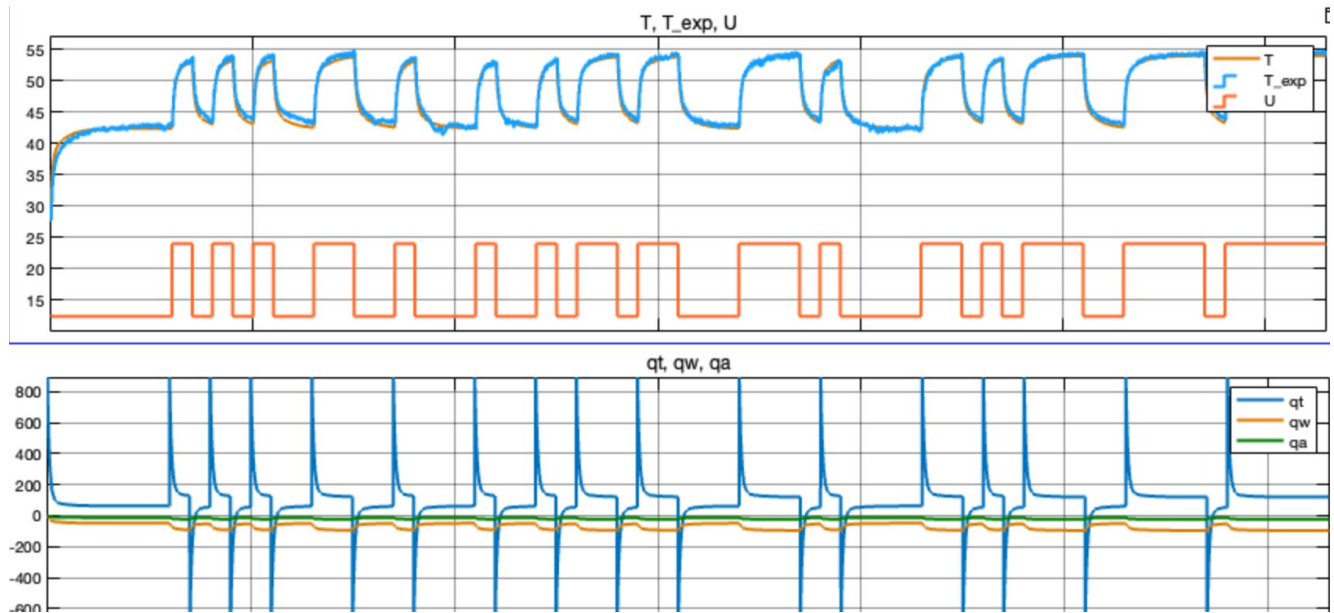


Figura 3 – Sinais de entrada ‘PRBS’ e saída típicas do processo térmico utilizados na identificação.

2. Projeto e Simulação do Controle PI-Anti-Windup

Com o modelo do processo obtido na seção 1, é possível projetar um controle PI via LGR, com a ferramenta linear da toolbox de controle do matlab SISOTOOL(). O efeito da saturação e do Anti-Windup, demandam o Simulink. (Prj1CDig.slx, Prj1CDig_m.m)

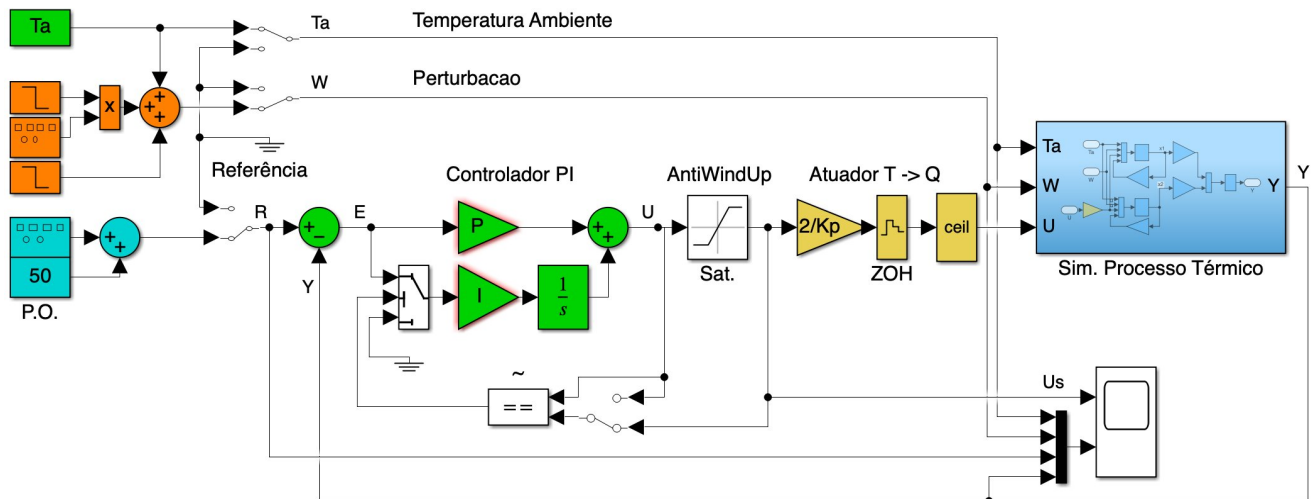


Figura 4 – Controle PI Anti-Windup de Processo Térmico.

2.1 O Efeito Windup

Todos os atuadores de sistemas físicos apresentam saturação, limite máximo e mínimo que restringe as variáveis de entrada e internas ao processo. Quando o controle em malha fechada utiliza canal integral este pode acumular erro que não é devido à diferença entre entrada e saída, mas produzido pela saturação do atuador. Este fenômeno é conhecido como “windup” e causa em geral oscilações indesejadas na resposta do sistema. Um circuito “anti-windup” mantém nula a entrada do integrador durante a saturação do atuador de tal forma que, ao sair da saturação, não haja erro “indevidamente” acumulado pelo integrador. Ver exemplo ilustrativo na Figuras 5 – controlador PI com e sem “anti-windup”.

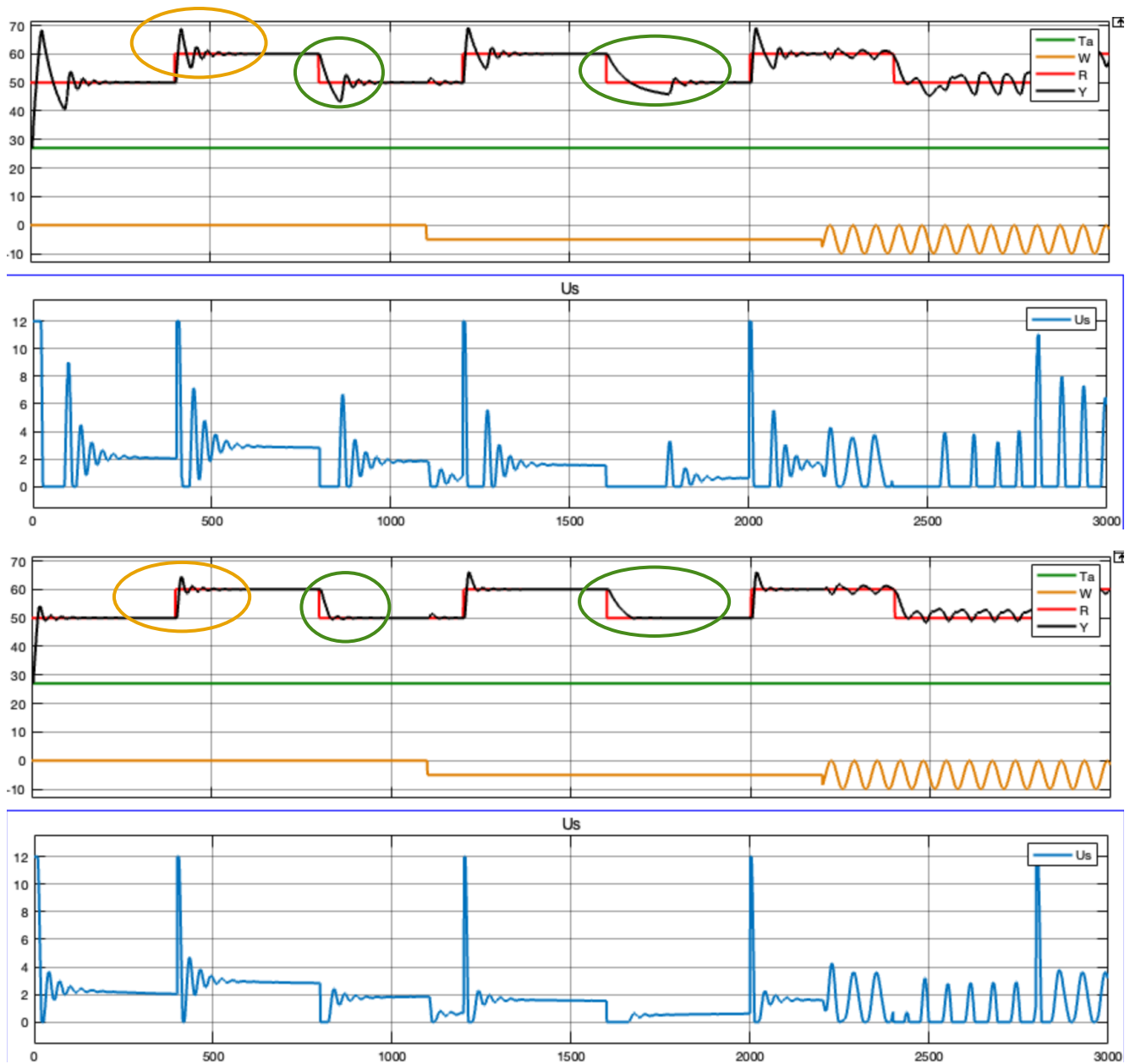


Figura 5 – Controle PI convencional (acima) e com Anti-Windup (abaixo). O acúmulo de erro, enquanto a variável manipulada está saturada, degrada a resposta do sistema em malha fechada. Ao sair da saturação o erro acumulado indevidamente precisa ser “desacumulado”.

2.1 Especificações de Projeto

- Sobrepasso projetado em malha fechada $M_p\%$.
- Tempo de pico, T_p .
- Rejeição de perturbações constantes por partes.
- Período de amostragem de 1 *seg*.
- Resolução da temperatura $\leq 0,1^\circ\text{C}$.

Grupos	1, 11	2,12	3,13	4,14	5,15	6,16	7,17	8,19	9,19	10
Parâmetros										
ω_0 rad/s	0.065	0.065	0.065	0.065	0.065	98	98	98	0.098	98
P.O. [$^\circ\text{C}$]	45	50	55	60	45	50	55	60	45	50
M_p [%]	10	10	10	25	25	45	45	45	45	45
T_p [s]	30	30	20	20	25	25	30	30	20	20

3. Resultado Esperado

Mostrar na tela do PC, em tempo real, a referência, saída e sinal de controle para o controle PI com rejeição de perturbações. A figura 6 ilustra um resultado típico obtido com o controlador PI sem Anti-Windup.

4. Relatório

Incluir no relatório o LGR, simulações, circuito implementado, código fonte e resultados experimentais. Observar o template disponibilizado.

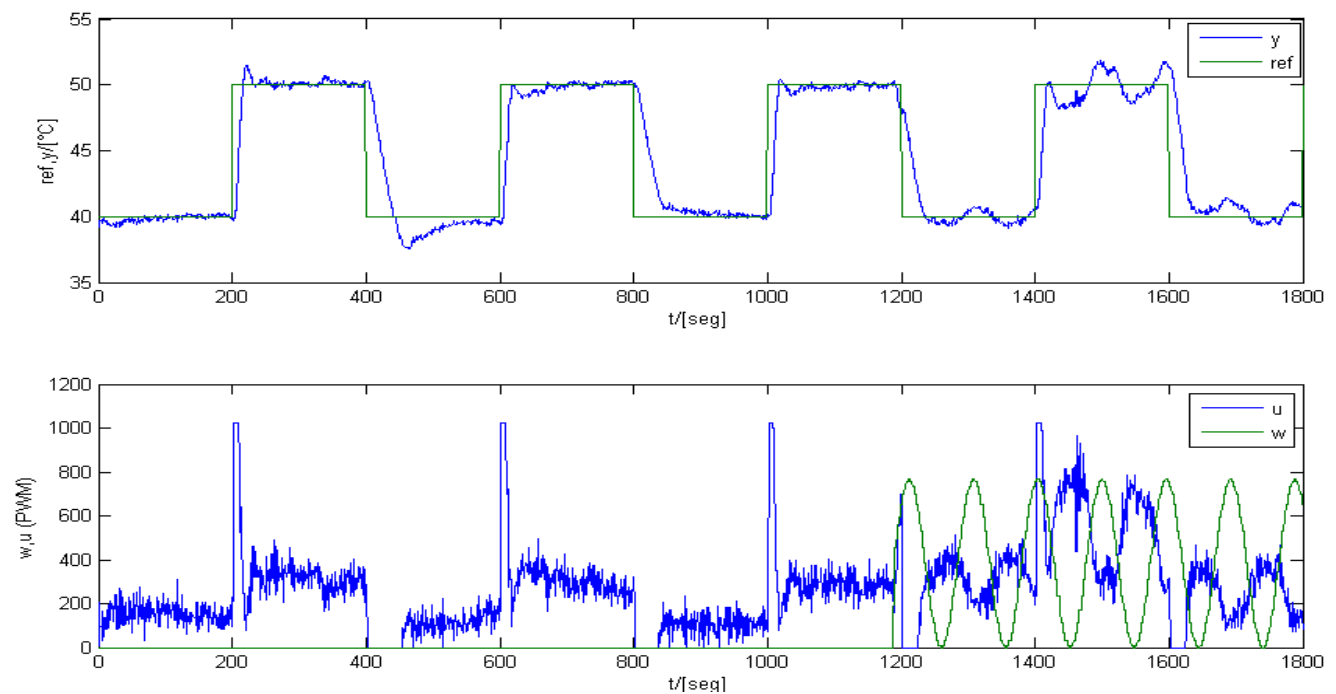


Figura 6 – Resposta típica do Controlador PI (sem Anti-Windup). A partir de $t=1200$ s, vê-se a perturbação senoidal.