

ES670

Projeto de Sistemas Embarcados

Projeto Final - Grupo A7



171859 Gabriel Barbosa Haj
182806 Lucas Issao Yonei

21 de junho de 2023

1 Documentação do Sistema

Requisitos Funcionais	Requisitos Não Funcionais
Controlador de Temperatura	Overshoot Máximo 2 °C
Aquisição e ajuste de parâmetros do sistema por meio de comunicação serial UART	Temperatura máxima de 90 °C
Ajuste de temperatura de referência via botões	O sistema deve aquecer o mais rápido possível
Visualização de parâmetros do sistema através de telas no LCD	Atuação do controlador via aquecedor resistivo e cooler para resfriamento
Controle do backlight do LCD via botão Enter	Sistema deve ser alimentado por uma fonte 12V

Tabela 2: Tabela de requisitos funcionais e não funcionais do sistema

1.1 Detalhamento das especificações

- O sistema exibirá as informações localmente via LCD 2x16 que possui 4 telas controladas via botões direita e esquerda sendo elas:
 - Tela 1: Temperatura Desejada e Temperatura Atual (Aqui o usuário poderá alterar o valor da temperatura desejada através dos botões cima e baixo)
 - Tela 2: Duty Cycle do Heater e Duty Cycle do Cooler
 - Tela 3: Parâmetros do controlador, Kp e Ki
 - Tela 4: Velocidade de rotação do Cooler
- Controle Proporcional-Integral
- Botões serão utilizados para alterar o valor da temperatura desejada, trocar de telas , e o botão enter para controlar o backlight do LCD
- A temperatura do sistema é medida através de um sensor analógico de temperatura
- A velocidade de rotação do Cooler é medida através de um led-tacômetro
- O controlador aquece o sistema através da atuação de um resistor excitado por PWM
- O controlador esfria o sistema através da atuação de um cooler excitado por PWM
- O sistema pode ter seus parâmetros alterados e aqisitados via comunicação serial UART

1.2 Diagrama de Blocos

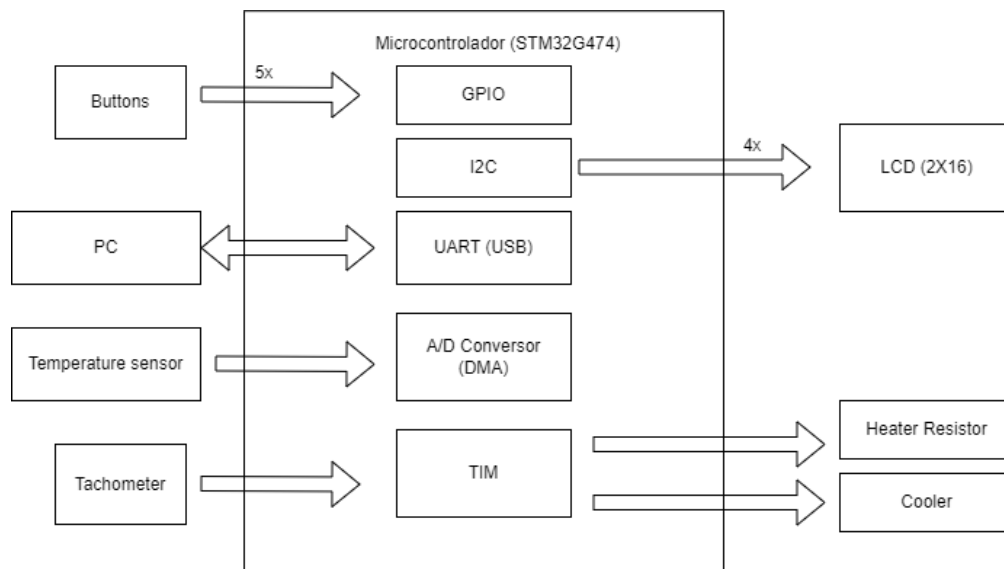


Figura 1.1: Diagrama de blocos do sistema

1.3 Diagrama de Camadas

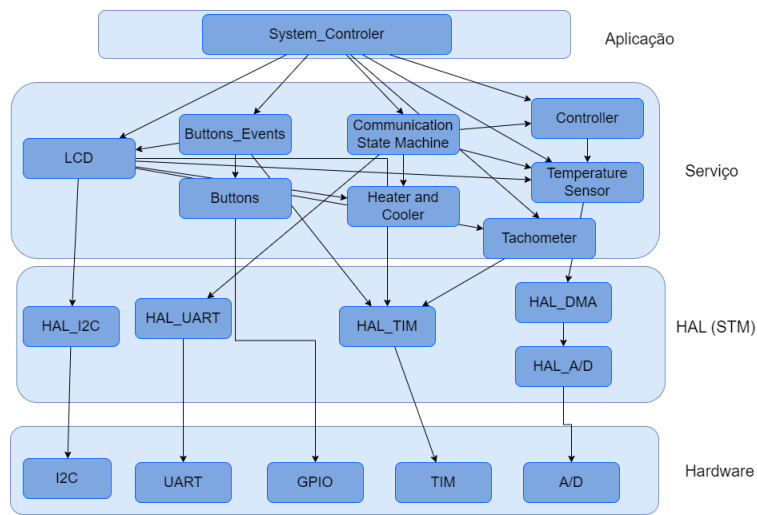


Figura 1.2: Diagrama de camadas do sistema

1.4 Diagrama de Classes

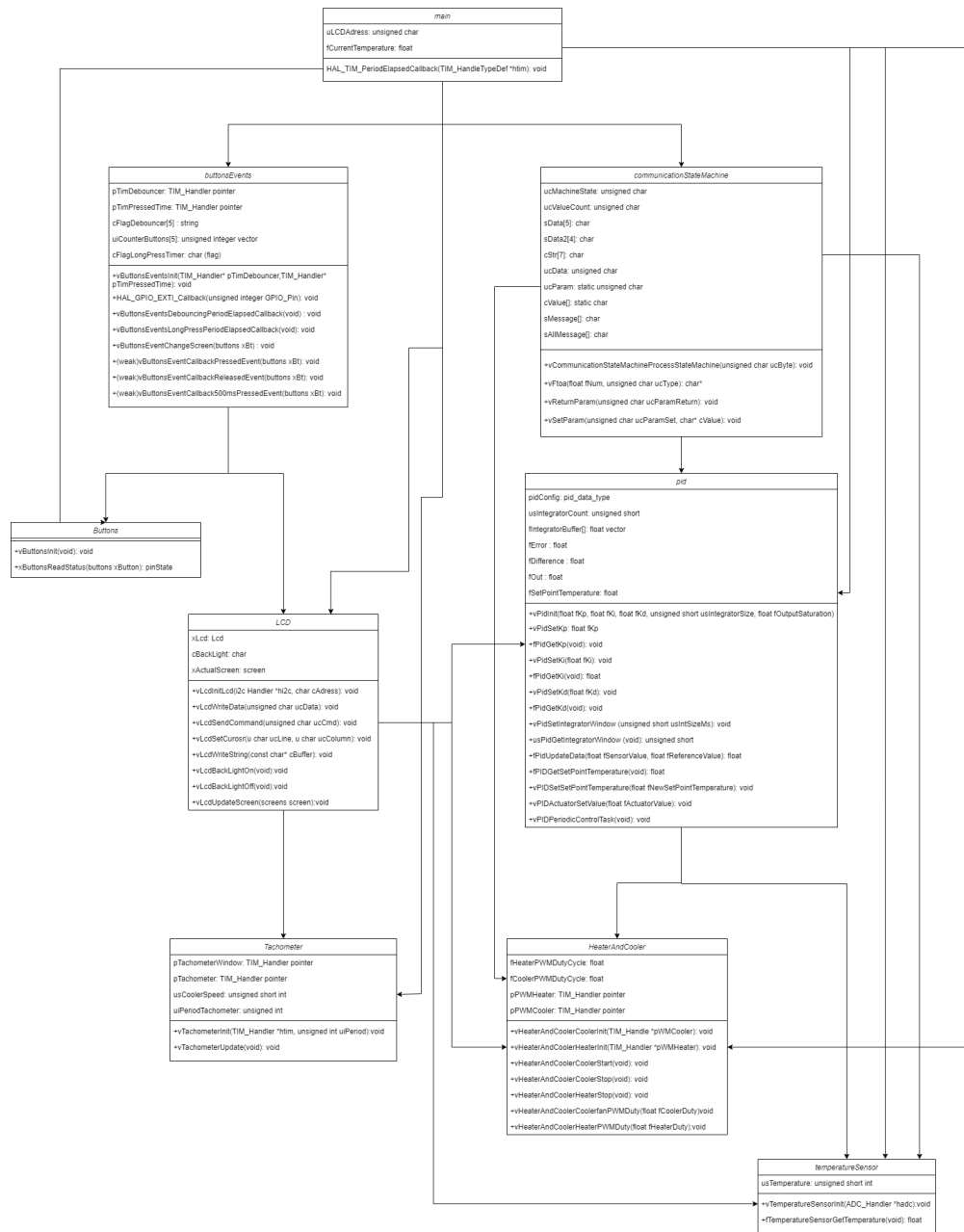


Figura 1.3: Diagrama de Classes do sistema

1.5 Fluxogramas pertinentes

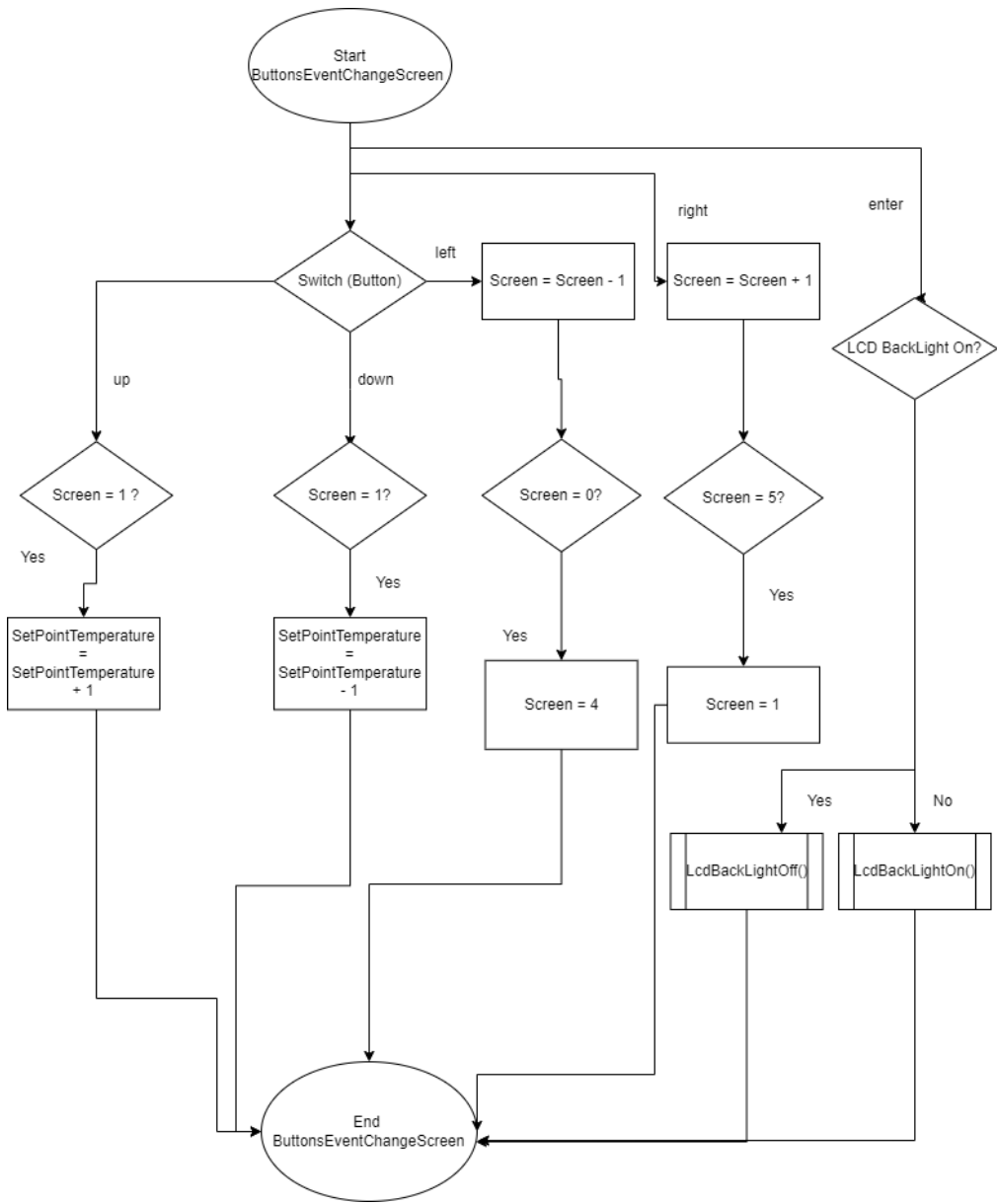


Figura 1.4: Fluxograma de troca de telas do LCD

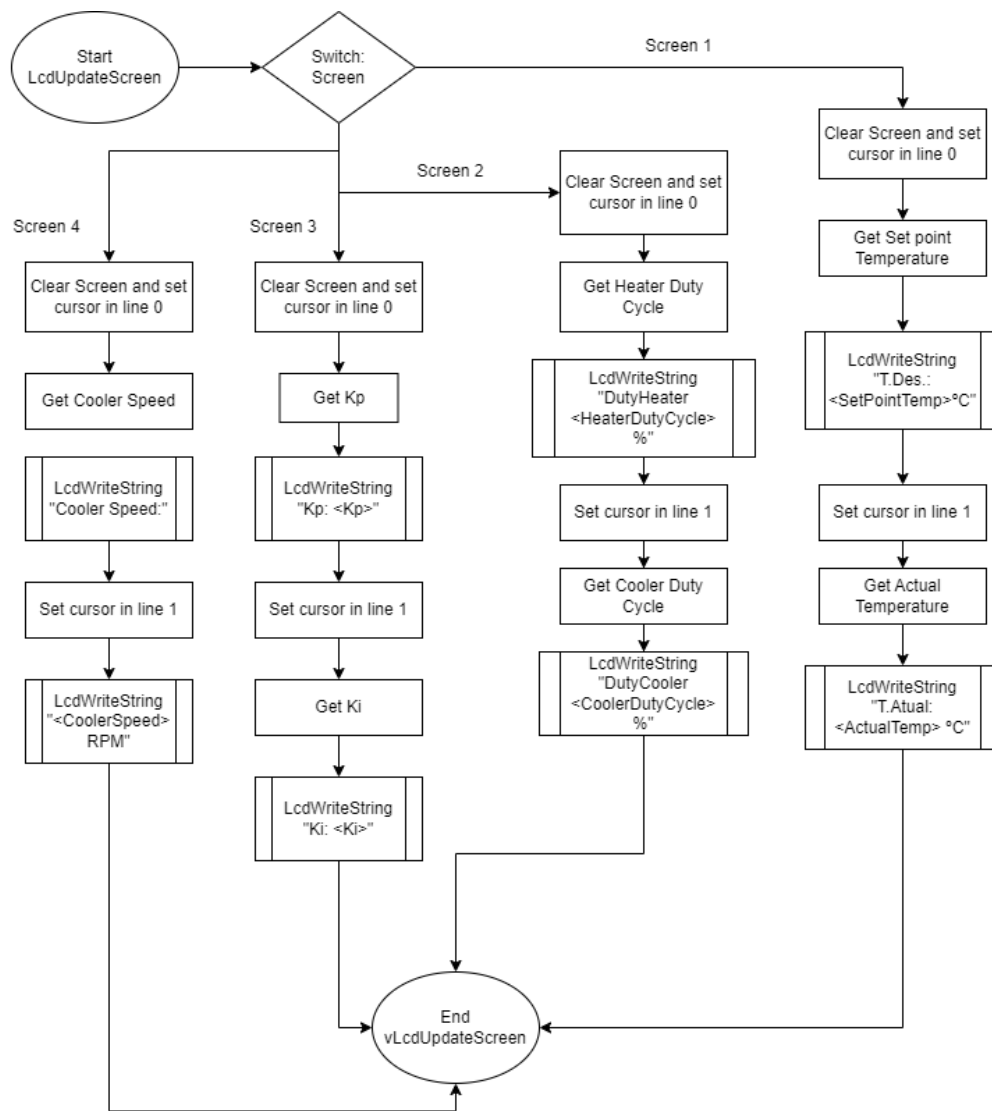


Figura 1.5: Fluxograma de interação com os valores do LCD

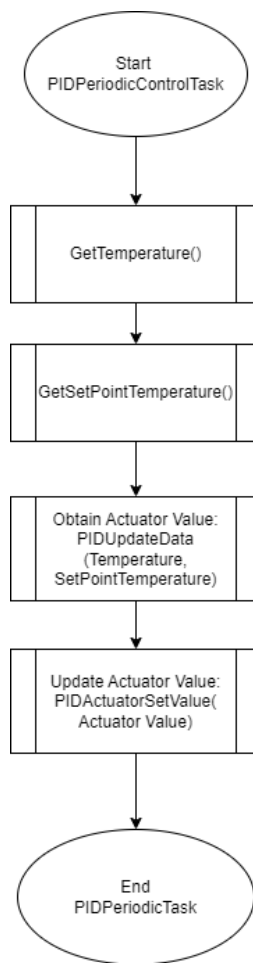


Figura 1.6: Fluxograma da função periódica de controle

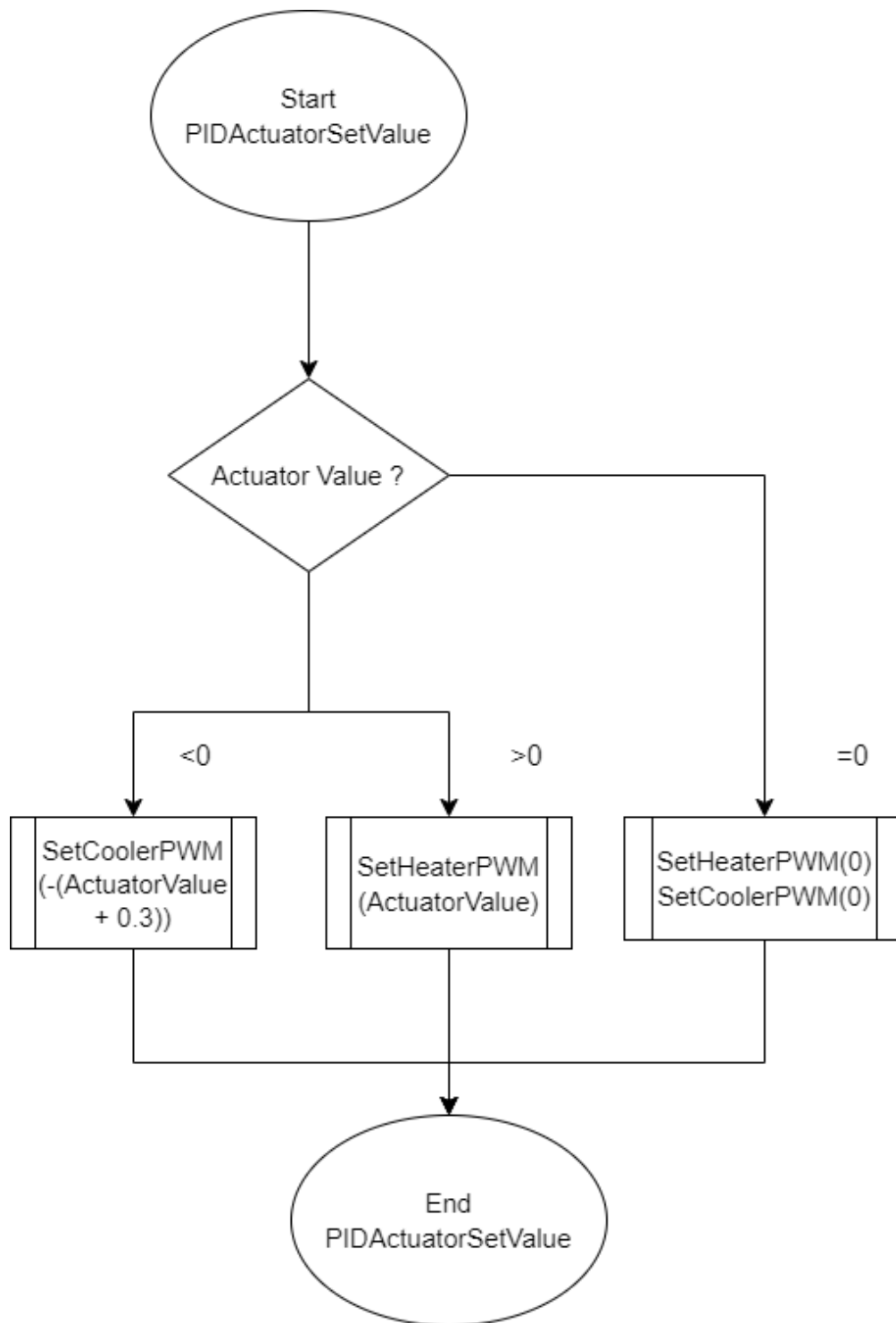


Figura 1.7: Fluxograma de controle dos valores dos atuadores

2 Procedimento de Sintonização do Controlador

Inicialmente optou-se por utilizar um controlador PI para alcançar bons resultados e caso não fossem alcançados poderia-se utilizar o PID.

Além disso o sistema de controle é baseado no acionamento de dois atuadores. O resistor para aumentar a temperatura e o cooler para diminuí-la.

Para sintonização do controlador utilizamos o método de Ziegler-Nichols de malha fechada. Zeramos os

valores dos parâmetros de controle e aumentamos gradativamente o parâmetro proporcional até encontrarmos um valor que leva-se a instabilidade do sistema. Esse valor foi de 1000.

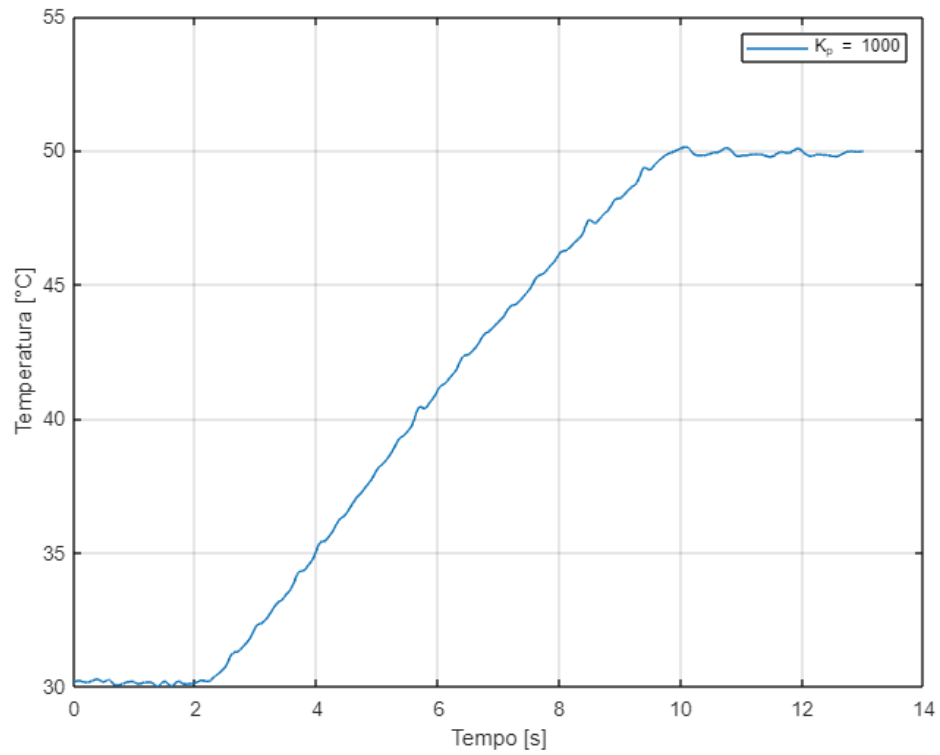


Figura 2.1: Comportamento do sensor de temperatura para uma entrada de 50 °C

Como podemos ver, o sistema levou em torno de 8 segundos para alcançar a temperatura desejada com poquíssimo overshoot. Iremos então avaliar a temperatura em cima da linha de 50.

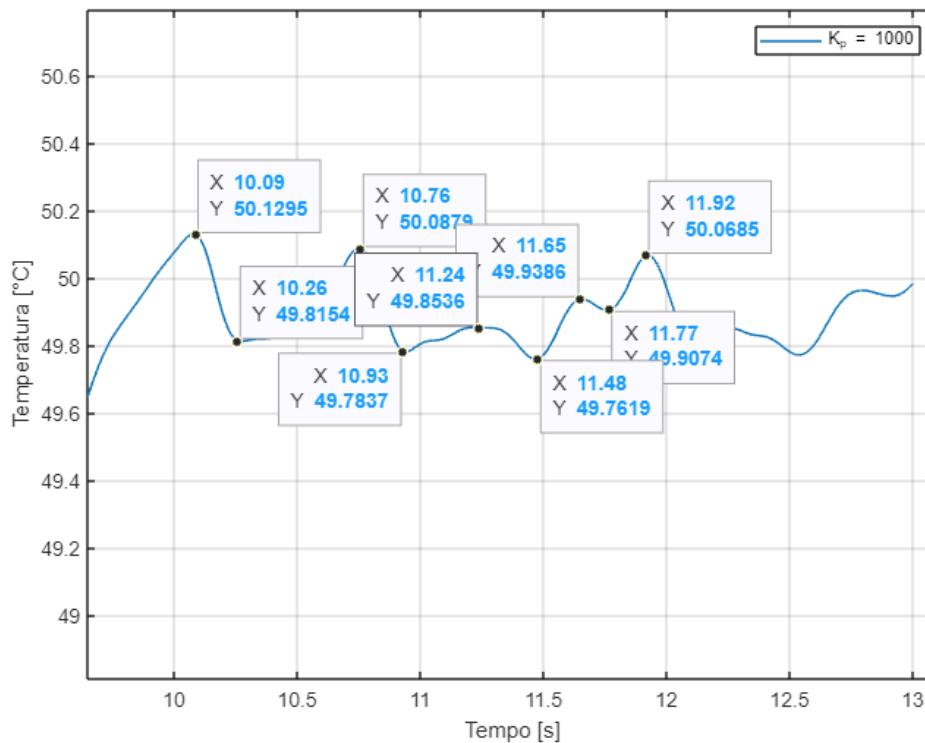


Figura 2.2: Comportamento do sistema em torno dos 50 °

A partir dos valores retirados da oscilação em torno dos 50°, pôde-se calcular os parâmetros de Ziegler-Nichols. Calculamos o Período crítico como 0.2s e Ganho crítico como 1000. Chegamos então a $K_p = 450$ e $K_i = 0.17$. Porém, implementamos $K_p = 600$ e $K_i = 1$.

Além disso criamos um treshhold para a atuação do cooler. O cooler só irá atuar quando o esforço de controle para a diminuição da temperatura for maior que 30%. E o valor máximo de esforço será de 70%

3 Manual de Utilização

3.1 Interface Local

O programa já vem com os parâmetros de controle setados. Na interface local o usuário poderá apenas ajustar a temperatura desejada e ligar e desligar a backlight do LCD. O usuário poderá interagir com o LCD através dos botões direita e esquerda, que trocará as telas do LCD. Em cada tela ele terá acesso a uma informação diferente do sistema. No total, o sistema possui 4 telas.

Apenas na primeira tela o usuário será capaz de aumentar e diminuir o valor da temperatura desejada através dos botões de cima e baixo, 1 unidade de temperatura a cada 500ms ou a cada pressionamento.

3.2 Interface remota via UART

Através da interface remota via comunicação serial UART o usuário será capaz de alterar e obter os valores da temperatura atual, duty cycle do heater e do cooler e os parâmetros do controlador.

O usuário deverá seguir o seguinte padrão de escrita: `-<função><parâmetro><valor(opcional)>!`

As funções possíveis são : get 'g' e set 's' Os parâmetros possíveis são: temperatura 't', Duty Cycle do heater 'h', Duty Cycle do cooler 'c', ganho proporcional 'p', ganho integrativo 'i' e ganho derivativo 'd'. Além disso, a função get possui também o parâmetro 'a' que devolve todos os parâmetros.

Voce pode setar um parâmetro tanto usando virgulas quanto pontos. A resposta da função get terá o seguinte formato: `-a<parâmetro><Valor>!`

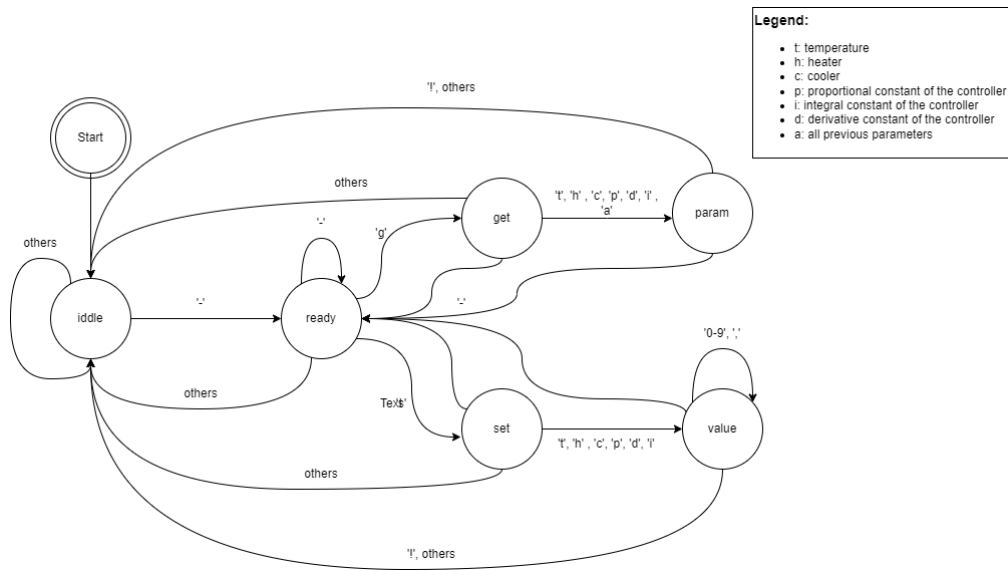


Figura 3.1: Diagrama de estados

4 Problemas Identificados e Não Resolvidos

Nos propomos a limitar a atuação do cooler exigindo que o esforço de controle necessário para acioná-lo fosse de no mínimo 30% e sua atuação fosse até 70%. Porém não conseguimos validar sua implementação.

5 Autoria dos Códigos Fornecidos

Os códigos não autorais utilizados foram os fornecidos pelo professor para o método PID e para o display LCD.