



universidade de aveiro

departamento de eletrónica, telecomunicações e informática

Curso 8309 - Mestrado Integrado em Engenharia Eletrónica e Telecomunicações

Disciplina 41489 – Sistemas de Instrumentação Eletrónica

Ano letivo 2019/20

Relatório

Projeto 2 – Sistema para Controlo de um Processo Térmico

Autores:

84667 Hugo Micael Resende Leal

84774 Miguel Filipe Pereira de Freitas Carvalhosa

Turma P2 Grupo 9

Data 18/05/2020

Docente Pedro Nicolau Faria da Fonseca

Resumo: O presente relatório visa apresentar o trabalho desenvolvido durante o projeto 2 da unidade curricular Sistemas de Instrumentação Eletrónica, o qual consiste no desenvolvimento de um sistema para controlo de um processo térmico com a placa chipKit Max32.

Introdução

O presente documento tem como objetivo apresentar o trabalho desenvolvido e os resultados para o Projeto 2, Sistema para Controlo de um Processo Térmico, da unidade curricular de Sistemas de Instrumentação Eletrónica.

O projeto 2 consiste no desenvolvimento de um sistema que permita controlar a temperatura de um elemento de aquecimento. Neste projeto é usado como elemento de aquecimento uma resistência de potência. O projeto em causa enquadra-se na unidade curricular pois permite aos alunos adquirirem competências na leitura de sensores como o usado (PT100), bem como no desenvolvimento de circuitos de acondicionamento para este tipo de sensores. Permite ainda aos alunos adquirirem experiência na implementação de controladores do tipo proporcional-integrador (PI) em microcontroladores e no desenvolvimento de interfaces com o utilizador via UART.

Este relatório segue uma abordagem “*top-down*”. Primeiramente será apresentada uma descrição da arquitetura do sistema desenvolvido, com recurso a diagramas de blocos, descendo de seguida até à especificação pormenorizada do hardware e do software desenvolvidos.

Descrição do problema e objetivos

O problema apresentado aos alunos consiste no desenvolvimento de um sistema para controlo de um processo térmico usando a placa de desenvolvimento *chipKit Max32* da Digilent. O principal objetivo do trabalho prático é permitir a um dado utilizador controlar a temperatura do elemento de aquecimento. Assim, o sistema deve possuir uma interface com o utilizador, no computador, que permita a este definir o *setpoint* de temperatura a que deverá estar o elemento de aquecimento. A interface com o utilizador deve mostrar a temperatura instantânea a que se encontra o elemento de aquecimento e a temperatura máxima atingida desde a inicialização do sistema. A leitura da temperatura é efetuada recorrendo a um sensor do tipo PT100.

O problema tem as seguintes especificações:

- Comunicação entre o computador e o microcontrolador deve ser realizada através da UART;
- O elemento de aquecimento deve ser uma resistência de potência ($10W$, 22Ω), alimentada com $12V$;
- A temperatura deve variar entre $40^{\circ}C$ e $70^{\circ}C$, com *step* de $1^{\circ}C$;
- A interface com o utilizador deve mostrar a velocidade instantânea, com resolução de $1^{\circ}C$;
- A interface com o utilizador deve exibir a temperatura máxima desde a inicialização do sistema;
- A função de controlo deve ser implementada com recurso a um controlador Proporcional-Integrador (PI).

Arquitetura do Sistema

A arquitetura do sistema está representada com recurso a diagramas de blocos e divide-se em 3 níveis:

- **nível 0**, onde se representa o sistema visto como um todo, estando representados os sinais de entrada e saída da generalidade do sistema;
- **nível 1**, onde se apresenta o sistema dividido pelos seus blocos principais e as entradas e saídas de cada um destes;
- **nível 2**, onde se apresenta a estrutura interna do microcontrolador e os blocos de hardware e software utilizados, bem como as entradas e saídas de cada um.

Diagrama de blocos de nível 0

Na Figura 11, encontra-se o diagrama de blocos de nível 0 do sistema. Neste, o sistema é representado na sua totalidade como apenas um bloco, estando representados apenas os sinais de entrada e de saída da globalidade do sistema.

Neste diagrama são representados os sinais de entrada e saída do sistema com as seguintes características:

- **S1**: Representa o *setpoint* da temperatura inserido pelo utilizador. Toma valores entre $40^{\circ}C$ e $70^{\circ}C$;
- **S2**: Informação da temperatura lida com recurso à PT100;

- **S3:** Sinal digital UART com informação do valor de temperatura instantânea e máxima lidas.

Diagrama de blocos de nível 1

Na Figura 12, encontra-se esquematizado o diagrama de blocos de nível 1 do sistema. Neste encontram-se representados os seguintes blocos:

- **chipKit Max32:** onde se realizará a geração e o envio do sinal PWM para a interface de potência, a interface com o utilizador via UART e a conversão de uma tensão analógica em tensão digital, com recurso à ADC.
- **Interface de potência:** bloco de hardware responsável pela entrega de corrente ao elemento de aquecimento.
- **Elemento de aquecimento:** bloco que representa o elemento sobre o qual pretendemos atuar e ler os valores de temperatura.
- **Sensor de Temperatura:** bloco que efetua a leitura da temperatura do elemento de potência e converte este valor num sinal em tensão analógico útil para processamento no microcontrolador.

No diagrama encontram-se ainda representados os seguintes sinais:

- **S4:** sinal PWM com frequência de 2kHz, entre 0 e 3.3V.
- **S5:** corrente entregue ao elemento de potência. Varia entre 0 e 550mA, conforme o *duty-cycle* do sinal PWM varie entre 0 e 100%.
- **S6:** sinal analógico de tensão entre 0.15V e 3.15V.

Diagrama de blocos de nível 2

Na Figura 13, encontra-se representado o diagrama de blocos de nível 2 do sistema. Neste diagrama, o sistema é representado por todos os blocos de software e hardware que o constituem, nomeadamente:

- **UART:** bloco para enviar e receber dados para o computador via UART.
- **Controlador PI:** bloco de software que implementa um algoritmo de um controlador proporcional-integrador que trata do controlo da temperatura do elemento de aquecimento para o *setpoint* desejado.
- **Gerador de PWM:** bloco para gerar um sinal PWM baseado em um *timer* e um *output compare module* do PIC32.
- **ADC:** bloco que realiza a conversão do sinal de tensão analógica à saída do circuito de acondicionamento de sinal num sinal em tensão digital entre 0 e 3.3V
- **Fonte de Corrente:** circuito que realiza a polarização da PT100. Gera uma corrente DC estável de 1mA.
- **PT100:** bloco que representa a leitura da temperatura do elemento de aquecimento. Gera um sinal analógico em tensão.
- **Circuito de acondicionamento de sinal:** circuito responsável por truncar e amplificar o sinal vindo do sensor de temperatura, de modo a maximizar a resolução da medição.

No diagrama estão também representados os sinais:

- **S7:** sinal em corrente de 1mA que tem como função polarizar a PT100.
- **S8:** sinal analógico de tensão proporcional à temperatura lida do elemento aquecimento.

Projeto e conceção

Neste ponto, encontra-se o dimensionamento e cálculos teóricos do hardware adicional usado, bem como as etapas de conceção de algum do software mais relevante. O esquema geral dos circuitos desenvolvidos encontra-se na Figura 14 dos anexos.

Hardware

Circuito de Aquecimento

O elemento de aquecimento utilizado neste projeto corresponde a uma resistência de potência de 22Ω com uma dissipação máxima de potência de 10W. De modo a ser possível controlar a temperatura a que este elemento se encontra, é necessário um circuito que forneça energia (corrente) ao mesmo e que a corrente fornecida varia de acordo com a proximidade do valor de temperatura lido com o *setpoint* de temperatura desejado. Para tal, o circuito dimensionado corresponde ao da Figura 1.

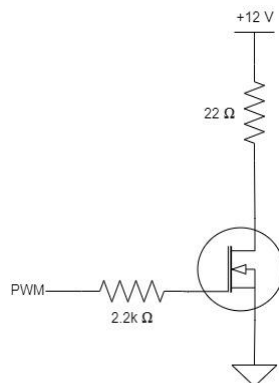


Figura 1 – Circuito de aquecimento projetado

O elemento central deste circuito é um mosfet de potência do tipo NMOS, o IRLZ44N. Este tem como função fornecer corrente ao elemento de aquecimento que se encontra ligado ao dreno do mesmo. A quantidade de corrente entregue à resistência de potência é controlada com o sinal PWM aplicado à porta do transistor. Quando o sinal de PWM tem um duty-cycle de 100% e, portanto, uma tensão média de 3.3V, o mosfet encontra-se na região de saturação e a corrente máxima é entregue ao elemento de aquecimento. Já quando o PWM tem um duty-cycle de 0%, o valor médio de tensão aplicada na porta do mosfet é 0, e assim o mesmo encontra-se em corte, sendo a corrente entregue à resistência de potência nula. Assim, a corrente entregue ao elemento de aquecimento varia proporcionalmente com o duty-cycle do PWM. O valor da alimentação deste circuito é 12V, conforme as especificações. A resistência de $2.2k\Omega$ tem a função de proteger a porta do mosfet. Recorrendo à lei de ohm, o valor de corrente máxima fornecida ao elemento de potência é de $I = \frac{12}{22} \approx 0.55$ A e o valor mínimo é 0.

Fonte de Corrente

Atendendo o datasheet da PT100^[5] em uso, é aconselhado realizar medições com uma corrente de medição de 1mA. Como tal, foi dimensionada uma fonte de corrente que polariza a PT100 com 1mA constantes. Visto que a PT100 varia a sua resistência de acordo com o valor de temperatura que mede, esta aumenta com o aumento da temperatura, a fonte de corrente projetada deverá fornecer 1mA independentemente destas variações de resistência. Tendo por base os slides do capítulo de temperatura da unidade curricular foi efetuado o dimensionamento da fonte de corrente com a topologia da Figura 2.

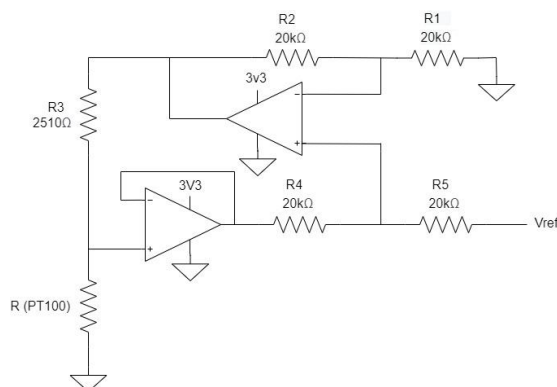


Figura 2 – Fonte de corrente projetada

Esta fonte de corrente gera uma corrente estabilizada para a carga, neste caso a PT100, que está relacionada diretamente com o dimensionamento da tensão de referência e a resistência R3. Atendendo à topologia do circuito fazendo as resistências R1, R2, R4 e R5 iguais faz com que as quedas de tensão nestas sejam igualmente iguais. Devido ao curto-circuito virtual no opamp de cima a tensão na entrada inversora e não-inversora do opamp serão iguais. Sendo as resistências R1, R2, R4 e R5 de igual valor e escolhendo uma tensão de referência igual a 2.5V, visto que esta tensão é facilmente gerada com recurso a um circuito tendo por base um TL431, as quedas de tensão nestas será de aproximadamente 1.25V. Tendo na entrada inversora do opamp de cima 1.25V, à saída do opamp a tensão será idealmente igual a 2.5V, visto que como dito anteriormente a queda de tensão em R2 será igual à queda em R1, devido a estas possuírem valores iguais. Assim, sabendo que temos aproximadamente 2.5V à saída do opamp de cima e sabendo que a corrente que queremos entregar à PT100 é de 1mA, recorrendo à lei de ohm o valor de resistência R3 necessário para este efeito é calculado.

$$R_3 = \frac{2.5}{1m} = 2500\Omega \quad I$$

O opamp mais em baixo no circuito é um buffer que tem como função garantir que a corrente não é desviada da carga para o restante circuito. A limitação deste circuito prende-se no facto de se a carga tiver um valor significativo em relação à resistência R3 (cerca de algumas centenas de ohm), então a queda de tensão na carga não pode ser desprezada e por isso, o circuito seria incapaz de fornecer 1mA como desejado. Porém, visto que para a gama de temperaturas de medição [40°C; 70°C] a resistência da PT100 varia na gama de [115.541 Ω ; 127.075 Ω] este problema não ocorre e são entregues 1mA conforme o previsto.

Em termos de componentes, os opamps utilizados são ambos MCP6022 e para a resistência R3 foram utilizadas duas resistências, uma de 2000 Ω e uma de 510 Ω (série E24), de modo a manter o valor o mais próximo possível do dimensionado teoricamente. A tensão de referência de 2.5V foi gerada com recurso a um TL431 e a uma resistência de 47 Ω conforme a *Figura 14* dos anexos.

Circuito de Acondicionamento de Sinal

O circuito de acondicionamento de sinal após medição da temperatura na PT100 tem como função maximizar a resolução da medição. Pretendemos medir temperaturas entre [40, 70] °C, o que corresponde a valores máximos e mínimos de resistência e consequentemente tensão da Tabela 1. No entanto, a gama de tensões que este sensor produz após medição é muito pequena em comparação com a gama de leitura da ADC, [0, 3.3] V. Assim, é necessário um circuito que trunque e amplifique o sinal à saída do sensor, de modo à gama de interesse ficar compreendida na gama de tensões da ADC, maximizando assim a resolução da leitura. Para este efeito foi projetado o circuito da Figura 3. Os valores de tensão da Tabela 1 foram calculados tendo por base os valores tabelados de resistência da PT100 consoante a temperatura a que se encontra multiplicados pela corrente de 1mA gerados pela fonte de corrente.

Temperatura (° C)	Resistência (Ω)	Tensão (mV)
40	115.541	115.541
70	127.075	127.075

Tabela 1 - Valores de resistência da PT100 e tensão para a gama de temperaturas considerada

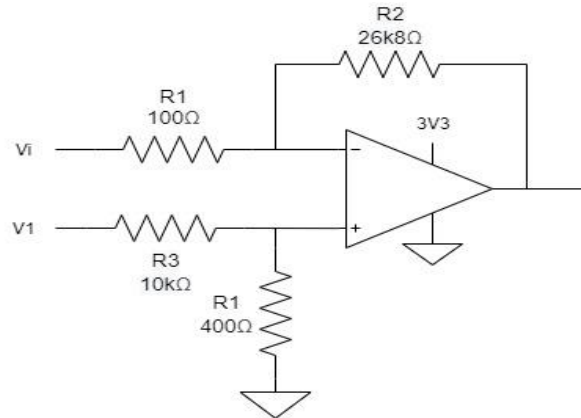


Figura 3 – Circuito de Acondicionamento projetado

De modo a ser possível saber que valores de temperatura estão fora da gama de medição, o sinal de saída da PT100 foi acondicionado para a gama de [0.15; 3.15] V e não para a gama total de tensão da ADC, [0; 3.3] V. O dimensionamento do circuito foi realizado a partir de uma reta do tipo $y = mx + p$, em que m é o ganho necessário para o sinal da PT100 ficar acondicionado na gama desejada de tensão e p a tensão na saída do circuito quando a entrada está a 0. Fazendo corresponder 115.541mV ao valor máximo da gama considerada e 127.075 ao valor mínimo, o parâmetro m foi obtido segundo o seguinte cálculo.

$$m = \frac{\Delta V_o}{\Delta V_{pt100}} \leq >$$

$$m = \frac{(0.15 - 3.15)V}{(127.015 - 115.541)mV} = -260 \text{ V/V}$$

Substituindo na reta $y = mx + p$, m pelo ganho obtido, y por 0.15 e x pelo valor da abcissa correspondente a 0.15 (127.015 m), é calculado o parâmetro p da reta (3).

$$p = 0.15 + (-260 * 127.015m) = 33.2$$

Assim, obtém-se a equação $V_o = -260 * V_i + 33.2$. Visto que o declive (m) é negativo e p é positivo o circuito é do “2º quadrante” e, como tal, tem a topologia da Figura 3. Definindo $\alpha = \frac{R_2}{R_1}$ e $\beta = \frac{R_4}{R_3}$ O próximo passo de dimensionamento é calcular os parâmetros α e β de modo a obter os valores das resistências. Partindo das seguintes relações foram obtidos.

$$\alpha = -m \leq > \alpha = 260$$

$$\beta = \frac{p}{(1 - m) * V_1 - p} \leq > \beta = 0.04$$

No cálculo de β , o valor da fonte de tensão V_1 usado foi de 3.3 V. Assumindo um valor de R_1 de 100 Ω , R_2 terá o valor de 26 k Ω (26.8 k Ω em valores standard E12). Assumindo um valor para R_3 de 10 k Ω , obteve-se para R_4 uma resistência de 400 Ω (390 Ω em valores standard E12). V_i corresponde ao sinal em tensão vindo da leitura da PT100 e o amplificador operacional utilizado foi de novo um MCP6022. Verificando o dimensionamento efetuado em termos de realização prática, visto que $|m| > \frac{p}{V_1} - 1 \leq > 260 > \frac{33.2}{3.3} - 1$, o circuito dimensionado é realizável.

Firmware

Interface com o utilizador via UART

O projeto em questão possui uma forte componente de interação com o utilizador, e para tal, é essencial o desenvolvimento de uma interface com o mesmo. Esta interface deve ser capaz de:

- mostrar ao utilizador os valores de temperatura instantânea e temperatura máxima desde o *power on* do sistema.
- adquirir valores de *setpoint* entre 40 e 70°C e uma ordem para desligar o aquecimento que equivale a adquirir

um *setpoint* de 0°C.

Para o primeiro ponto, após termos feito a aquisição e o cálculo dos valores de temperatura instantânea e máxima, realizamos apenas uma impressão destes valores no terminal via a UART1. Visto que interessa ao utilizador ter conhecimento destes valores de forma contínua, os valores são mostrados no terminal de forma periódica.

Para o segundo ponto, os valores de *setpoint* inseridos pelo utilizador são adquiridos usando uma rotina de serviço à interrupção da UART1, de modo a proceder-se à alteração do *setpoint* apenas quando desejado pelo utilizador, evitando parar a execução do programa em todas as iterações do ciclo de controlo. Os valores inseridos pelo utilizador de interesse são: entre 40 e 70°C e 0°C. Assim, sempre que o utilizador insere um carácter do teclado, é gerada uma interrupção e o programa vai para a rotina de interrupção para ser efetuada a leitura deste. Nesta rotina, o programa realiza um primeiro processamento, se o valor lido é um dígito, este é guardado numa estrutura de dados baseada numa *stack*. Quando a *stack* se encontrar cheia ou o utilizador inserir a tecla “enter”, uma *flag* é ativada. Esta *flag* tem a função de indicar que os dados estão prontos para processamento ao mesmo tempo que impede a rotina de interrupção de guardar mais dados na estrutura de dados.

Assim que os dados inseridos pelo utilizador se encontrem prontos para processamento, o programa executa a função “*print_UP*”. Nesta função são efetuados o *parse* e a validação dos dados inseridos na *stack*. Em primeiro, é retirado da *stack* o primeiro carácter a ser processado e de seguida é verificado se é um dígito. Caso seja um dígito e o único elemento presente na *stack*, é posteriormente verificado se este dígito corresponde ao número zero, único *setpoint* válido para um número com estas características, e de seguida, este é enviado para o controlador. Se a *stack* não estiver vazia, o programa verifica as condições análogas às anteriores. Verifica se é um dígito e se a *stack* está vazia. Caso isto se confirme, o programa faz a concatenação dos dois algarismos e verifica se este está entre os valores de 40 e 70, e de seguida envia para o controlador.

Para as operações anteriormente descritas, a *stack* tem implementada uma função para adicionar um elemento a esta, “*push*”, outra para retirar um elemento desta e devolvê-lo “*pop*” e uma última para verificar o estado desta (vazia, completamente cheia ou com espaço livre para adicionar elementos).

Todos os casos que não valores de 0°C ou entre 40 e 70°C correspondem a valores de *setpoint* inválidos e, portanto, se identificados pela função darão origem a uma mensagem de erro no terminal.

Na Figura 4, encontra-se um fluxograma que resume de forma gráfica o código efetuado para o segundo ponto descrito.

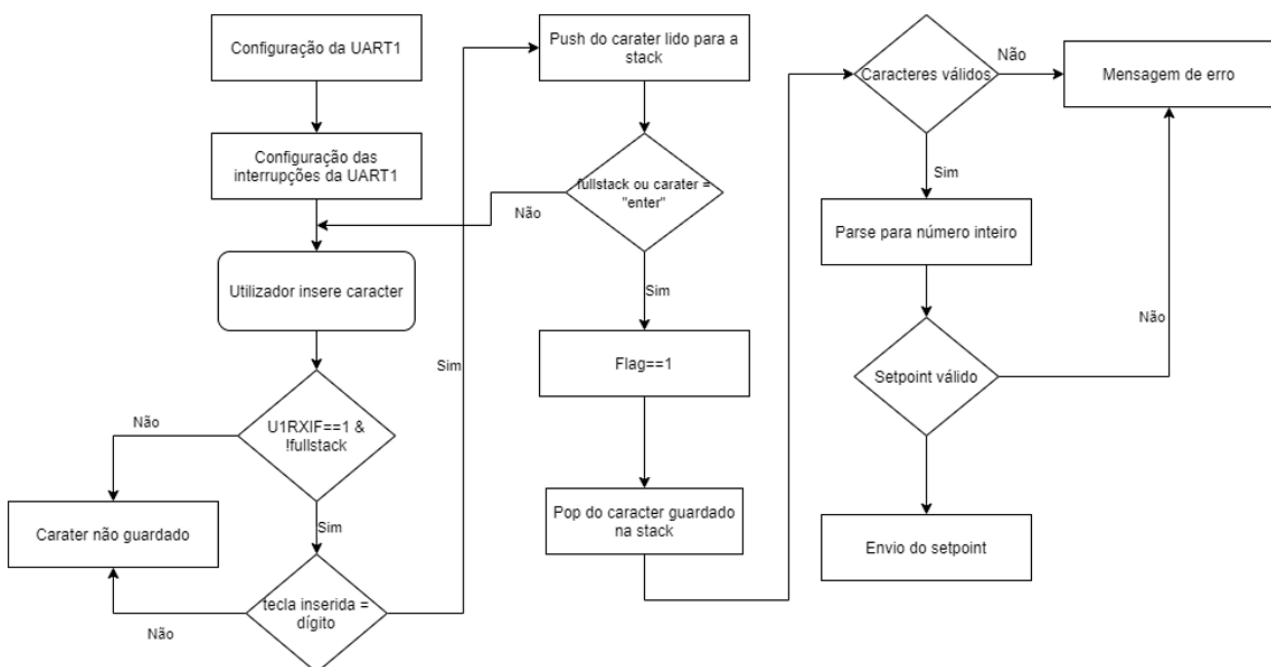


Figura 4 - Fluxograma da receção e validação de um *setpoint*

Amostragem da temperatura

O sensor utilizado neste projeto para medir a temperatura é uma sonda PT100. Tendo desenvolvido o hardware necessário para obter e acondicionar o sinal proveniente do sensor, é necessário um bloco de software capaz de processar o sinal para calcular a temperatura correspondente.

Sabemos que a função matemática que caracteriza o circuito de acondicionamento de sinal é:

$$V_{ADC} = -260 \times V_{PT100} - 33,2 \quad 6$$

em que V_{ADC} é a tensão à saída do circuito e que é lida pela ADC e V_{PT100} é a tensão à entrada do circuito proveniente do sensor PT100.

Como a ADC utilizada tem uma resolução de 10 bits, a função matemática que caracteriza a conversão da ADC é:

$$ADC_{VAL} = \frac{1024}{3,3} \times V_{ADC} \quad 7$$

em que ADC_{VAL} é o valor lido pela ADC.

Sabendo que a sonda PT100 é percorrida por uma corrente de valor igual a 1mA e juntando as equações 6 e 7, temos:

$$R_{PT100} = -0,0123948 \times ADC_{VAL} + 127,692 \quad 8$$

em que R_{PT100} é a resistência da sonda PT100.

Como o microcontrolador utilizado não possui unidade de processamento de vírgula flutuante, manipulou-se a equação 8 de modo a obter uma versão que permita utilizar valores inteiros sem perder precisão na medida:

$$(R_{PT100} \times 100) = -\frac{124 \times ADC_{VAL}}{100} + 12769 \quad 9$$

Tendo encontrado uma equação que permite calcular a resistência do sensor com base na leitura da ADC, é necessário converter esse valor para a temperatura correspondente. Tendo por base uma tabela com valores experimentais da resistência do sensor em função da temperatura, traçou-se um gráfico e verificou-se que os valores têm uma relação bastante linear. Assim, utilizou-se o método dos mínimos desvios quadrados para fazer uma regressão linear e obter a reta de ajuste dos pontos experimentais, como se pode observar na Figura 5 - Função temperatura / resistência e na equação 10.

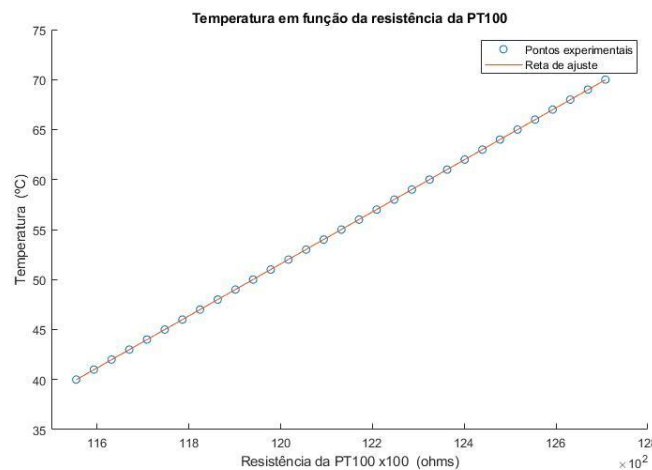


Figura 5 - Função temperatura / resistência

$$T = 0,026 \times (R_{PT100} \times 100) - 260 \quad 10$$

em que T é a temperatura.

Utilizando valores inteiros, obtém-se:

$$T = \frac{26 \times (R_{PT100} \times 100)}{1000} - 260 \quad 11$$

Com base nas equações 9 e 11 temos uma forma simples de calcular a temperatura através do valor lido pela ADC.

A amostragem da temperatura é realizada periodicamente utilizando um timer do pic32. Sempre que o timer termina a contagem, é gerada uma interrupção e incrementada uma variável global que funciona como um contador. Quando o contador atingir um determinado valor, é realizada a leitura da ADC e o cálculo da temperatura.

Controlador PI

Para colocar a temperatura no *setpoint* desejado pelo utilizador é necessário um controlador. Para este efeito, e seguindo as especificações do projeto, desenvolvemos um controlador do tipo proporcional-integrador (PI).

Este controlador calcula o sinal a aplicar ao atuador com base no sinal amostrado e é periódico devido à frequência de amostragem imposta no ponto anterior.

Partindo do modelo contínuo do controlador $Gr(s) = K * \left(1 + \frac{1}{sT_i}\right)$, em que K é o ganho e T_i corresponde à constante de integração do erro, é possível obter a expressão para este controlador no modelo discreto, obtendo-se a expressão $Gr(q) = \frac{s_0 + s_1 * q^{-1}}{1 - q^{-1}}$, no qual $s_0 = K * \left(1 + \frac{h}{T_i}\right)$ e $s_1 = -K$, e em que h é o intervalo de amostragem. A função de controlo deste controlador é dada por $u(k) = \frac{s_0 + s_1 * q^{-1}}{1 - q^{-1}} * (r(k) - y(k))$, em que $r(k)$ corresponde ao *setpoint* e $y(k)$ ao sinal à saída, conforme o esquematizado na Figura 6. A subtração do sinal $r(k)$ pelo sinal $y(k)$, pode ser definido como o sinal de erro ($e(k)$), assim, fazendo $e(k) = r(k) - y(k)$, e substituindo na expressão anterior do controlador, obtemos uma nova função de controlo definida por $u(k) = \frac{s_0 + s_1 * q^{-1}}{1 - q^{-1}} * e(k)$. Desenvolvendo esta última, chegamos à expressão usada no algoritmo para o sinal de controlo: $u(k) = u(k-1) + s_0 * e(k) + s_1 * e(k-1)$, computacionalmente bastante eficiente. Os resultados foram obtidos com recurso à referência bibliográfica [8], onde se encontra também a demonstração completa dos resultados matemáticos obtidos.

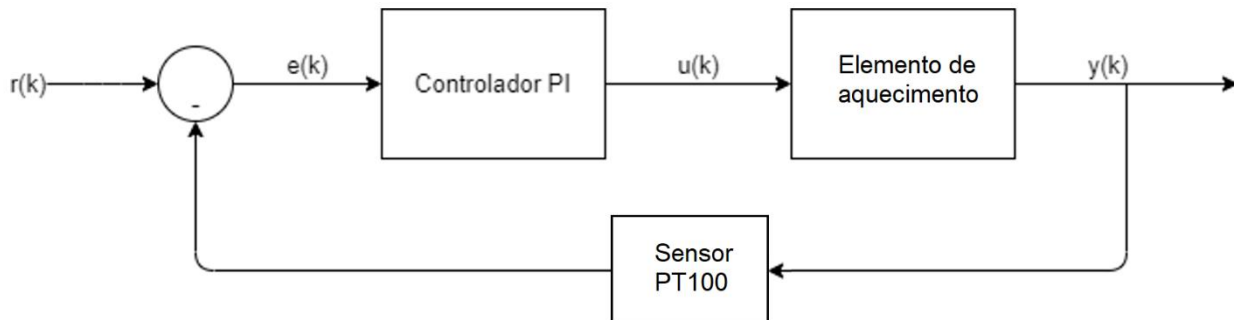


Figura 6 – Representação geral do controlador implementado

O algoritmo segue os seguintes pontos de desenvolvimento:

- definição dos parâmetros do controlador: h, K e T_i ;
- declaração das variáveis para o sinal de controlo (em k e k-1), referência, erro (em k e k-1) e saída a 0;
- cálculo dos parâmetros s_0 e s_1 ;
- em ciclo infinito, igualar o sinal de referência ao *setpoint* inserido pelo utilizador, igualar o sinal de saída à temperatura e cálculo do sinal de erro em k;
- cálculo do sinal de controlo e igualar os sinais de controlo e erro em k-1 aos sinais de controlo e erro em k,

respetivamente;

- Saturar o sinal de controlo. Sendo o número de passos do sinal PWM de 256, o sinal de controlo é saturado com valores entre 0 e 255.
- Enviar sinal de controlo como argumento da função que gera o sinal PWM a ser enviado para o circuito de aquecimento.

É, então, necessário obter os parâmetros h , K e T_i para a melhor resposta possível. Com melhor resposta possível compreende-se que a saída consegue chegar ao *setpoint* desejado ajustando o parâmetro K , e de seguida o erro desta em regime estacionário tem de ser nulo, para o *setpoint* ser exatamente o definido pelo utilizador, através do ajuste do parâmetro T_i . Visto que quanto maior o termo integrador pior será o regime transitório, ou seja, este faz com que o sistema em questão demore mais tempo a estabilizar no *setpoint* desejado. Assim, o parâmetro T_i , deve ser dimensionado de modo a garantir que o erro em estado estacionário seja nulo e que o sistema tenha uma resposta rápida o suficiente em regime transitório. Estes foram obtidos através de experimentação e análise da resposta do sistema aos diversos parâmetros aplicados.

Atuação sobre o elemento de aquecimento

Como mencionado anteriormente, para variar a quantidade de energia entregue ao elemento de aquecimento, é necessário gerar um sinal PWM com frequência 2kHz e *duty cycle* variável entre 0 e 100%.

Para isso, foi utilizado o timer 3 em conjunto com o *output compare* do pic32 para gerar o sinal PWM numa saída digital.

Resultados e Análise

Hardware

Ao nível de hardware foram efetuados testes dos circuitos projetados recorrendo ao simulador LTSPICE.

Circuito de Aquecimento

O circuito de aquecimento foi simulado conforme a topologia do circuito anteriormente descrito. Foram efetuadas simulações aos valores de corrente entregues ao elemento de aquecimento quando na base era aplicado um sinal de PWM com valores de *duty-cycle* de 20%, 50% e 80%, de modo a verificar a variação de corrente com o *duty-cycle* do sinal PWM. Os resultados apresentados estão representados na Figura 7.

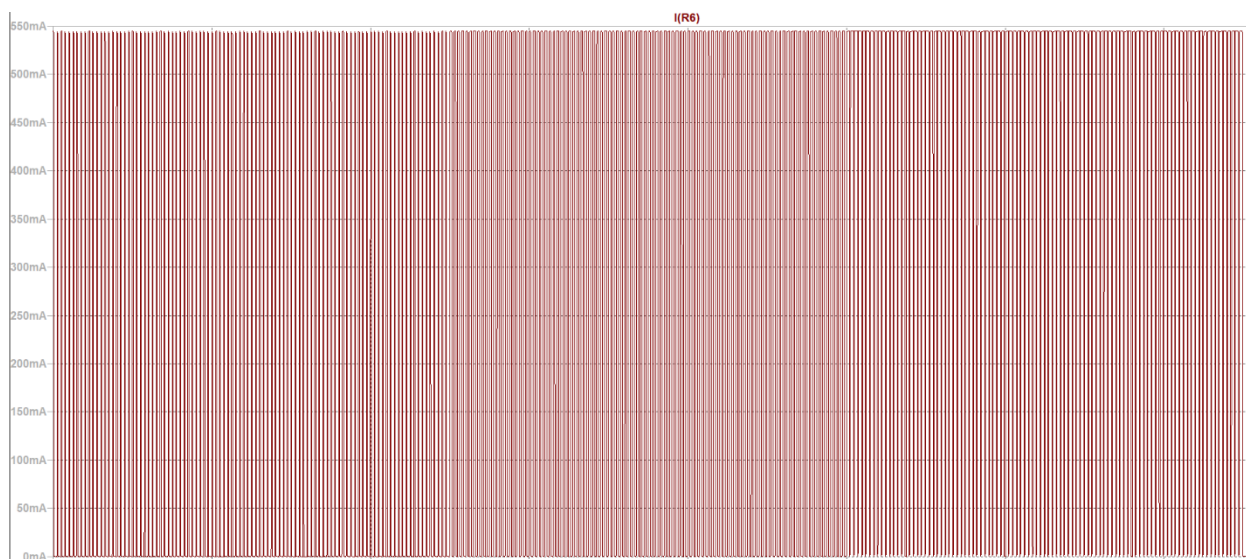


Figura 7- Forma de onda da corrente fornecida ao elemento de aquecimento

A Figura 7 demonstra que efetivamente o sinal em corrente varia o seu *duty-cycle* e, portanto, varia a corrente média aplicada ao elemento de aquecimento. Nos anexos (anexo 3) encontra-se uma outra simulação efetuada que sobrepõe os

sinais de PWM e a corrente na resistência de potência. Este foi incluído nos anexos, pois a sobreposição dos dois sinais prejudica a visualização correta dos mesmos. Interessa-nos agora quantificar os valores médios de corrente para o sinal de corrente obtido. Tal é representado na Tabela 2. A partir dos resultados obtidos nesta tabela, verifica-se que como esperado o valor médio da corrente entregue ao elemento de aquecimento varia de acordo com o *duty-cycle* do PWM, e portanto, os resultados obtidos encontram-se de acordo com o dimensionamento teórico.

<i>Duty-Cycle</i> (PWM) / %	Média I / mA
20	115.19
50	278.64
80	442.08

Tabela 2 – Valores de corrente média fornecida ao circuito de aquecimento

Fonte de Corrente

A fonte de corrente dimensionada foi simulada para o valor limite de carga para a gama de medição pretendida, visto que se o funcionamento do circuito para esta carga máxima for válido, para cargas inferiores o mesmo se verifica. Como dito anteriormente, o valor máximo de carga é $127.075\ \Omega$ que corresponde à resistência da PT100 para um valor de temperatura de 70°C . Simulando o circuito para a carga referida no simulador, obteve-se os resultados da Figura 8. Este gráfico apresenta no eixo dos y a corrente na carga em mA e no eixo das abcissas a escala de tempo usada na simulação em ms.

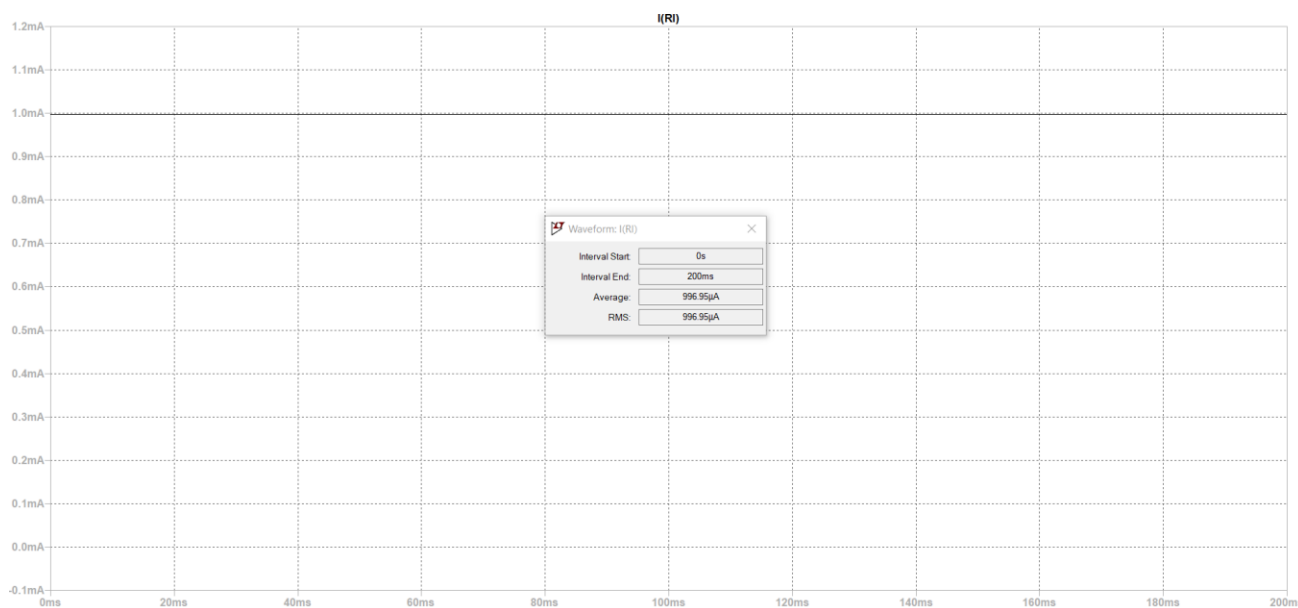


Figura 8 – Corrente fornecida à resistência da PT100 que corresponde a 70°C

O resultado obtido de corrente entregue pela fonte de corrente para uma resistência de carga de $127.075\ \Omega$ foi de $996.95\mu\text{A}$, o que se demonstra bastante próximo do 1mA desejado e, portanto, o funcionamento deste circuito encontra-se verificado com êxito para as especificações deste problema.

Circuito de Acondicionamento de Sinal

A simulação do circuito de acondicionamento foi efetuada colocando no sinal de entrada uma onda sinusoidal com o mesmo offset DC (0.121308 V) e a mesma amplitude de sinal (0.005767 V) dos valores que o sinal analógico à saída da PT100 poderia tomar e, de seguida, observar a saída do circuito. Os resultados obtidos para o sinal à entrada e respetivo sinal à saída encontram-se representados na Figura 9 e Figura 10, respetivamente.

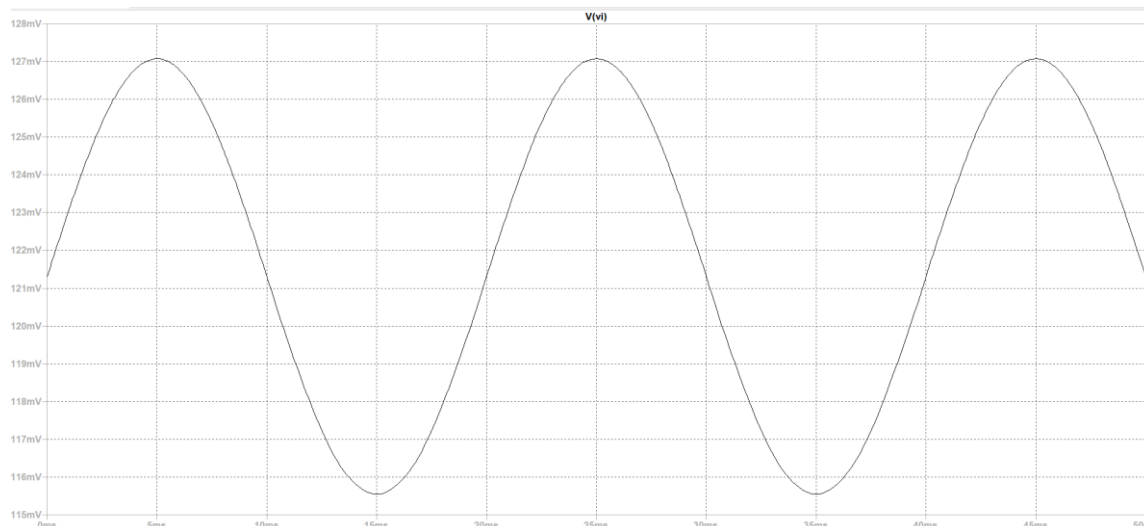


Figura 9 – Sinal aplicado à entrada do circuito de acondicionamento

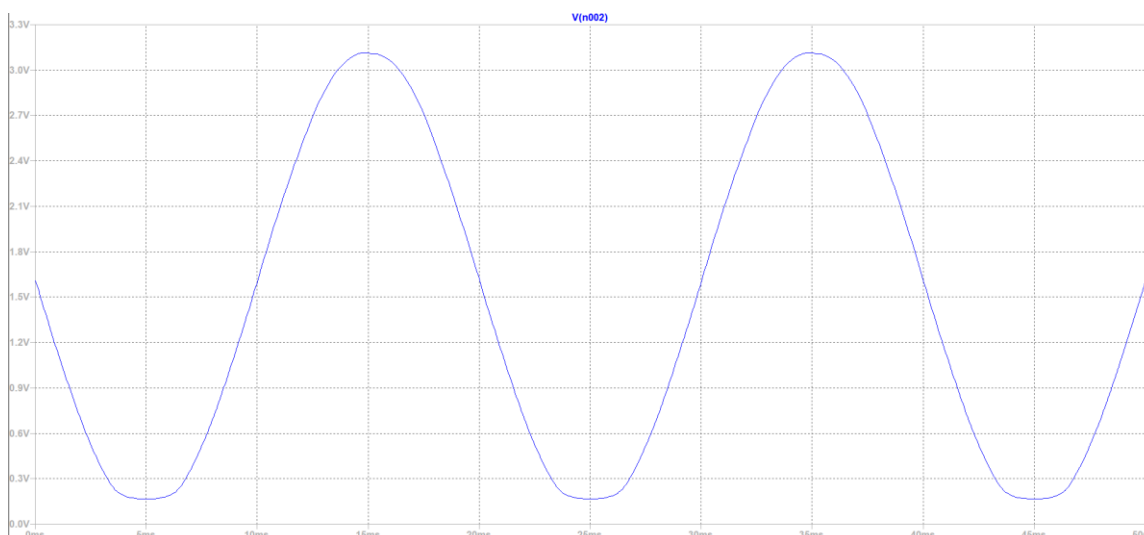


Figura 10 - Sinal à saída do circuito de acondicionamento

Em ambas as figuras, no eixo das ordenadas encontra-se representado o sinal em tensão (no caso da entrada em mV e da saída em V) em ordem ao tempo de simulação. Os resultados obtidos estão de acordo com o dimensionamento teórico efetuado. O sinal à saída do circuito simulado encontra-se próximo da gama ao qual foi dimensionado. Com aproximações da janela dos resultados no simulador o valor máximo do sinal à saída é de aproximadamente 3.11 V e o valor mínimo próximo dos 165 mV, próximos dos valores de projeto. Observa-se ainda uma inversão da fase dos sinais da entrada para a saída, que se deve ao facto de o circuito de acondicionamento usado apresentar uma configuração não-inversora.

Conclusão

Após análise dos resultados obtidos, verifica-se que o sistema desenvolvido cumpre com os requisitos presentes no problema proposto.

O trabalho desenvolvido teve impacto positivo na medida em que deu aos alunos competências no trabalho com interfaces de potência, leitura de sensores de temperatura do tipo PT100 e implementação de algoritmos de controlo em

microcontroladores. Permitiu ainda o desenvolvimento de boas práticas de programação em C e em microcontroladores e, findado este projeto os alunos têm módulos base já configurados e prontos para utilização em projetos futuros. Infelizmente não foi possível validar o correto funcionamento do sistema em ambiente real, devido à atual situação de pandemia, no entanto fez-se um esforço para validar o funcionamento dos principais blocos com recurso a simulação. Assim, conclui-se que o projeto foi finalizado com êxito.

Referências

- [1] Pedro Fonseca. “Guia para a redação de relatórios”. Departamento de Eletrónica, Telecomunicações e Informática. Universidade de Aveiro. 2012.
- [2] Pedro Fonseca. “Lab assignments guide”. Departamento de Eletrónica, Telecomunicações e Informática. Universidade de Aveiro. 2020.
- [3] Pedro Fonesca, “Sistemas de Instrumentação Eletrónica” apontamentos. Departamento de Eletrónica, Telecomunicações e Informática. Universidade de Aveiro. 2020.
- [4] Pedro Fonseca, Paulo Pedreiras. “Temperature Slides”, slides aulas teóricas “Sistemas de Instrumentação Eletrónica”. Departamento de Eletrónica, Telecomunicações e Informática. Universidade de Aveiro. 2020.
- [5] LabFacility. Datasheet: Temperature & Process Control. PT100 datasheet. Electronic Publication.
- [6] Alexandre Mota, “Sistemas e Sinais Discretos”, apontamentos. Departamento de Eletrónica, Telecomunicações e Informática. Universidade de Aveiro. 2019.

Anexos

Anexo 1: Diagramas da arquitetura do sistema



Figura 11- Arquitetura nível 0 do sistema

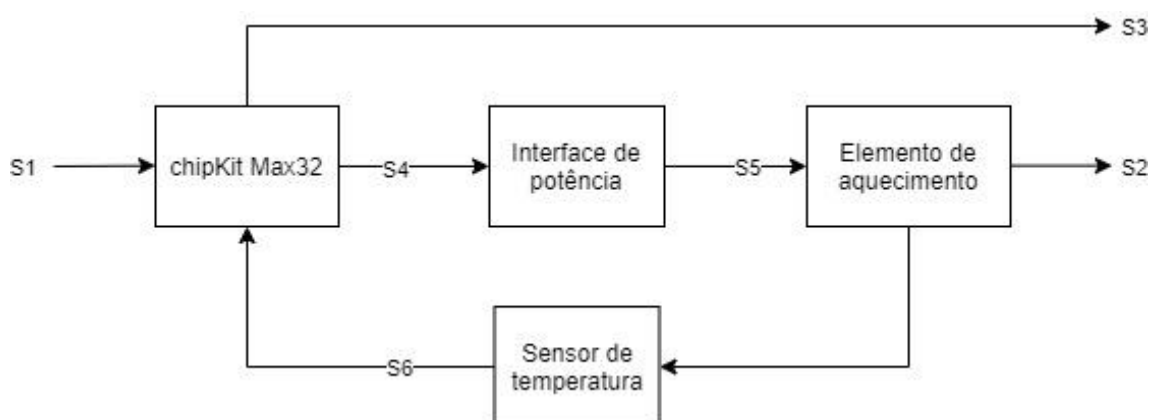


Figura 12 - Arquitetura nível 1 do sistema

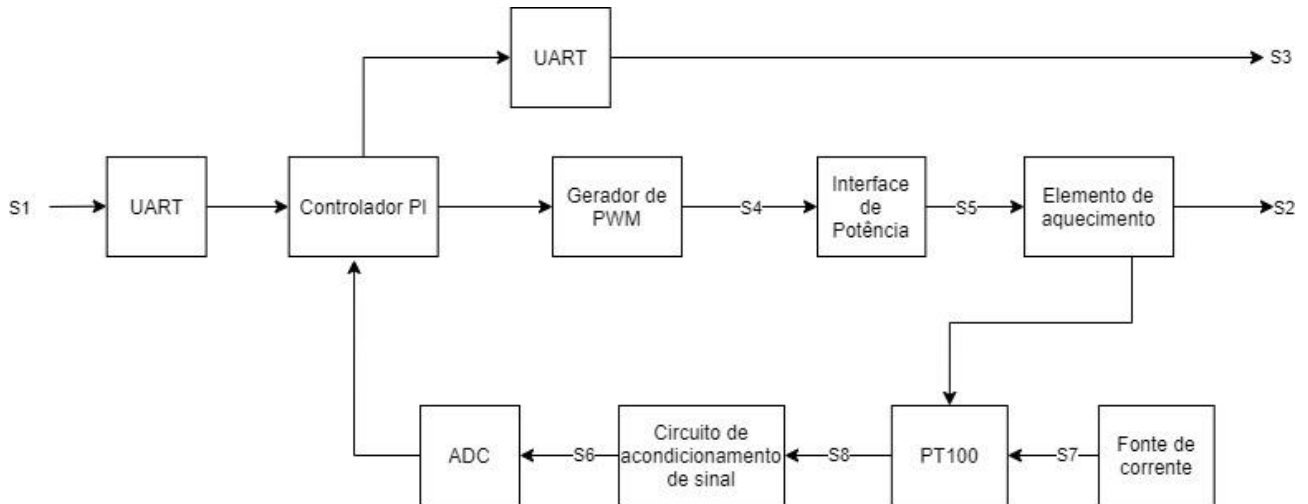


Figura 13 - Arquitetura nível 2 do sistema

Anexo 2: Esquema elétrico do sistema

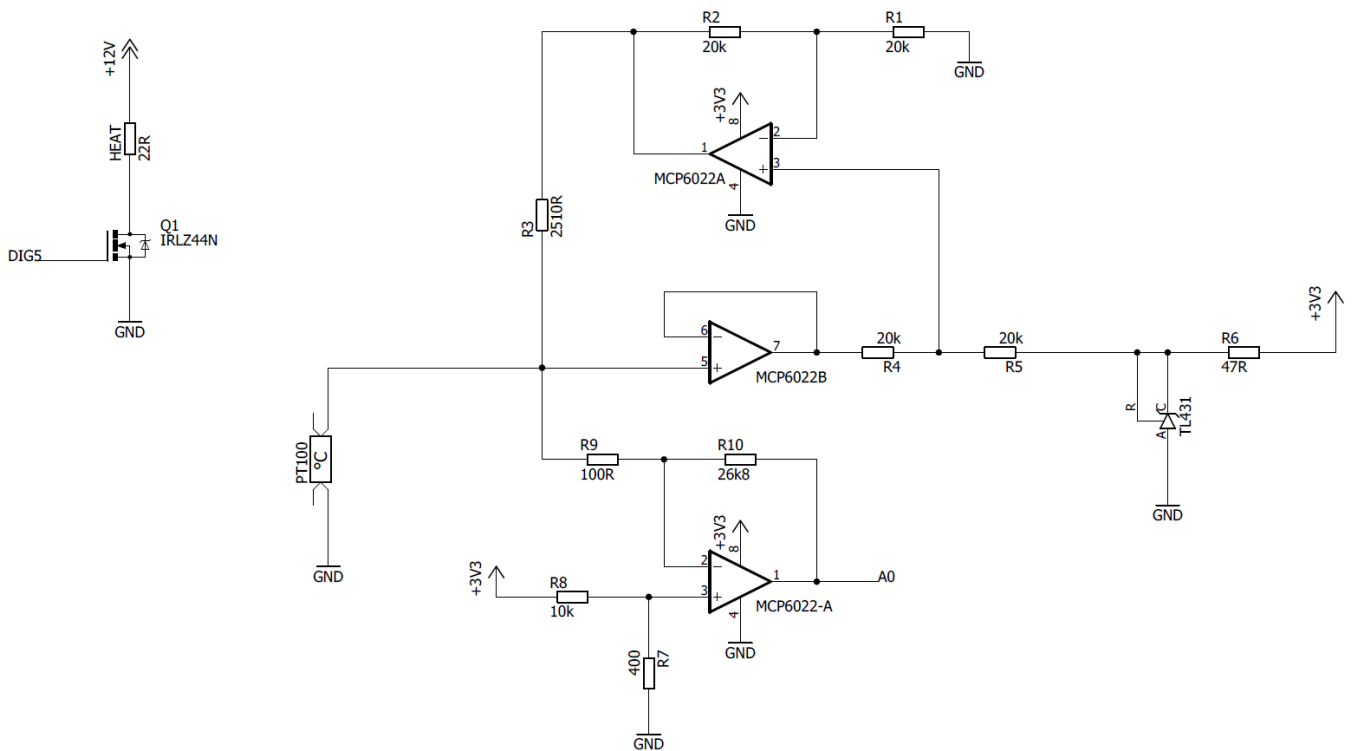


Figura 14 - Esquema elétrico do sistema

Anexo 3: Suporte aos resultados de simulação do circuito de aquecimento

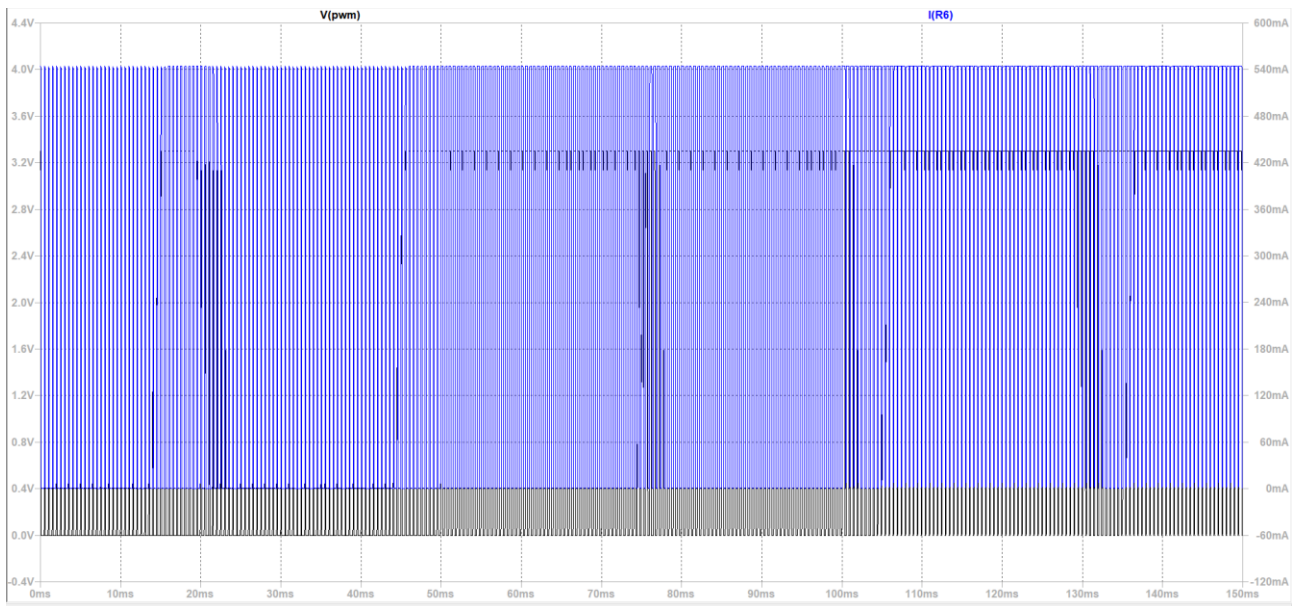


Figura 15- Sinais PWM (a preto) e corrente entregue ao elemento de aquecimento (a azul) sobrepostos