

Universidade Federal de Itajubá

Engenharia de Computação

Programação Embarcada e Laboratório de Programação Embarcada

Documento Técnico – ECOP04 e ECOP14

Simulação de Ar Condicionado

Aluno:

Felipe Souza Fernandes – Matrícula: 2020006453

1 de agosto de 2021



Felipe Souza Fernandes

Relatório do Projeto Final de Programação Embarcada

Simulação de Ar Condicionado

1 de agosto de 2021

Sumário

Resumo	4
1.Introdução	5
2.Desenvolvimento	5
2.1-Escolha do projeto	5
2.2-Funcionamento	6
3.Código	9
3.1-Bibliotecas utilizadas e funções criadas	9
3.2-Main	10
4.Dificuldades encontradas	10
5.Conclusão	11

Resumo

Neste documento técnico estará enunciado o processo de desenvolvimento do projeto final das matérias de Programação Embarcada (ECOP04) e Laboratório de Programação Embarcada (ECOP14). Para isso, foi utilizado ferramentas como MPLAB X (IDE), o qual foi utilizado para o desenvolvimento e implementação do programa, XC8, compilador utilizado, PICSimLab, simulador de um sistema embarcado, e a linguagem C, utilizada para escrever o código. O objetivo do projeto foi criar um simulador de ar condicionado, no qual haveriam as funções fan (ventilador), heat (aquecedor) e cool (resfriamento), com a utilização dos diversos componentes da placa como o heater e o cooler. Além disso, é possível escolher a potência e o tempo de funcionamento do ar condicionado.

Abstract

This technical document will state the process of development of the final project of the subjects known as Embedded Programming (ECOP04) and Embedded Programming Laboratory (ECOP14). In order to conclude this project, some tools were used, such as MPLAB X (IDE), used for the development of the program, XC8, name of the compiler used, PICSIMLAB, an embedded system simulator, and C language, an programming language used to write the code. The goal of this project was to create an air conditioning simulator, in which functions such as fan (fan), heat (heater) and cool (cooling) were added, using the several components of the board, such as the heater and the cooler. In addition, it's possible to choose the power and operating time of the air conditioner.

1. Introdução

Desde os princípios, o homem tenta, de alguma forma, controlar o mundo ao seu redor e, portanto, ser capaz de mudar algo quando desejar. Um desses fatores é a temperatura, essencial para a existência da humanidade. Sendo assim, através da tecnologia, no início dos anos 1900 surgiu um aparelho que seria capaz de realizar exatamente isso, o ar condicionado.

O Ar Condicionado é uma invenção que surgiu em meados do século XX e logo já era sonho de aquisição de parte da população. A capacidade de controlar a temperatura atraiu as diversas pessoas que buscavam uma maneira de esfriar o cômodo em um dia quente, ou, até mesmo, aquecer o quarto naqueles dias mais frios. Funcionando de maneira semelhante a uma geladeira, o ar condicionado possui um sistema de ventilação, no qual o ar do ambiente ou passa em um sistema de refrigeração interno para resfriar ou é aquecido por outros métodos.

Com o objetivo de simular esse aparelho essencial na casa de muitos, foi usada a placa PicGenios e o microprocessador pic18F4520. Aquela possui diversos componentes para representar as diversas funcionalidades do ar condicionado. Através do LCD, do display de 7-segmentos, do heater, do cooler, do potenciômetro, dos LEDs e dos relês é possível simular de maneira aceitável essa tecnologia, bastando, para isso, a implementação através da linguagem C e dos seus diversos recursos.

2. Desenvolvimento

2.1 – Escolha do projeto

A fim de decidir qual seria o projeto realizado, diversos pontos foram considerados, entre eles a utilidade do produto para a sociedade, a dificuldade de implementação do código para se aproximar o máximo do real, e, principalmente, a capacidade da placa em rodar o programa criado. Dessa forma, surgiu a ideia de simular um ar condicionado, já que esse atende todas as medidas citadas anteriormente.

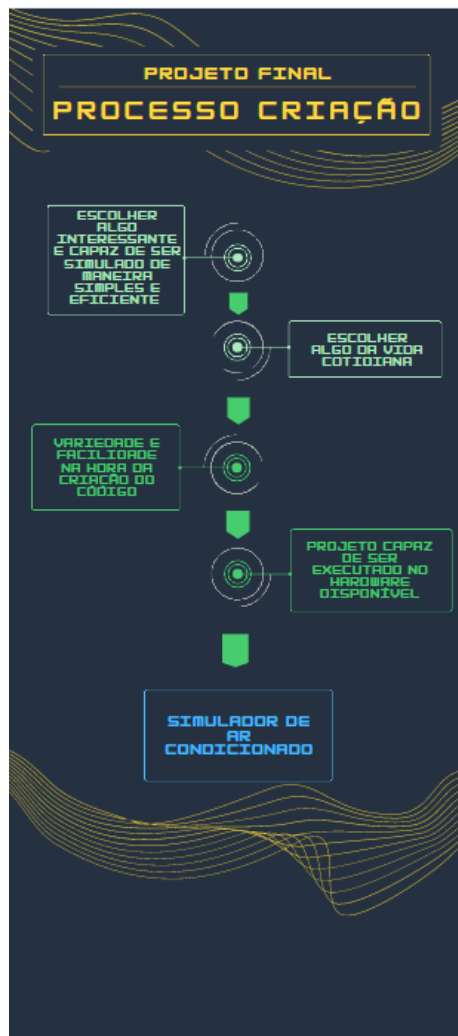


Figura 1 – Fluxograma – Processo de Criação

2.2 – Funcionamento

O ar condicionado criado através do programa é, em um primeiro momento, inicializado, que pode ser notado pelo ligamento e desligamento dos LEDs de forma sequencial de cima para baixo e da mensagem programada para aparecer no LCD. Após isso, chega-se à tela principal do simulador, no qual o usuário deverá escolher, através do teclado, entre as três opções: Fan, Heat, Cool. A escolha resultará em caminhos semelhantes, entretanto diferentes. Para simular o carregamento da opção, o primeiro relê ligará e desligará três vezes seguidas seguido pelo segundo relê.

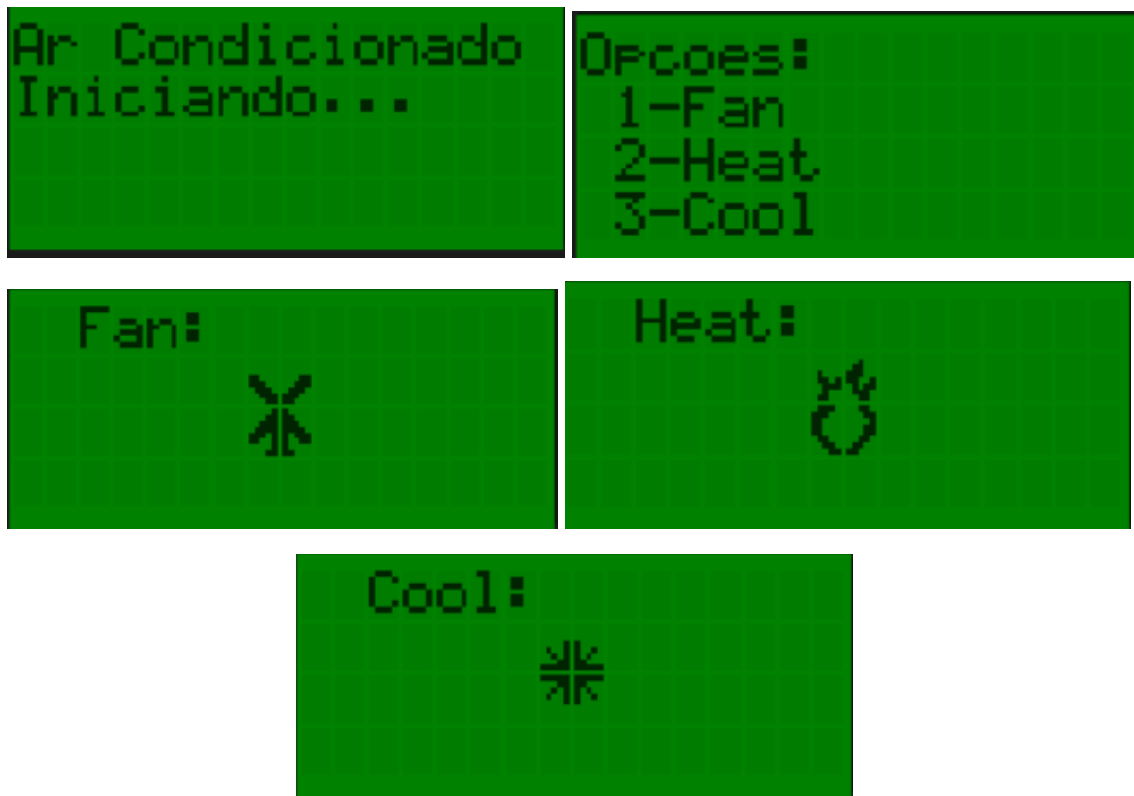


Figura 2 – Telas iniciais com resultado da opção escolhida

Após a escolha principal, é necessário escolher a potência através do potenciômetro que varia entre três forças: Low (L), Medium (M) e High (H). Essa decisão tem resultados diferentes dependendo da escolha feita anteriormente. Para o caso do Fan, essa escolha é meramente ilustrativa. Já para o Heat, potencias mais altas significam que o heater fará com que a temperatura suba mais rapidamente. Por último, para o Cool, a escolha altera a potência que o pwm manda para o cooler, variando sua força.



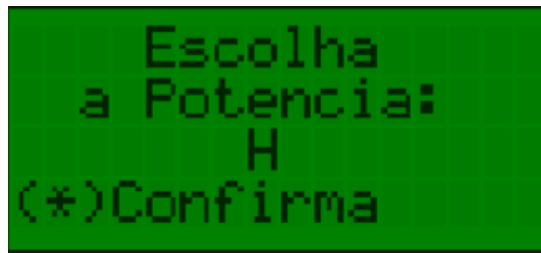


Figura 3 – Escolha da potência

Por último, o usuário escolherá o tempo de funcionamento do equipamento, sendo capaz de escolher entre três opções que são: 2 minutos, 5 minutos e 10 minutos. Em seguida, o processo iniciará, o que poderá ser observado através do display de 7-segmentos, que exibirá o tempo restante, e do cooler e/ou do heater, que estarão ligados ou desligados de acordo com as escolhas. Dessa forma, o programa finaliza, exibindo uma última mensagem no LCD.

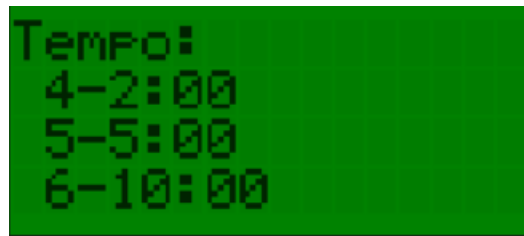


Figura 4 – Escolha do tempo

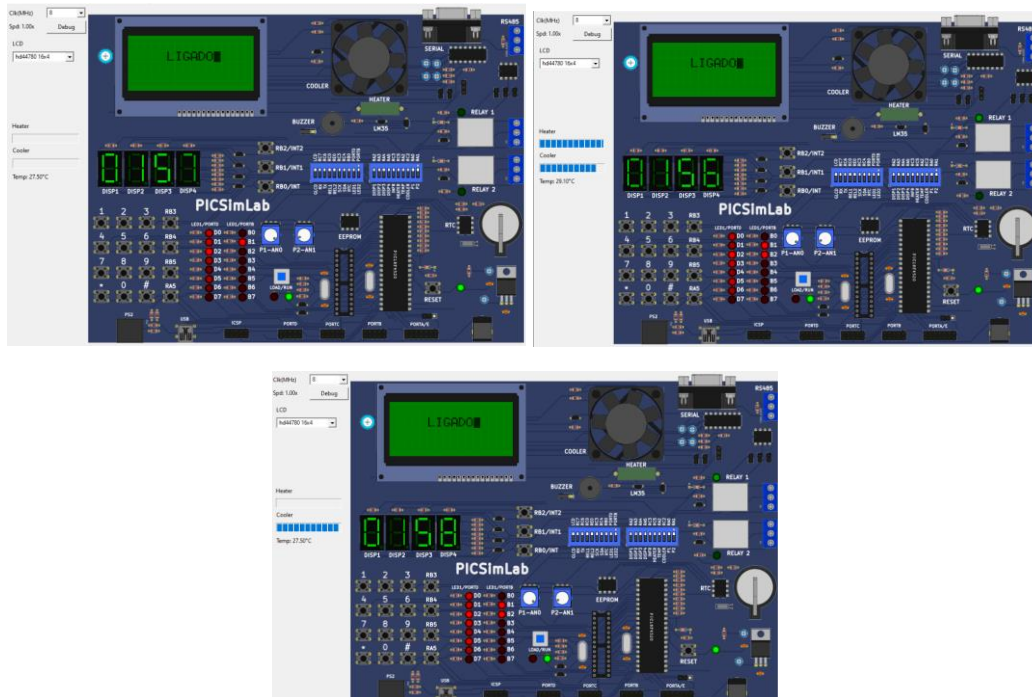


Figura 5 – Funcionamento do Fan, Heat e Cool, respectivamente

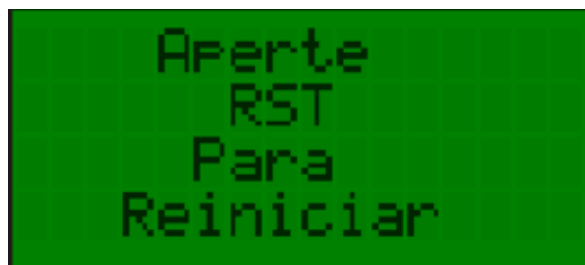


Figura 6 – Tela final

3. Código

3.1 – Bibliotecas utilizadas e funções criadas

Foram utilizadas diversas bibliotecas para o bom funcionamento do programa, entre elas é importante citar: `pic18f4520.h`, `config.h`, `bits.h`, `ssd.h`, `keypad.h`, `lcd.h`, `atraso.h`, `adc.h`, `pwm.h`, `io.h` e `itoa.h`.

Para ajudar durante a execução do programa foram criados os arquivos “funções.h” e “funções.c”. No primeiro, foram declaradas as funções que foram implementadas no segundo. No total foram criadas 11 funções que são as seguintes:

- `ConfiguraLed` – configuração inicial do PORTD para o funcionamento dos LEDs.
- `LigarLed` – liga todos os LEDs de cima para baixo.
- `DesligarLed` – desliga todos os LEDs de cima para baixo.
- `ImprimeLogo` – recebe e imprime um logo (ventilador, fogo, floco de neve) no LCD.
- `temperatura` – exibe a temperatura lida através do `adc_amostra` no LCD.
- `desligaDisplay` – desliga o display de 7-segmentos.
- `menuTempo` – exibe no LCD informações como ligado e desligado e no display de 7-segmentos o tempo restante de funcionamento do programa.
- `aquecimento` – simula a função `heat` de um ar condicionado, com a ativação do heater e do cooler (para evitar que es quente muito).
- `resfriamento` – simula a função `cool` de um ar condicionado, com a ativação do cooler.
- `potencia` – exibe o menu para a escolha da potência e recebe a informação através do potenciômetro e do teclado.
- `tempo` – exibe o menu para a escolha do tempo de funcionamento do programa e recebe a informação através do teclado.
- `rele` – liga e desliga os reles 1 e 2.

3.2 – Main

A função main é bem simples, sendo responsável pela chamada das funções na ordem correta de forma a simular um ar condicionado real.

Nessa função são criados os logos que são utilizados durante o programa e as diversas variáveis que serão utilizadas ao decorrer desse. Ela é responsável pelas mensagens de iniciação do ar condicionado e da chamada dos LEDs. Em seguida, ela exibe o principal menu do programa, no qual o usuário escolherá entre as opções Fan, Heat e Cool através do teclado. Dependendo da escolha a main chamará as funções específicas para cada. Por fim, é responsável por exibir no LCD a tela final e finalizar ou reiniciar o programa.

O arquivo contendo a main e demais bibliotecas e funções utilizadas durante a execução do programa estará disponibilizado no GitHub.

4. Dificuldades encontradas

No desenvolvimento do programa diversas dificuldades foram encontradas, entre elas o funcionamento do cooler e do heater, da leitura da temperatura e a execução simultânea de diversos processos/componentes.

O funcionamento do cooler e do heater foi um desafio durante a implementação, entretanto, era necessário, uma vez que o programa se baseia em um ar condicionado. Inicialmente, o programador não sabia ativar e desativar esses componentes, mas, através do código disponibilizado pelo PICSimLab, esse problema inicial foi resolvido. Após isso surgiram outros problemas, como a temperatura que subia muito com a ativação apenas do heater, que foi resolvido com a ativação conjunta do cooler em uma potência menor (pwm). Além disso, a temperatura da placa, devido à um limite imposto pelo PICSimLab, não descia abaixo de 27,5 C. Esse problema não teve uma solução ideal, uma vez que a temperatura que aparece após o encerramento da opção Cool é meramente ilustrativa.

A leitura da temperatura foi outra intempérie durante a realização do projeto final, uma vez que a função `adc_amostra` recebia um valor de temperatura que era exibido no LCD como 0 C. Esse problema foi resolvido, facilmente, com a mudança do parâmetro enviado na função citada anteriormente.

Por último é importante citar a execução simultânea de componentes, principalmente do display de 7-segmentos com os demais. Para a resolução desse problema foi necessário apenas a organização e chamada das funções necessárias, obtendo um resultado aceitável.

5. Conclusão

A conclusão do projeto vem acompanhada de um sentimento de dever cumprido, mesmo levando em consideração os diversos problemas e as horas investidas. O resultado final foi bastante satisfatório, uma vez que foi capaz de simular de maneira interessante o funcionamento de um ar condicionado.

A realização do projeto final foi uma ótima maneira de colocar em prática os conhecimentos obtidos através das matérias de Programação Embarcada e Laboratório de Programação Embarcada. Por meio do surgimento de problemas e resolução desses foi possível aproximar mais do que seria o desenvolvimento de um sistema embarcado na vida real, sendo, portanto, uma experiência que acrescentou muito ao profissional.