

*Vinicius Silva Gonçalves*

## ***Simulação do relógio digital para treino***



**UNIFEI**  
Universidade Federal de Itajubá/MG

**Brasil**

**2021**

## **OBJETIVO**

Este relatório tem por objetivo apresentar um passo a passo de como funciona e se usa o relógio digital para treino

# INTRODUÇÃO

## Relógio digital para treino

O projeto desenvolvido tem por objetivo simular as funções básicas de um relógio digital usado para treinos, algo parecido com a mi band 1 da fabricante Xiaomi. Neste projeto as funções que seriam acionadas via Touche são acionadas pelo teclado do simulador PICSIMLab, as funções acionadas são a de iniciar um cronometro após ter um sinal sonoro para a pessoa se preparar e acionar alguns leds para indicar que o tempo está rodando, caso o mesmo botão de acionamento seja apertado o tempo é zerado, desligar o led e parar o cronometro mostrando o tempo de corrida no LCD, mostrar a hora atual, mostrar a quantidade de passos e a frequência cardíaca da pessoa.

### Componentes:

- Display de 7 segmentos
- Leds
- Buzzer
- Teclas
- Lcd

### Ferramentas usadas:

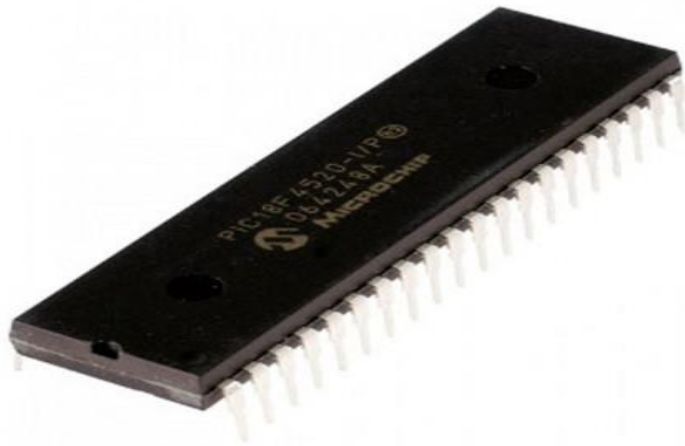
PicsIMLab – Link para Download:

[https://sourceforge.net/projects/picsim/files/v0.7.5/picsimlab\\_0\\_7\\_5\\_win64\\_setup.exe/download](https://sourceforge.net/projects/picsim/files/v0.7.5/picsimlab_0_7_5_win64_setup.exe/download)

MPLAB - Link para Download: <https://www.microchip.com/en-us/development-tools-tools-and-software/mplab-x-ide>

Compilador MPLAB XC C – Link para Download: <https://www.microchip.com/en-us/development-tools-tools-and-software/mplab-xc-compilers>

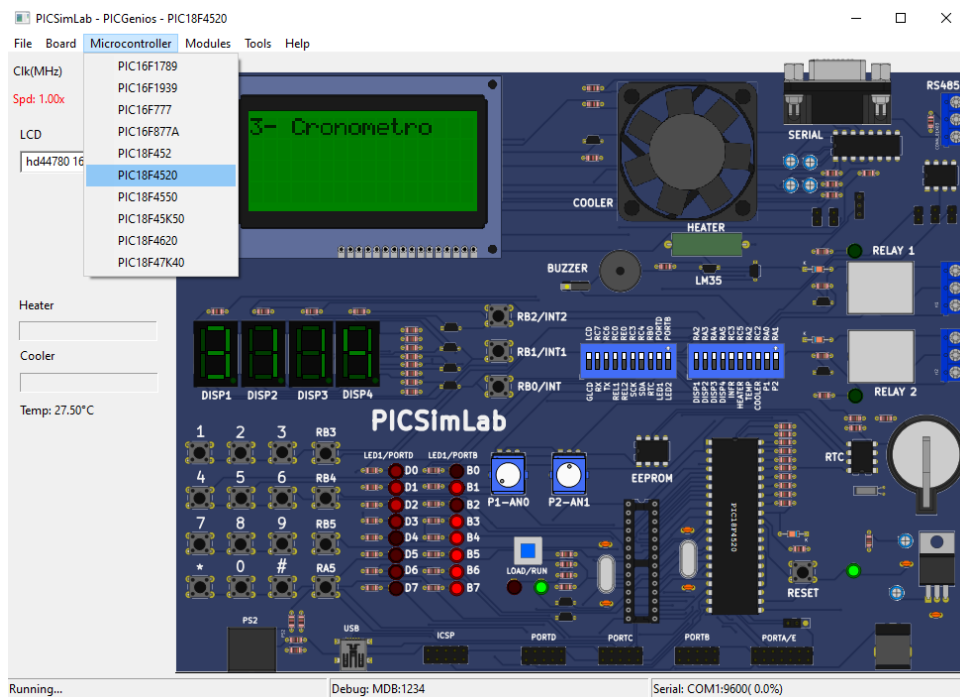
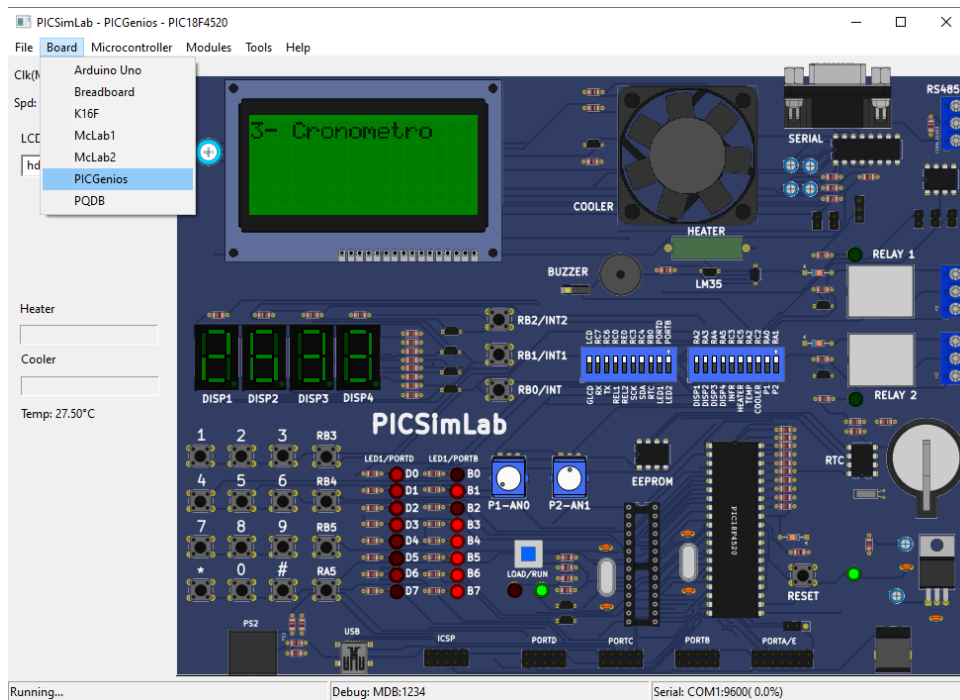
## Simulação do Hardware



**Imagem 1: Microcontrolador PIC18F4520**

Neste projeto foi utilizado o microprocessador PIC18F4520 e a placa PICGenios que foi simulado através do PICSIMLab versão 0.7.5.

A configuração do PICSIMLab é feita selecionando no canto superior esquerdo a opção "Board", em seguida selecionando a opção "PICGenios" e após isso "Microcontrolador", seguindo esses passos é possível selecionar o microprocessador que deseja, lembrando que nesse projeto foi utilizado o PIC18F4520. É preciso também expandir o painel LCD, isso é feito clicando na opção lateral "LCD" e selecionando o "hd44780 16x4".



**Imagem 2: Configurando a placa para simular o projeto**

## Software

A compilação e a edição do código foram feitas na IDE MPLAB, para configurar o projeto, é preciso abrir o MPLAB, clicar em “File” e em seguida “New Project”, após isso deve aparecer a opção “Microchip Embedde”, esta deve estar com “Standalone Project” selecionado. Logo em seguida selecione a “PIC18F4520” em “Select Tool” “PICSimLab”, então é só clicar no “XC8”, podendo assim

escolher um nome para o projeto, para finalizar basta selecionar “Set as main project” e o projeto é criado.

Para rodar o Relógio digital treino é preciso baixar o projeto pelo GitHub, clicar com o botão direito em “Header Files”, em seguida em “Add existing item”, buscar na pasta onde foram instalados os arquivos do GitHub e selecionar todos as bibliotecas (arquivos “.h”), verificando se a função “copy” está selecionada, caso contrário o usuário estará editando os arquivos “.c”, logo após fazer isso deve se clicar em “Build Main Project” para construir o projeto.

Após isso, no PICSImLab, é preciso selecionar “File” no canto superior esquerdo e em seguida “Load Hex”, uma janela será aberta, após isso o usuário deve ir até a pasta onde o MPLAB criou o programa e o salvou, então ele deve clicar na pasta com o nome do projeto que o próprio salvou, em seguida é necessário clicar na pasta “dist”, “default” e “production”, uma após a outro, após fazer isso, vai aparecer um arquivo “.hex”, este arquivo é o que deve ser selecionado para que ocorra a simulação no “PICSImLab”.

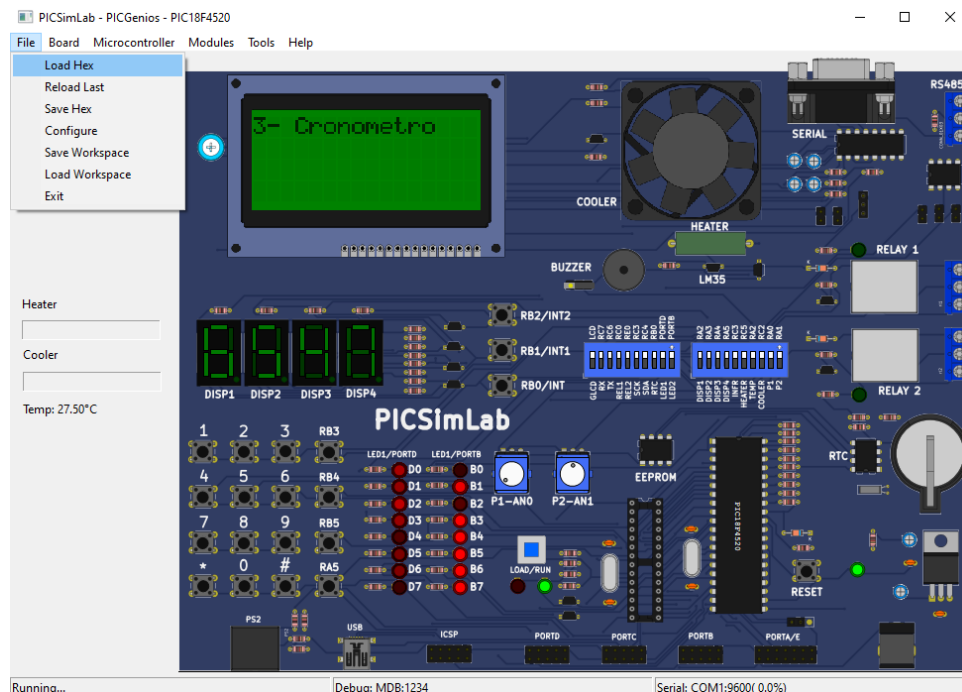
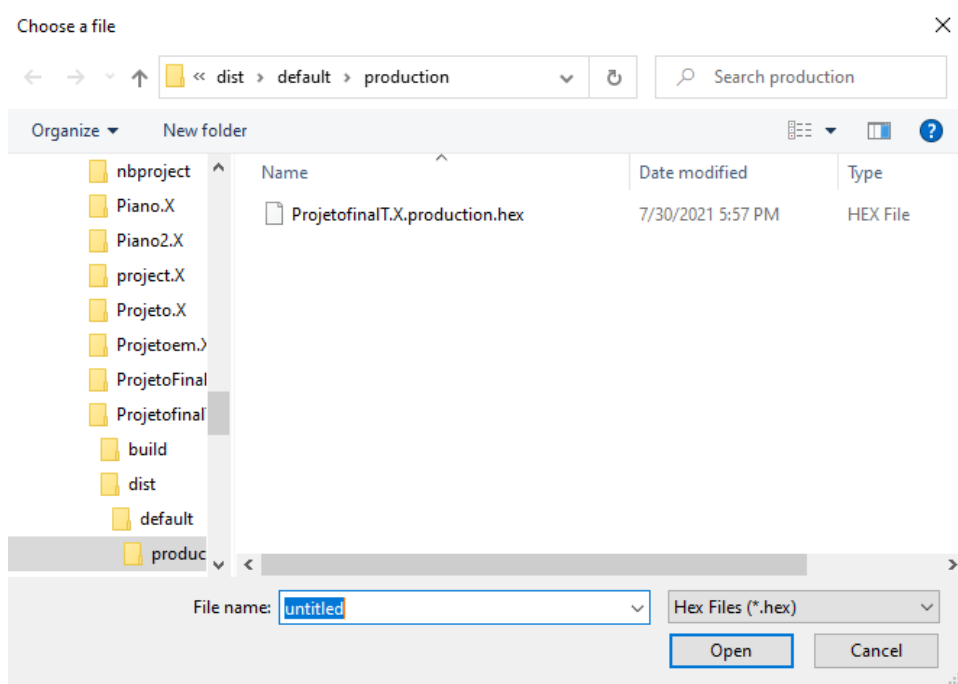
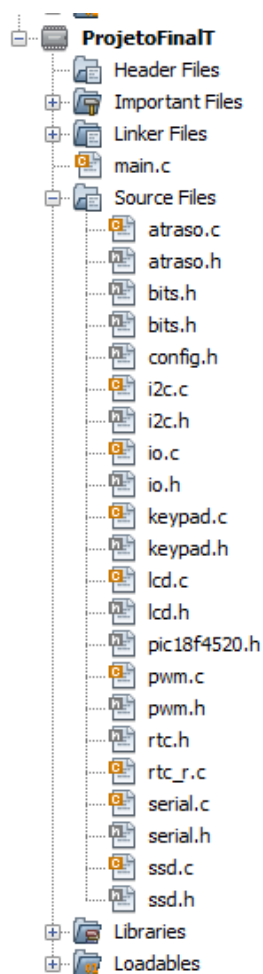


Imagem 3 – Executando o programa no PICSImLab



**Imagem 4 – Executando o programa no PicSimLab**



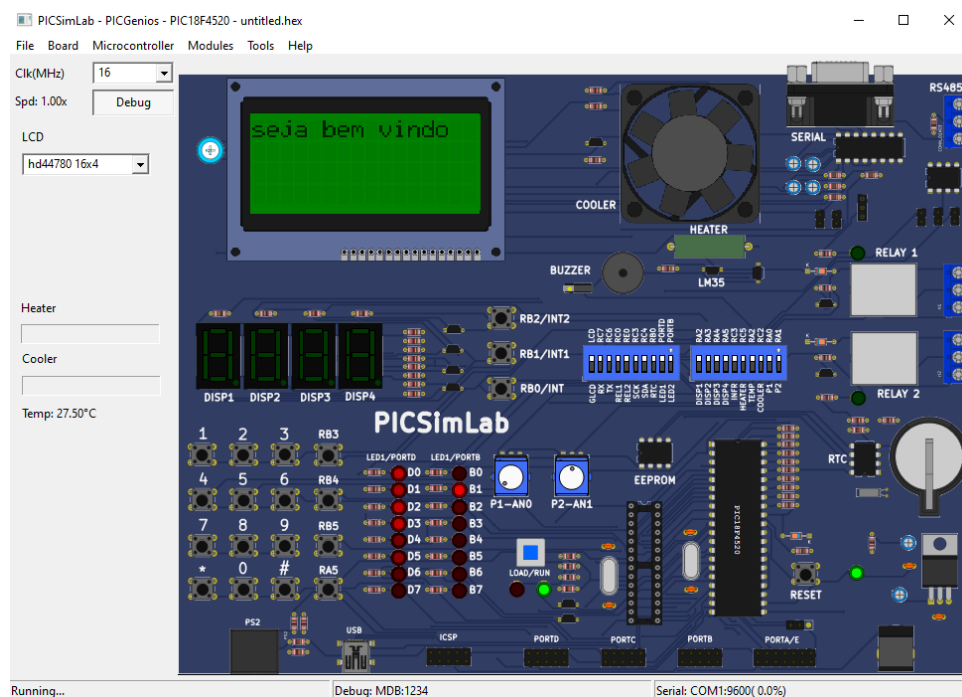
**Imagem 5 – Bibliotecas e arquivos.”h” e arquivos.”c”**

## Funcionamento

Todas os arquivos.”c” e arquivos.”h” usados são disponibilizados no SIGAA pelo professor da disciplina de ECOP04 da Universidade Federal de Itajubá.

Após chamar todas as bibliotecas usadas, é definido alguns nomes para o PORTD e para alguns pinos que são usados para representar o display de 7 segmentos, após isso aparece a main, onde são declaradas todas as variáveis e mensagens que serão utilizadas no código.

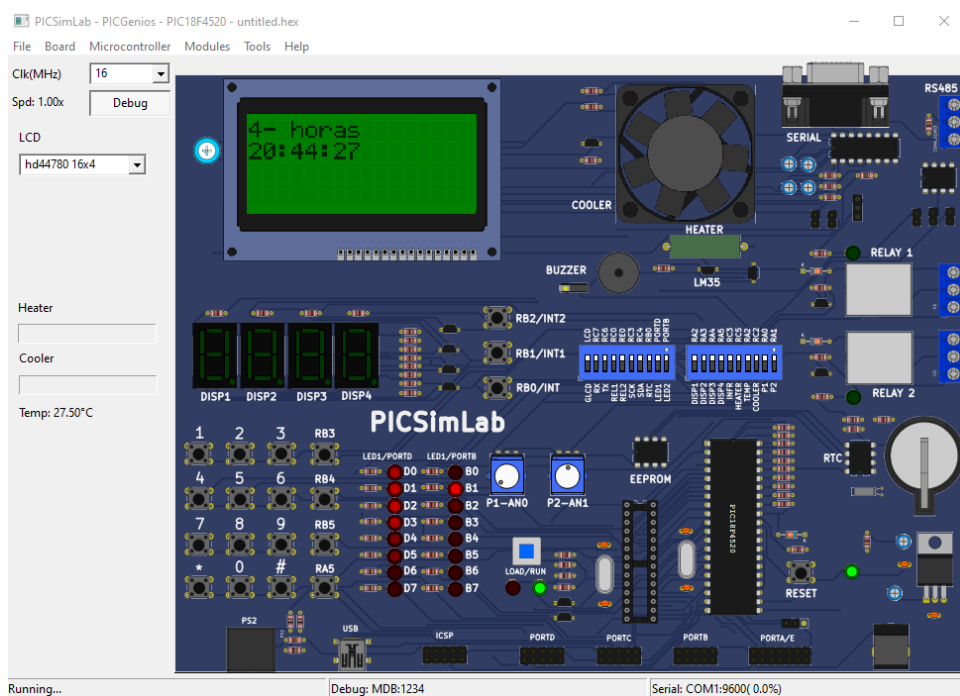
Ao executar o código aparece uma mensagem de boas vindas.



**Imagem 6 – mensagem de boas vindas .**

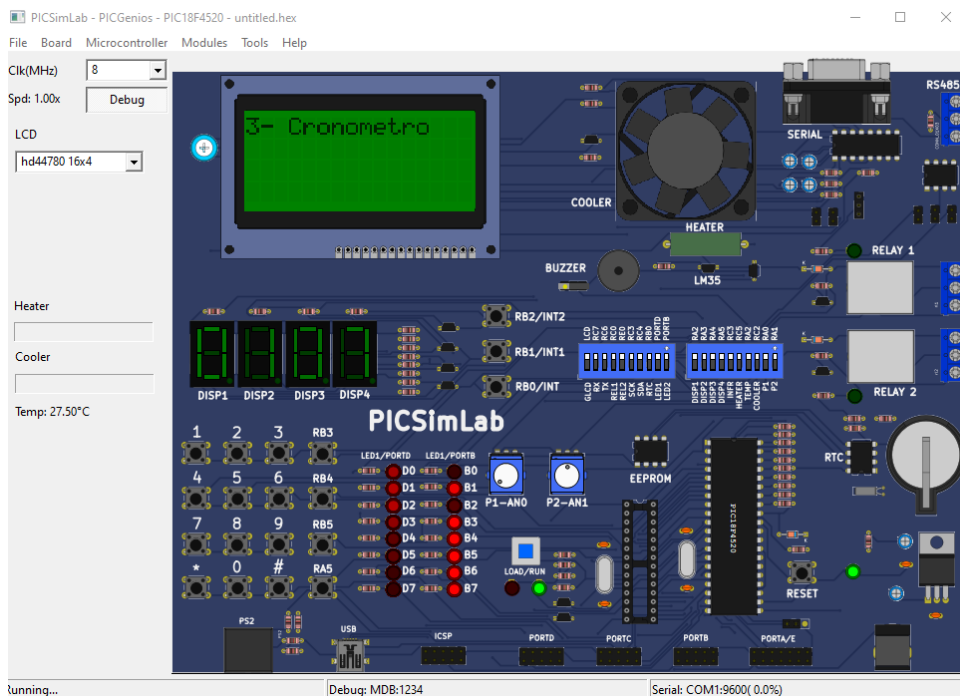
Se o usuário apertar o primeiro botão, aparece uma mensagem e é printado a hora atual no display LCD, caso o usuário aperte de novo o botão, a hora é atualizada.





**Imagem 7 – Funcionamento da tecla 1**

Caso o botão 2 seja apertado, o buzzer é acionado, dando um curto período de tempo para o usuário se prepara, após o sinal sonoro o display de 7 segmentos começa a contar, além disso os leds do PORTB são acionados, indicando que o cronometro está ligado. Caso o usuário aperte o botão 2 novamente, o cronometro é zerado



**Imagem 8 - Funcionamento da tecla 2**

Apertando o botão 3, o usuário terá acesso ao número de passos no painel LCD, o número de passos começa em zero e aumenta a de acordo com que o tempo do cronometro passa.

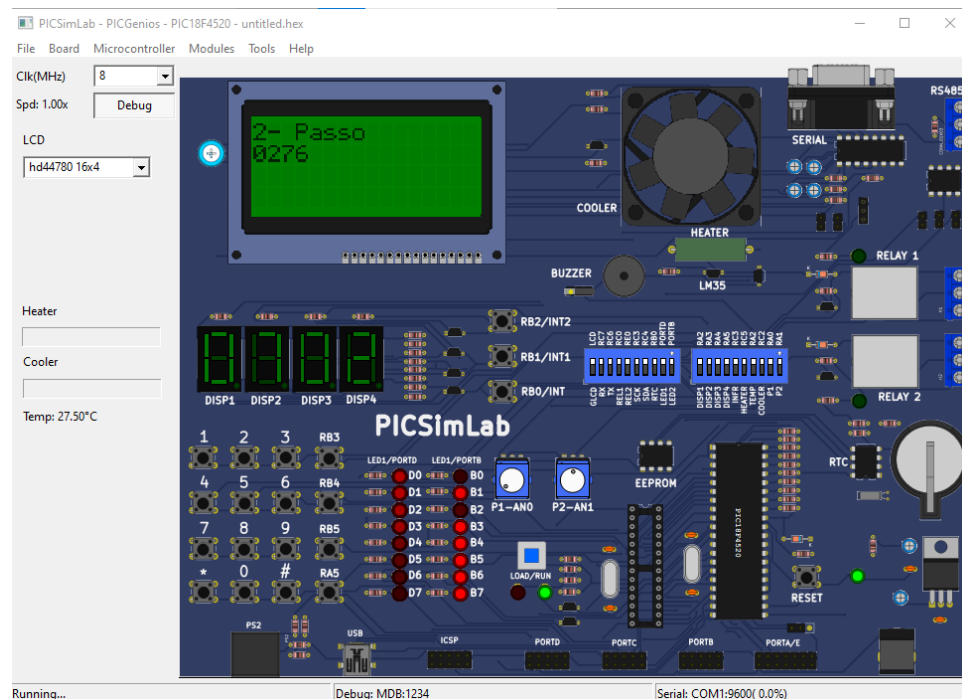


Imagem 9 – Funcionamento da tecla 3

Ao pressionar a tecla 4 o usuário poderá ver o batimento cardíaco por minuto

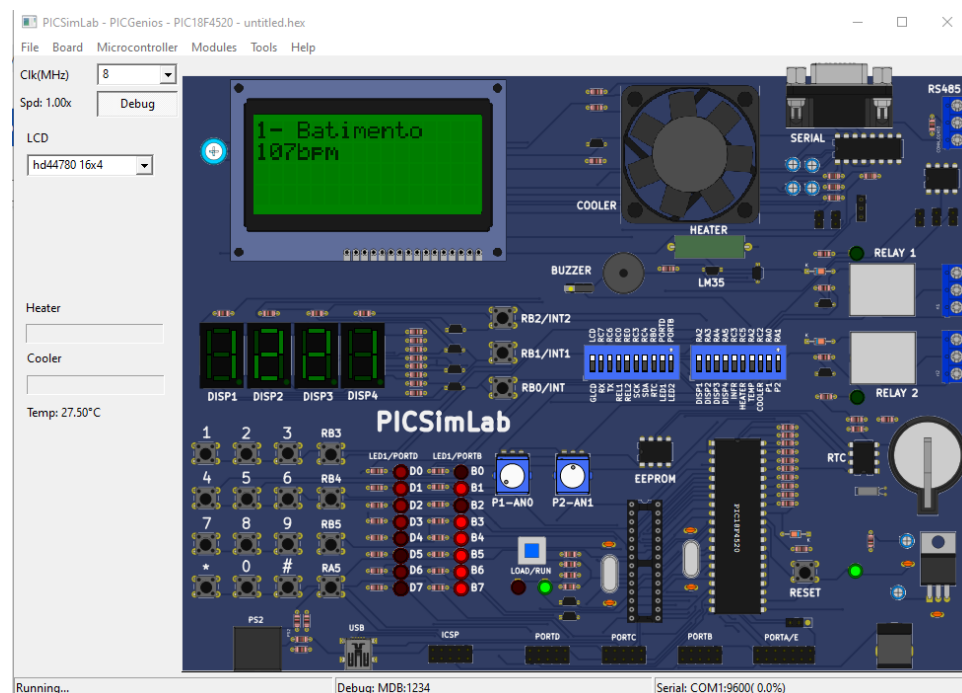


Imagem 10 – tecla 4

Ao pressionar a tecla 5, os leds do PORTB são desligados, indicando que o cronometro parou, além disso o tempo em que ele ficou ativo é mostrado no painel lcd e todos os display, exceto o penúltimo são desativados, mostrando que o tempo parou.

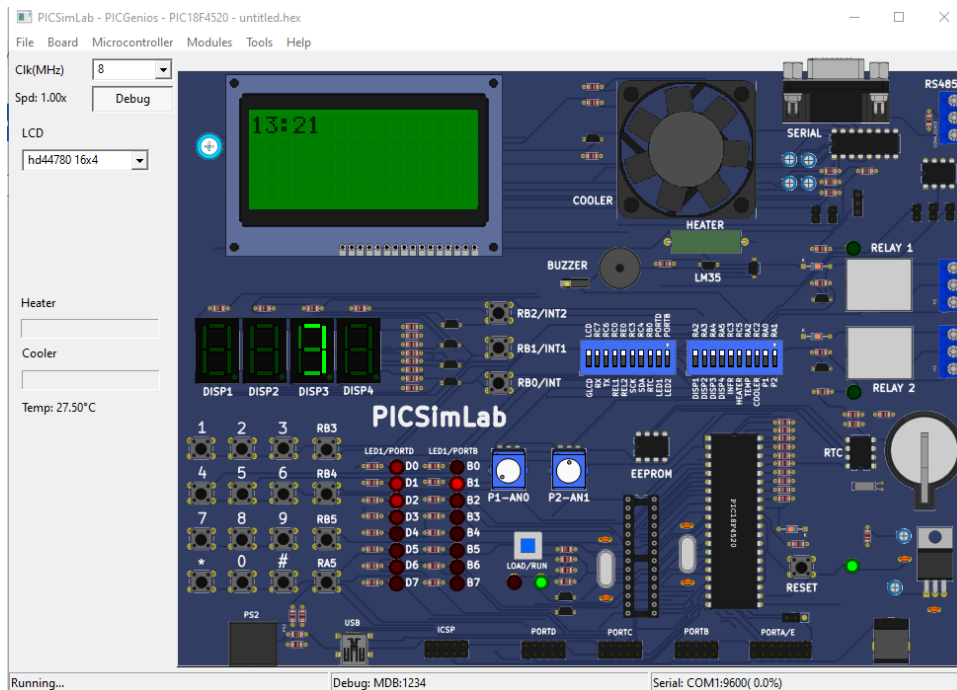


Imagem 11 – tecla 5

## Conclusão

O projeto cumpre o esperado, é usado cinco componentes do PICSIMLab, um dos requisitos desejados pelo professor.