

Trabalho de Programação – Processador CESAR16i 2024/1

1. Descrição Geral

Desenvolver duas funções para serem executadas no Computador CESAR. As funções são as seguintes:

- Relógio, com controles de teclado;
- Leitura de string de teclado, com apresentação de cursos piscante;

Seu programa, com as duas funções e a área de stack, deve estar completamente contido na área de memória entre 0000H e 7FFFH.

- Seu programa deve iniciar no endereço 0000H;
- O fundo da pilha deve ser o endereço 7F00H
- Cada par de endereços a seguir deve receber o endereço de início de uma das funções implementadas:
 - 7F00H e 7F01H, recebe o endereço de início da função de Relógio.
 - 7F02H e 7F03H, recebe o endereço de início da função de Leitura de String;
- Os endereços 7F06H até FF7F devem permanecer sem uso pelo seu programa;
- Os endereços FF80H até FFFFH são os periféricos, que seu programa deve utilizar para implementar as duas funções.

Logo depois que seu programa terminar de realizar todas as atividades necessárias de inicialização, seu programa deve efetuar um JMP para o endereço 8000H.

No endereço 8000H será colocado um programa que vai chamar suas funções. Portanto, suas funções podem ser chamadas várias vezes e em qualquer ordem.

Este programa utilizará uma instrução “JSR R7,endereço” para chamar suas funções. Portanto, ao final da execução de cada função, deve haver um “RTS R7”.

Por exemplo, para chamar as duas funções, uma após a outra, o programa poderá usar o seguinte código:

```
ORG      H8000
MOV      #0, R0                ; Chamada para função de Relógio
JSR      R7, (H7F00(R0))
MOV      #2, R0                ; Chamada para função de String
JSR      R7, (H7F00(R0))
HLT
```

2. Descrição das funções

A seguir estão descritas as duas funções. Todas elas devem ser encerradas por um “RTS R7”. Além disso, antes de encerrar e retornar para o chamador, estas funções devem limpar o visor.

2.1. Relógio, com controles de teclado

Esta função apresenta, no visor, um relógio com horas, minutos e segundos, todos com 2 dígitos. O relógio deve ser apresentado a partir da posição mais a esquerda do visor. O restante do visor deve permanecer “limpo”. **Para implementar o relógio, deve ser usada a interrupção de timer.** O programa principal deve atualizar o visor com as informações do relógio e efetuar a leitura do teclado.

Esta função deve responder a algumas teclas, da seguinte forma:

- “H” ou “h”: incrementa as horas do relógio;
- “M” ou “m”: incrementa os minutos do relógio;
- “Z” ou “z”: zera o relógio;
- CTRL-F: encerra a função e retorna para o chamador.

2.2. Leitura de string de teclado, com apresentação de cursos piscante

Esta função efetua a leitura do teclado e coloca a tecla lida no visor. Durante a espera pelo teclado, a função deve apresentar um cursor piscante no visor. Sempre que uma tecla for digitada, o caractere deve ser colocado no visor, na posição do cursor. Em seguida, o cursor deve ser movido uma posição para a direita.

O cursor inicia na posição mais a esquerda. Ao digitar as teclas, caso o cursor chegue na posição mais a direita do visor, o visor deve ser completamente limpo e o cursor deve ser posicionado na posição mais a esquerda. **Para ler o teclado, a função deve usar a interrupção de teclado.**

O cursor é implementado através da alternância entre o caractere do visor e o caractere “_” (*underscore*). A frequência de alternância entre os dois símbolos deve ser de 4 vezes por segundo (ou seja, cada caractere deve permanecer no visor por 250ms). **Para determinar esta temporização, deve ser usada a interrupção de timer.**

Além de colocar no visor as teclas visíveis, esta função deve responder a algumas teclas de controle, da seguinte forma:

- CR ou CTRL-M: posiciona o cursor na posição mais a esquerda do visor;
- LF ou CTRL-J: limpa o visor (preenche com espaços), mantendo o cursor inalterado;
- BS ou CTRL-H: move o cursor para a esquerda e apaga (coloca espaço) o caractere que estiver nesta nova posição;
- CTRL-F: encerra a função e retorna para o chamador.

3. Correção e Entregas

Para realizar os testes, sua implementação será carregada na memória e, em seguida, carregado o programa de teste, a partir do endereço 8000H. Para isso, será usada a “carga parcial” disponível no simulador do CESAR16i. Depois dessas duas cargas a execução será iniciada a partir do endereço H0000, e será verificado o comportamento correto de suas funções.

Você deverá entregar um arquivo fonte (arquivo .CED) com a sua implementação, escrito em linguagem simbólica do CESAR16i, o qual será montado com o montador Daedalus. Seu programa não pode ter erros de montagem. O código do programa fonte deverá conter comentários descritivos da implementação.

O trabalho deverá ser entregue até a data prevista indicada no sistema Moodle. Não serão aceitos trabalhos entregues após o prazo estabelecido. Trabalhos não entregues até a data prevista receberão nota zero. Trabalhos que tenham erro de montagem receberão nota zero.

4. Observações

Recomenda-se a troca de ideias entre os alunos. Entretanto, a identificação de cópias de trabalhos acarretará na aplicação do Código Disciplinar Discente e a tomada das medidas cabíveis para essa situação (**tanto o trabalho original quanto os copiados receberão nota zero**).

Recomenda-se, também, que sejam usados os programas de exemplo e exercícios disponibilizados durante as aulas.

O professor da disciplina reserva-se o direito, caso necessário, de solicitar uma demonstração do programa, onde o aluno será arguido sobre o trabalho como um todo. Nesse caso, a nota final do trabalho levará em consideração o resultado da demonstração.