

PRIMEIRA PROVA REGIMENTAL - PROJETO
LINGUAGEM DE MONTAGEM

NOTA:

DATA: 16/10/2017

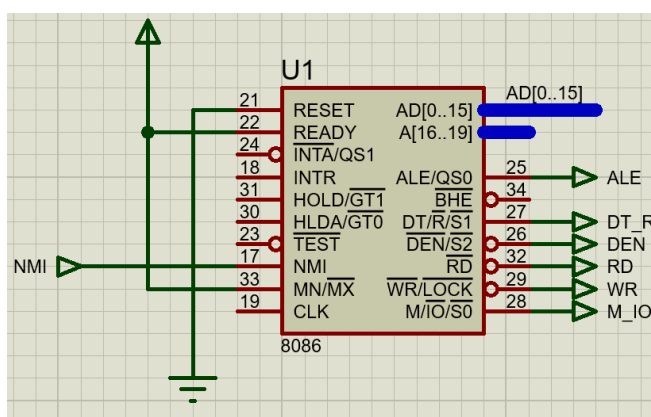
NOME: _____

RA: _____

A organização, legibilidade e limpeza serão consideradas na correção.
Todas as suas respostas devem ser justificadas.

Considere o microprocessador de 16 bits 8086 com estado de operação em modo mínimo:

Figura 1 - Microprocessador 8086



A Figura 2 mostra que o barramento de dados (pinos AD0 a AD7) está conectado a um *buffer* bidirecional 74HCT245 (registrador de 8 bits) que realiza a interface com o periférico programável de entrada e saída de dados 8255A (três portas I/O de 8 bits), que integra ao microcontrolador o display LCD e o teclado matricial, conforme apresenta a Figura 3.

Figura 2 - CI's 74HCT245 (*buffer*) e 8255A (periférico)

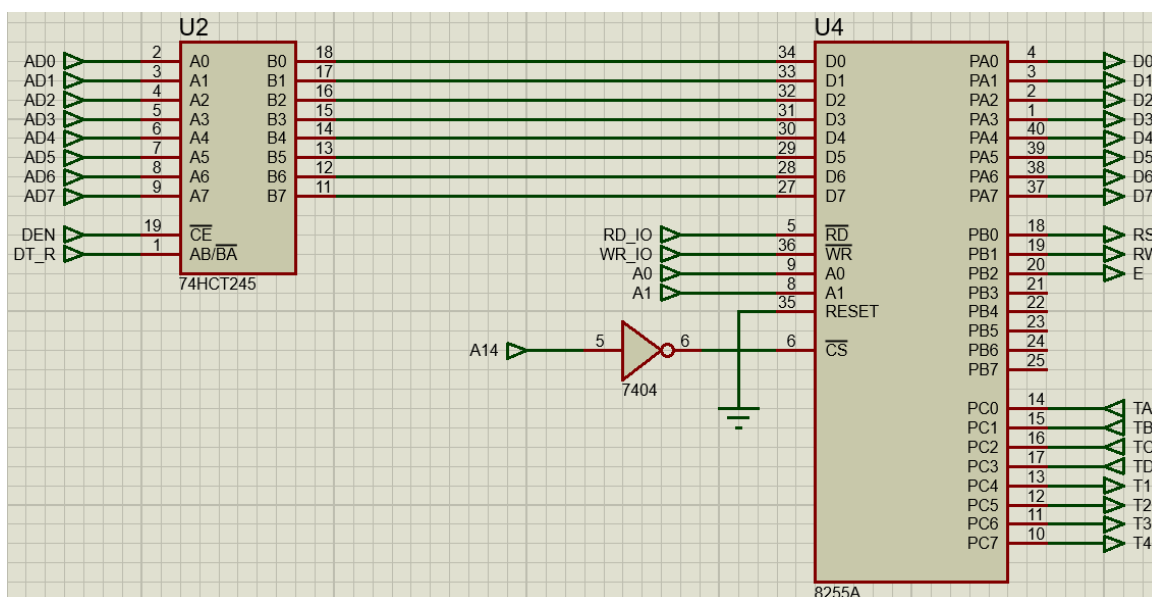
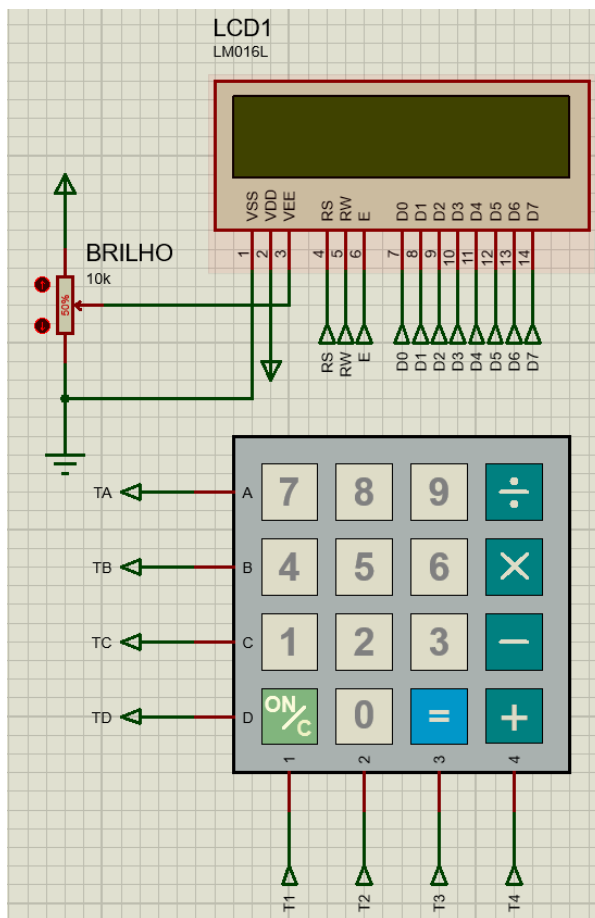
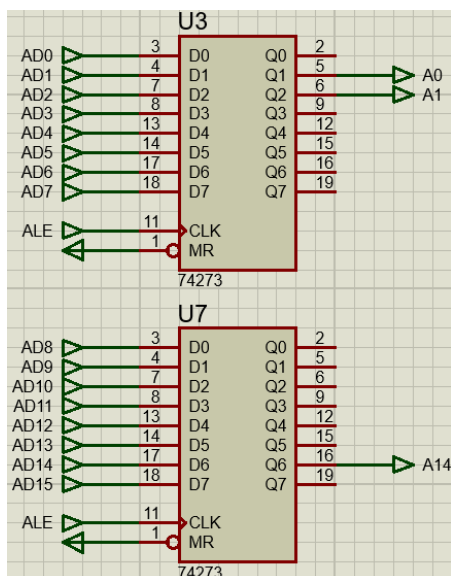


Figura 3 - Display LCD e Teclado Matricial



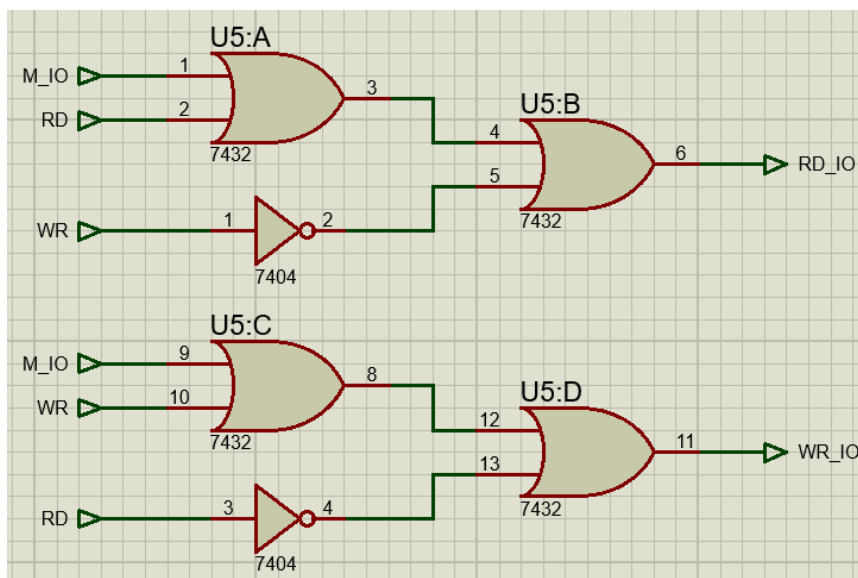
Admita que o barramento de endereços do microcontrolador acione o periférico 8255A por meio dos terminais AD1, AD2 e AD14, que ficam disponíveis ao programador com o auxílio do *flip-flop* tipo *D* 74273 (8 bits) da Figura 4, conectado aos pinos AD0 a AD15 do 8086.

Figura 4 - CI's 74273 (*flip-flop*)



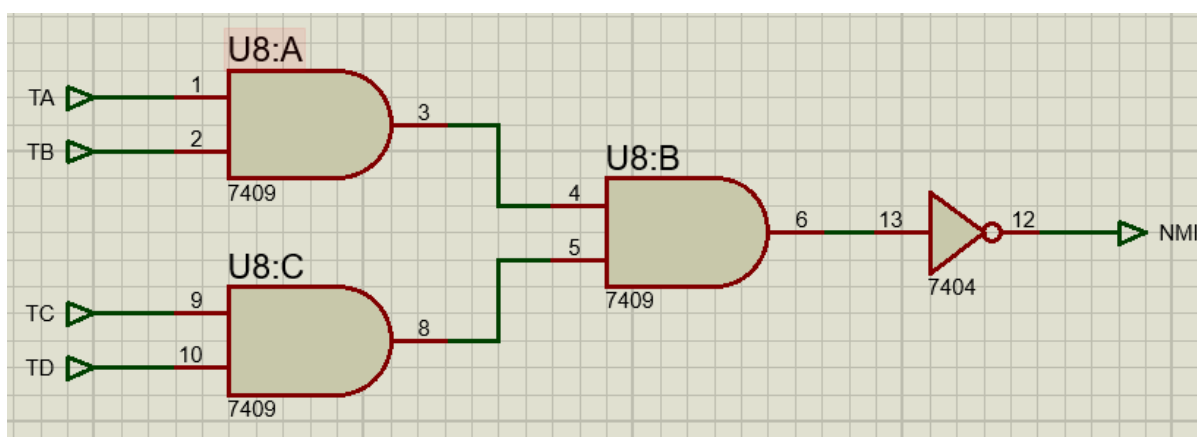
Com o auxílio de algumas portas lógicas é possível garantir a sincronização dos dados para leitura e escrita do periférico 8255A por meio do funcionamento interno do 8086, conforme a mostra a Figura 5:

Figura 5 - Acionamento de I/O do 8255A



Considerando que, se qualquer tecla for pressionada, os estados das linhas TA, TB, TC e TD do teclado matricial serão alterados de acordo com a lógica da varredura das teclas, levando ao acionamento da interrupção NMI (não mascarável, isto é, obrigatória) do microcontrolador 8086, indicando que a matriz teve seu *status* alterado, conforme indica a Figura 6:

Figura 6 - Acionamento da Interrupção do 8086



Além disto, o display LCD utilizado neste circuito é do tipo HD44780U, que consiste em duas linhas de 16 caracteres com modo de operação de 4 ou 8 bits, em que este último foi adotado na transmissão de dados junto ao CI 8255A.

EXERCÍCIO 01 – 0,5 ponto

Elabore uma sub-rotina em linguagem de montagem (*assembly*) que configure as portas I/O do CI 8255A de acordo com a Tabela 1.

Tabela 1 - Configuração do 8255A

Porta	Bits	Operação
A	PA7 a PA0	
B	PB7 a PB0	Escrita
C	PC7 a PC4	
C	PC3 a PC0	Leitura

- Atribua um endereço de memória de 16 bits que faça que o byte de programação seja enviado para o registrador de configurações do 8255A, isto é, que pelo menos contenha na sua palavra os bits $A_{14} = 1$, $A_1 = 1$ e $A_0 = 1$;
- Determine o byte de programação que melhor se adeque ao problema proposto na Tabela 1. Sugestão: procure no *datasheet* do fabricante o valor de comando;
- Utilize a instrução OUT para programar o CI; o registrador AL deve armazenar a palavra de configuração das portas de 8 bits, enquanto que o registrador DX deve armazenar o endereço de 16 bits atribuído no item “a”;
- A sub-rotina deve ser nomeada de INIT_IO;
- Utilize a pilha se necessário;

EXERCÍCIO 02 – 0,5 ponto

Elabore um conjunto sub-rotinas em linguagem de montagem (*assembly*) que realize a leitura e a escrita do periférico 8255A, considerando que o mesmo já esteja configurado de acordo com a Tabela 1, e que tenha como operando de entrada e saída o registrador AH;

- Atribua três endereços de memória de 16 bits que realizem o acesso individual nas portas do CI, em que pelo menos contenha nas suas palavras os bits A_{14} , A_1 e A_0 na condição imposta pela Tabela 2:

Tabela 2 - Acesso as Portas do 8255A

Porta	Bits
A	$A_{14} = 0$ e $A_0 = 0$
B	$A_{14} = 1$ $A_1 = 0$ e $A_0 = 1$
C	$A_1 = 1$ e $A_0 = 0$

- b) De acordo com a Tabela 1, há três sub-rotinas de escrita e uma de leitura que devem ser elaboradas; o dado escrito deve ser passado à sub-rotina por meio do registrador AH e o dado lido deve estar disponível em uma posição de memória interna do 8086 chamada PORTC_VAL;
- c) Utilize a instrução OUT para escrita e IN para leitura; o registrador AL deve armazenar a palavra o dado de saída ou entrada das portas de 8 bits, enquanto que o registrador DX deve armazenar um dos endereços de 16 bits atribuídos no item “a”;
- d) As sub-rotinas devem nomeadas de: OUT_PORTA, OUT_PORTB, OUT_PORTC e IN_PORTC;
- e) Utilize a pilha se necessário;

EXERCÍCIO 03 – 0,5 ponto

Elabore uma sub-rotina de delay que tenha como operando o registrador CX.

- a) Utilize a instrução JZCX e LOOP;
- b) Considere que se CX = 0001h, o delay é de 22 μ s (tempo aproximado do código);
- c) A sub-rotina deve ser nomeada de DELAY;
- d) Utilize a pilha se necessário;

EXERCÍCIO 04 – 0,5 ponto

Elabore uma sub-rotina que envie uma palavra de comando para o display LCD e tenha como operando o registrador AH.

- a) Carregue o caráter de comando no barramento de dados do display, isto é, escreva o dado na PORTA;
- b) Habilite o display LCD no modo de escrita/comando, isto é, faça E = 1 (enable), RW = 0 (escrita) e RS = 0 (operação de comando);
- c) Carregue o caráter de controle no barramento de configurações do display, isto é, escreva o dado no PORTB;
- d) Aguarde 500 μ s (delay);
- e) Desabilite o display LCD do modo de escrita/comando, isto é, faça E = 0 (enable), RW = 0 (escrita) e RS = 0 (operação de comando);
- f) Escreva o novo dado no PORTB;
- g) A sub-rotina deve ser nomeada de COMANDO_LCD;
- h) Utilize a pilha se necessário;

EXERCÍCIO 05 – 0,5 ponto

Elabore uma sub-rotina que envie um caractere para ser impresso no display LCD e tenha como operando o registrador AH.

- a) Carregue o caráter no barramento de dados do display, isto é, escreva o dado na PORTA;
- b) Habilite o display LCD no modo de escrita/caractere, isto é, faça $E = 1$ (enable), $RW = 0$ (escrita) e $RS = 1$ (envio de caractere);
- c) Carregue o caráter de controle no barramento de configurações do display, isto é, escreva o dado no PORTB;
- d) Aguarde $500\ \mu s$ (delay);
- e) Desabilite o display LCD do modo de escrita/caractere, isto é, faça $E = 0$ (enable), $RW = 0$ (escrita) e $RS = 1$ (envio de caractere);
- f) Escreva o novo dado no PORTB;
- g) A sub-rotina deve ser nomeada de ESCREVE_LCD;
- h) Utilize a pilha se necessário;

EXERCÍCIO 06 – 0,5 ponto

Elabore uma sub-rotina que inicialize o display LCD e deixe-o pronto para uso.

- a) Limpe o barramento de controle do display LCD, isto é, faça $E = 0$, $RW = 0$ e $RS = 0$;
- b) Carregue o caráter de controle no barramento de configurações do display, isto é, escreva o dado no PORTB;
- c) Aguarde 20 ms (delay);
- d) Envie o comando 38h ao display (configura modo de operação);
- e) Envie o comando 0Ch ao display (configura o cursor);
- f) Envie o comando 06h ao display (configura o deslocamento do cursor para direita);
- g) Envie o comando 01h ao display (limpa a tela);
- h) Aguarde 5 ms (delay);
- i) A sub-rotina deve ser nomeada de INIT_LCD;
- j) Utilize a pilha se necessário;

EXERCÍCIO 07 – 0,5 ponto

Elabore uma sub-rotina que limpe a tela do display LCD e deixe-o pronto para uso.

- a) Envie o comando 01h ao display (limpa a tela);

- b) Aguarde 5 ms (delay);
- c) A sub-rotina deve ser nomeada de CLEAR_LCD;
- d) Utilize a pilha se necessário;

EXERCÍCIO 08 – 0,5 ponto

Elabore uma sub-rotina que escreva um conjunto de caracteres (*string*) no display LCD.

- a) Utilize o símbolo “\$” como terminador da *string*;
- b) Utilize o registrador SI como operando de entrada por meio da instrução LEA;
- c) Utilize as instruções CLD e LODSB para criar o loop de impressão de caracteres;
- d) A sub-rotina deve ser nomeada de TXT_LCD;
- e) Utilize a pilha se necessário;

EXERCÍCIO 09 – 0,5 ponto

Elabore uma sub-rotina que permita posicionar o cursor do display LCD em uma posição qualquer da sua matriz de 16 linhas por duas colunas.

- a) Utilize o registrador BX como operando de entrada, isto é, defina BH como linha (1 a 2) e BL como coluna (1 a 16);
- b) Se o BH = 1, carregue o registrador AH com comando 80h (primeira linha);
- c) Se o BH = 2, carregue o registrador AH com comando C0h (segunda linha);
- d) Após o valor de AH ser inicializado, some o número de colunas no mesmo e decamente uma unidade, pois o display entende somente valores de colunas de 0 a 15;
- e) Envie um comando para o display LCD com valor armazenado final de AH;
- f) A sub-rotina deve ser nomeada de CURSOR_POS_LCD;
- g) Utilize a pilha se necessário;

EXERCÍCIO 10 – 0,5 ponto

Complete o código do programa principal com um loop infinito que permita realizar a varredura das colunas do teclado matricial; os valores são ativos em zero.

- a) Envie o comando E0h (desliga barramento T1) para a PORTC;
- b) Aguarde 20 ms (delay);
- c) Envie o comando D0h (desliga barramento T2) para a PORTC;
- d) Aguarde 20 ms (delay);
- e) Envie o comando B0h (desliga barramento T3) para a PORTC;

- f) Aguarde 20 ms (delay);
- g) Envie o comando 70h (desliga barramento T4) para a PORTC;
- h) Aguarde 20 ms (delay);
- i) Utilize a pilha se necessário;

EXERCÍCIO 11 – 0,5 ponto

Toda vez que uma tecla for pressionada, o pino NMI do 8086 sofrerá uma mudança de estado lógico que acionará uma interrupção interna que irá desviar o código do programa principal para uma sub-rotina. Por padrão, a interrupção realiza a leitura de dois endereços de 16 bits para definir o local de salto: o primeiro é o valor do registrador de instruções IP, armazenado em 0008h, e o segundo o quanto o mesmo é deslocado no segmento de códigos, armazenado em 000Ah.

Considere que o endereço de salto armazenado em 0008h seja 4000h e que não haja deslocamento, isto é, que 000Ah armazene 0000h. Complete o código da sub-rotina de interrupção para realizar a leitura da tecla pressionada:

- a) Realize a leitura da PORTC (assume-se que o exercício “10” esteja operando);
- b) Crie uma tabela de decodificação chamada KEY_INT que irá armazenar todos os valores possíveis que podem ser lidos pela PORTC e atribuir o código ASCII de qual tecla foi pressionada;

Tabela 3 – Tabela de Decodificação do Teclado

PORTC	Código ASCII	Tecla Equivalente
E7h	20h	Espaço
B7h	3Dh	=
7Eh	2Fh	/
7Dh	2Ah	*
7Bh	2Dh	-
D7h	30h	0
EBh	31h	1
DBh	32h	2
BBh	33h	3
EDh	34h	4
DDh	35h	5
BDh	36h	6
EEh	37h	7
DEh	38h	8
BEh	39h	9
77h	2Bh	+

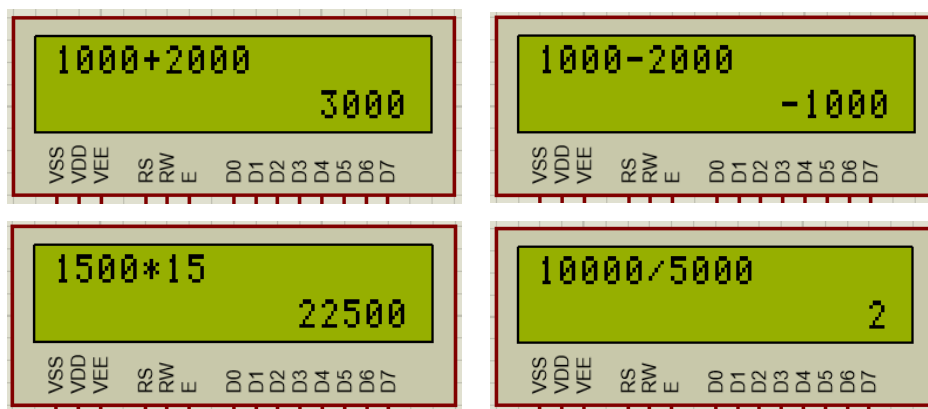
- c) Invoque uma sub-rotina chamada TECLADO dentro da interrupção que irá decodificar o estado dos bits lidos e definir qual caractere foi pressionado;
- d) Utilize o registrador SI e a instrução LEA para realizar a varredura da tabela;
- e) Utilize as instruções CLD e LODSB para criar o loop de impressão de caracteres;
- f) Salve a tecla lida em uma posição de memória chamada TECLA;
- g) Utilize a pilha se necessário;

EXERCÍCIO 12 – 4,5 pontos

Complete o código da sub-rotina de interrupção para emular uma calculadora de 16 bits com quatro operações básicas, em que receba apenas dois operandos positivos, mas que possa retornar valores negativos. A mesma deve:

- a) Realizar as operações básicas de soma, subtração, multiplicação e divisão, e imprimir os resultados, sem zeros à esquerda, conforme mostra a Figura 7;

Figura 7 - Operações Básicas da Calculadora



- b) Ter apenas dois operandos de entrada que variem no intervalo de 0 a 65535. Caso algum deles seja maior que o máximo ou o resultado final seja maior que o mesmo, retornar, após a tecla “igual” for pressionada, a mensagem “#INF!” no canto inferior direito do display, conforme indica a Figura 8. Se o segundo operando for zero, a mensagem a ser exibida é “#DIV/0”;
- c) Se apenas o primeiro operando for digitado e a tecla “igual” for pressionada, não havendo a operação de matemática envolvida, exibir apenas valor do mesmo de acordo com o exemplo da Figura 9;
- d) Caso haja um operador matemático e a tecla “igual” for pressionada, assumir que o segundo operando é zero, seguindo a coerência do exemplo da Figura 10;

e) Caso a tecla ON/OFF seja pressionada, na primeira vez liga o cursor do display LCD (basta enviar o comando 0Fh), mas na segunda desliga (basta enviar o comando 0Ch); o processo sempre fica alternando entre ligar/desligar o cursor quando a mesma tecla é acionada, conforme indica a Figura 11;

Figura 8 - Tratamento de *Overflow* e Divisão por Zero

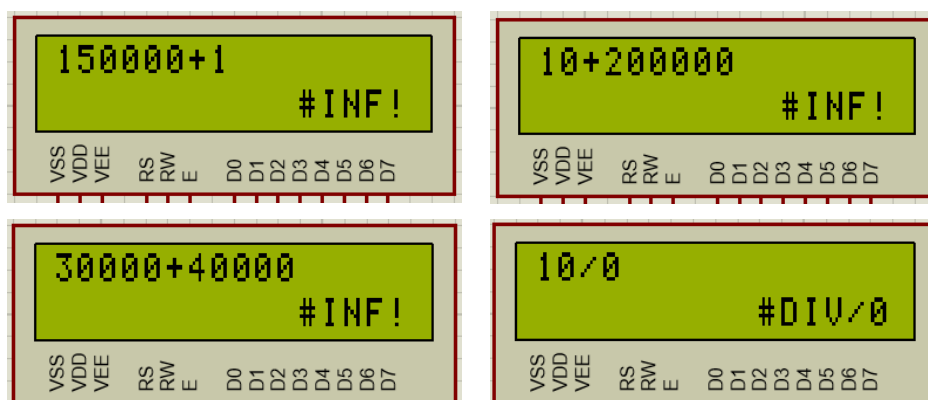


Figura 9 - Tratamento para Falta de Operador Matemático

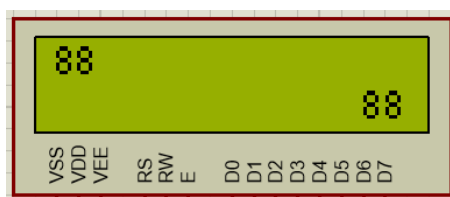


Figura 10 - Segundo Operando Vazio

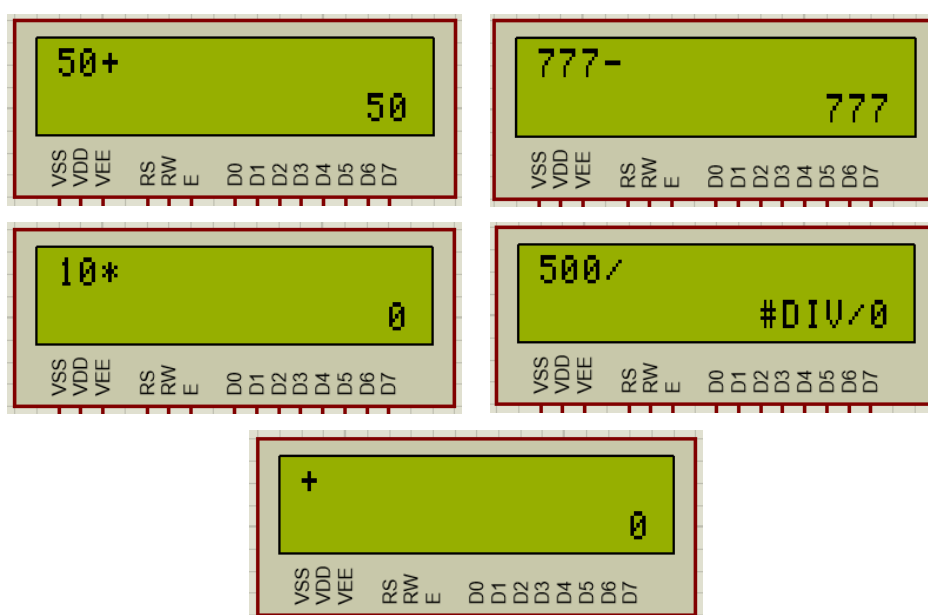
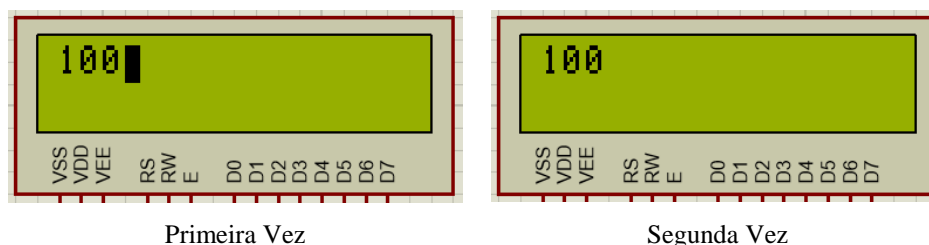


Figura 11 - Cursor do Display LCD



f) Depois que a tecla “igual” for pressionada e os resultados e avisos de erros forem lançadas na tela do display, qualquer evento que ocorra no teclado, exceto o ON/OFF do cursor, devem apagar o mesmo e reiniciar a calculadora automaticamente;