

Atividade Semanal 5

Marcelo Eduardo Reginato - 13676965

Rhayna Christiani Vasconcelos Marques Casado - 13676429

SEL0614 - Aplicação de Microprocessadores

September 24, 2024

Questão 10

Responder com Verdadeiro (V) ou Falso (F) às seguintes afirmações.

(V) A pilha é uma memória RAM sequencial do tipo FIFO.

(V) A pilha geralmente é utilizada para armazenar endereço de retorno de subrotinas e também de interrupções.

(F) O ponteiro de pilha (Stack Pointer) é um registrador que aponta para um endereço da memória ROM, que é o endereço de retorno do programa após o atendimento a uma interrupção ou sub-rotina.

(V) As instruções PUSH e POP são exclusivas para operações com pilha.

(V) A instrução CALL [endereço] deve ser usada para indicar qual endereço o programa deve desviar no caso de um atendimento à interrupção ou chamada de sub-rotina.

(V) A instrução RET, colocada no final de uma sub-rotina, faz com que o último endereço armazenado na pilha seja carregado no registrador PC (program counter).

(F) A área da RAM interna dedicada à pilha é determinada pelo ponteiro SP, um dos SFRs, que possui tamanho 8 bits, mesmo tamanho do barramento de endereço da CPU.

(V) Geralmente são baseadas em flip-flops tipo D.

Questão 11

Refletir se existe diferença entre o endereço armazenado em um espaço de pilha e o endereço armazenado no Stack Pointer (SP)?

Existe, a pilha armazena os dados que estão sendo alocados ou realocados durante a execução, já o StackPointer aponta para o próximo dado que será alocado ou realocado.

Questão 12

Supondo que um banco de 8 LEDs foi conectado à Porta P1 e um banco de 8 Switches conectado à P2 (EdSim51). Acender o LED 0 (pode ser qualquer outro) ao acionar o Switch 7 (pode ser qualquer outro). Apagar o LED ao desligar o Switch.

Foram utilizados o bit da porta P2.0 como Switch 0 e o bit da porta P2.1 como LED 0.

```
1  ORG 0000h    ; origem em 0000h
2  JMP main     ; salta para a main
3
```

```

4 main:
5
6     JB P2.0, $ ; verifica se P2.0 foi zerado, caso nao esteja trava o
        codigo aqui
7     CPL P1.0   ; complementa P1.0
8     JNB P2.0, $ ; verifica se P2.0 foi setada, caso nao esteja trava o
        codigo aqui
9     CPL P1.0   ; complementa P1.0
10
11    JMP main    ; deixa o codigo em loop
12    END        ; finalizacao do codigo

```

Questão 13

Supondo que foram conectados um banco de 8 switches na Porta P2 (EdSim51). Escrever um programa em Assembly para monitorar, quando uma das chaves for pressionada, e indicar o número da chave pressionada em R0 (Se SW3 foi pressionada, então $R0 = 3$).

```

1     org 0000h          ; define origem no endereco 00h
2
3 waiting:
4     MOV A, #0FFh       ; move para A o valor 0xFF (todos os bits 1)
5     CLR C              ; limpa o Carry
6     SUBB A, P2         ; subtrai o valor de P2 em A
7     JZ waiting        ; se o valor de A for 0 (nao houveram chaves
        pressionadas em P2) segue em loop de varificacao
8
9     MOV R0, #00h       ; move o valor 0x00 para R0
10    MOV A, P2          ; move o valor de P2 para o acumulador
11
12 switch_number:
13    RRC A              ; rotaciona atraves do carry o valor de A para a
        direita, verificando qual a chave esta pressionada
14    JNC done           ; se o carry for 0 terminou o codigo
15    INC R0             ; se o carry nao for zero incrementa o contador R0
16    JMP switch_number  ; continua em loop ate o carry ser 0
17
18 done:
19    END                ; finalizacao do codigo

```

Questão 14

Criar uma subrotina de delay de 50 milissegundos a partir da contagem de ciclos de instruções e intervalo de tempo. Essa estrutura poderá servir para piscar um LED a cada 50 ms (ver exemplo na aula correspondente).

```

1     org 0000h          ; origem em 0000h
2
3     ACALL delay        ; chama subrotina de delay
4
5 delay:
6     MOV R1, #100       ; mov 100 para o registrador R1
7

```

```

8 inicio:
9     MOV R0, #250      ; mov 250 para o registrados R0
10    DJNZ R0, $         ; decrementa R0 ate seu valor ser igual a 0
11    DJNZ R1, inicio   ; decrementa R1 e faz todo esse ciclo anterior
                        ; novamente ate R1 chegar em 0
12    RET
13    END               ;finalizacao do codigo

```

Questão 15

Colocou-se 3 LEDs nos endereços P1.0, P1.1 e P1.2 no microcontrolador e 3 chaves nos endereços P2.0, P2.1 e P2.2. Considerando que os LEDs acendem quando é colocado nível baixo na saída e as chaves, quando pressionadas, colocam nível baixo na porta, explique o funcionamento do programa a seguir quando cada chave é pressionada.

```

1     ORG 0000H
2
3     Leitura:
4         JNB P2.0, PX
5         JNB P2.1, PY
6         JNB P2.2, PZ
7         LCALL Leitura
8
9     PX:
10        MOV P1, #0
11        RET
12
13    PY:
14        MOV P1, #00000101b
15        RET
16
17    PZ:
18        MOV A, P1
19        CPL A
20        MOV P1, A
21        RET
22
23    FIM:
24        SJMP FIM

```

O código fica em loop fazendo a verificação de P2.0, P2.1 e P2.2. Quando P2.0 é pressionada todos os LEDs conectados acendem, pois foi movido #0 para a porta P1 inteira, quando P2.1 é pressionada move #00000101b para a porta P1, de forma que acendem os LEDs P1.0 e P1.1 e por fim quando P2.2 é pressionada move-se o conteúdo de P1 para o acumulador, complementa o acumulador e retorna esse resultado para P1. No mais o código fica em loop por conta da subrotina chamada em LCALL Leitura.