

## LISTA 10 - Resolução

### Questão 1)

A arquitetura é Von Neumann, pois o barramento de dados (RAM) é o mesmo do de endereços (ROM Flash).

### Questão 2)

*Ciclo de busca:* Ciclo de busca de um Opcode de uma instrução a partir da posição de memória definida pelo Program Counter (PC). Esse Opcode é então armazenado no Instruction Register (IR) para ser executado.

*Ciclo de execução:* Decodifica a instrução e a executa a instrução (caso ela ocupe apenas uma posição). Ou busca os demais bytes da instrução na memória do programa, iniciando um novo ciclo de busca. O conteúdo do PC é incrementado uma, duas ou três unidades dependendo do tamanho da instrução.

*Ciclo de máquina:* Ciclo de busca + ciclo de execução.

*Ciclo de instrução:* O tempo necessário para a execução de uma instrução completa. Pode ser composto por vários ciclos de máquina como nas instruções "INC DPTR" (2 ciclos) ou "MUL A, B" (4 ciclos).

### Questão 3)

*Ordem das respostas:* V F V F F F V V

*Correções (na ordem):*

- A instrução SETB deixa o bit em nível alto, e não baixo.
- Algumas vezes possuem. O registrador PSW possui nos bits 3 e 4, RS0 e RS1. Esses dois bits decidem qual banco de registradores de uso geral será usado para R0 a R7.
- Os flags são Special Function Registers e estão entre 80h e FFh.
- 20h a 2Fh são GPR que podem ser endereçados por bit (não SFR). 80h é um SFR (o P0) que pode ser endereçado por bit. CFh não é um SFR.
- CISC é comum para arquitetura Von Neumann.

### Questão 4)

- a) 20h
- b) 30h
- c) 95h

### Questão 5)

ORG 0000h  
CLR A  
ADD A, #34h  
MOV R0, #dado

```

ADD A, R0
MOV 28h, A
END

```

### Questão 6)

*Program Counter:*

Avanço em bytes de cada instrução:  $1 + 2 + 2 + 1 + 2 = 8$

PC: 0008h

*Tempo de execução:*

Ciclos de máquina de cada instrução:  $1 + 1 + 1 + 1 + 1 = 5$

Tempo:  $5 \times 1\mu = 5\mu$

### Questão 7)

Bytes	T	Programa	Descrição do programa	Valores recebidos				
				PC	SP	R0	A	23h
		ORG 0	Define que as instruções serão escritas em 0000h.					
2	12	MOV A, #73h	Inserir 73h no acumulador.	0000h	07h	00h	00h	00h
2	12	ADD A, #19h	Soma 19h no acumulador.	0002h	07h	00h	73h	00h
2	12	MOV R0, #23h	Inserir 23h no registrador R0.	0004h	07h	00h	8Ch	00h
3	24	LCALL Rotina	Chama a sub-rotina "Rotina".	0006h	07h	23h	8Ch	00h
2	24	MOV R0, 23h	Inserir o conteúdo do end. 23h no registrador R0.	0009h	07h	23h	8Ch	8Ch
		Loop:						
2	24	SJMP Loop	Pula o programa para "Loop".	000Bh	07h	8Ch	8Ch	8Ch
		Rotina:						
		ORG 41h	Define que as instruções serão escritas em 0041h.					
1	12	MOV @R0, A	Inserir o valor do acumulador no end. apontado por R0	0041h	08h	23h	8Ch	00h
1	24	RET	Retorna da sub-rotina.	0041h	08h	23h	8Ch	8Ch

### Questão 8)

O programa joga 4Dh no registrador R2 e limpa o acumulador. Em seguida, entra em um ciclo, jogando o valor do endereço indicado em R2 no acumulador (inicialmente o end. 4Dh), incrementando R2 e verificando após isso se R2 é igual a 51h, caso em que o ciclo se encerra. O acumulador é igual a soma dos valores contidos nos endereços 4Dh, 4Eh, 4Fh e 50h. Pela tabela:

$$A = 48h + 8h + 11h + 0Fh = 70h$$

R2 = 51h ao fim da operação (mas o valor contido em 51h não é somado).

### Questão 9)

O programa irá controlar o comportamento dos LEDs conforme o pressionamento das chaves.

Chave 2.0 pressionada: Todos os LEDs acendem.

Chave 2.1 pressionada: Somente o LED 1.1 acende.

Chave 2.2 pressionada: Os LEDs acesos se apagam, e os apagados acendem.

**Questão 10)**

Diretivas são mnemônicos que indicam informações no programa, como endereço inicial, reserva da área de dados, definir equivalência entre identificadores e valores. Contudo, não geram código de máquina, portanto, não alteram o tempo de execução do programa.

Caso seja feita essa mudança na questão 5, a única alteração será no registrador PC que inicia em 0010h e seu valor final é, portanto, acrescido em 0010h.

**Questão 11)**

O programa insere valores em ordem crescente em endereços em ordem decrescente. Começa inserindo 0 em 0Ch, 1 em 0Bh, 2 em 0Ah, 3 em 09h e assim por diante, até inserir 0Bh no endereço 01h, quando o registrador decremente, vai para 00h e sai do loop "prog".