

**DISCIPLINA: LABORATÓRIO DE OAC – LOAC**

# **RELATÓRIO: LABORATÓRIO 01**

**2021**

**ALUNO: Ludmila Maria Pereira da Silva**

**Prof<sup>(a)</sup>. Dr<sup>(a)</sup>. Misael de Moraes**

[illegible]



Na linha 10, acontece uma troca entre o acumulador A e o conteúdo da memória em 2A.

x0	x1	x2	x3	x4	x5	x6	x7
00	00	2A	02	00	00	00	00
08	00	00	00	00	00	00	00
10	00	00	00	00	00	00	00
18	00	00	00	00	00	00	00
20	2A	00	00	00	00	00	00
28	02	00	15	00	00	00	00
30	00	00	00	00	00	00	00
38	00	00	00	00	00	00	00
40	00	00	00	00	00	00	00

HEX	DEC	BIN	OCT	CHAR
A: 33	51	00110011	63	3
B: 00	0	00000000	0	

PSW: C AC F0 RS1 RS0 OV - P

R7	R6	R5	R4	R3	R2	R1	R0
00	00	00	00	00	02	2A	00

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	tQz...	(y*uQ.w3..														
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	*..	....	(....7./													
0020	63																c															
0030																																

Logo após, na linha 11, o conteúdo da memória em 01h é movido para a memória 32.

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	tQz...	(y*uQ.w3..														
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	*..	...	(...	7./												
0020	63																c															
0030																																

Na linha 12 o #TAB é movido para o registrador DPTR.

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	tQz...	(y*uQ.w3..														
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	*..	...	(	...	7.	/										
0020	63																c															
0030																																

Na linha 14, o valor em A é somado ao DPTR, obtendo um endereço de 16 bits na memória:

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	tQz...	(y*uQ.w3..														
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	*	..	....	(	...	7.	/									
0020	63																c															
0030																																

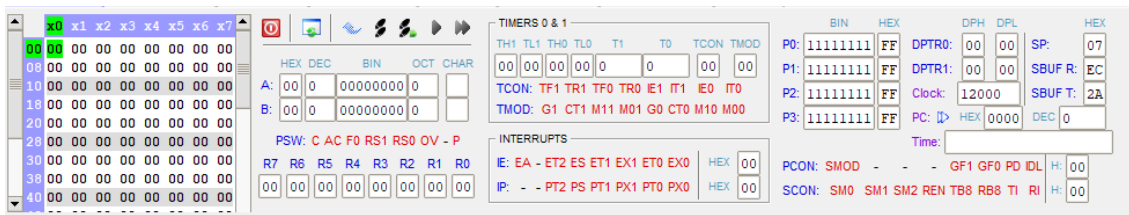
Finalizadas as análises dos Registradores deste programa, segue-se para a próxima etapa do experimento, que pede explicação sobre cada sentença contida na listagem. Em paralelo, parte-se também para a fase c, que pede uma análise mais esmiuçada sobre cada linha de comando.

```

1 ORG      0000H                      ;endereço inicial do programa
2 INICIO:                      ;rótulo da operação
3          MOV      A, #051h          ;A constante é movida para o acumulador . Modo de endereçamento
; imediato. Mnemonica: MOV. Operando: A,#051h.
4          MOV      R2, #02h          ;O valor #02h é movido para o registrador R2. Modo de
; endereçamento registrador e imediato. Mnemonica: MOV. Operando: R2,#02h
5          MOV      28h, 02h          ;O conteúdo da memória em 02h é movido para memória de 28h. Modo
; de endereçamento direto. Mnemonica: MOV. Operando: 28h, 02h
6          MOV      R1, #42           ;O valor 42 é movido para o registrador R1. Modo de endereçamento
; registrador e imediato. Mnemonica: MOV Operando: R1,#42.
7          MOV      51h, #12          ;O valor 12 é movido para a memória 51h. Modo de endereçamento
; direto e imediato. Mnemonica: MOV. Operando: 51h, #12
8          MOV      @R1, #33h         ;O numero 33h é movido para a memória ram endereçada no R1. Modo
; de endereçamento imediato e indireto. Mnemonica: MOV. Operando @R1, #33h.
9          SWAP      A                ; Troca os nibbles do acumulador. Mnemonica: SWAP. Operando: A.
10         XCH       A, 2Ah           ;O conteúdo da memória de 2Ah é trocado com acumulador. Modo de
; endereçamento direto. Mnemonica: XCH. Operando: A,2Ah.
11         MOV      32,01h            ;O conteúdo da memória 01h é enviado para a memória de 32. Modo
; de endereçamento direto. Mnemonica: MOV. Operando: 32,01h.
12         MOV      DPTR, #TAB        ;O valor de TAB é movido para o Registrador DPTR. Modo de
; endereçamento imediato. Mnemonica: MOV. Operando: DPTR,#TAB
13         MOV      A,28h             ;O valor 28h é movido para o Acumulador. Modo de endereçamento
; direto. Mnemonica MOV: Operando: A,28h.
14         MOVC     A,@A+DPTR         ;o valor A é somado com o registrador DPTR e movido para o
; acumulador. Modo de endereçamento: Indexado. Mnemonica: MOVC. Operando: A,@A+DPTR.
15         LJMP     INICIO            ; Retorna a posição de memória armazenada no rótulo do INICIO.
; Mnemonica: LJMP. Operando: INICIO
16 TAB:      DB       37h, 12, 47, 99 ; posiciona 4 bytes de forma ordenada a partir da posição atual.
; Mnemonica: TAB. Operando: DB 7h,12,47,99.
17 END                                ;informa ao compilador o final da execução

```

Antes de compilar o código, esta é a interface do programa:



Assim que o programa se inicializa, no endereço de memória 0000H, a memória de dados e do programa antes de avançar para a primeira instrução mnemônica:

	x0	x1	x2	x3	x4	x5	x6	x7										
00	00	00	00	00	00	00	00	00										
08	00	00	00	00	00	00	00	00										
10	00	00	00	00	00	00	00	00										
18	00	00	00	00	00	00	00	00										
20	00	00	00	00	00	00	00	00										
28	00	00	00	00	00	00	00	00										
30	00	00	00	00	00	00	00	00										
38	00	00	00	00	00	00	00	00										
40	00	00	00	00	00	00	00	00										

Logo após, na linha 3, a primeira instrução mnemônica será executada. A constante hexadecimal é movida para o acumulador A:

	x0	x1	x2	x3	x4	x5	x6	x7																										
00	00	00	00	00	00	00	00	00											0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
08	00	00	00	00	00	00	00	00																										
10	00	00	00	00	00	00	00	00																										
18	00	00	00	00	00	00	00	00																										
20	00	00	00	00	00	00	00	00																										
28	00	00	00	00	00	00	00	00																										
30	00	00	00	00	00	00	00	00																										
38	00	00	00	00	00	00	00	00																										
40	00	00	00	00	00	00	00	00																										
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	tQz	.	.	{y*uQ.w3..														
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	*	.	.	.	.	(	.	.	.	.	7	.	/					
0020	63																																	
0030																																		

A seguir, na instrução contida na linha 4, move o valor de 02h para o registrador R2:

Na próxima linha, a 5, o conteúdo de memória em 02h é movido para a memória de 28h:

Na linha 6, O valor 42 é movido para o Registrador R1:

Na linha 7, o valor 12 é movido para a memória do 51h:

Na linha 8, o número 33h é movido para a RAM endereçada no R1:

Linha 9, a mnemônica SWAP troca os nibbles do acumulador:

	x0	x1	x2	x3	x4	x5	x6	x7	
00	00	2A	02	00	00	00	00	00	
08	00	00	00	00	00	00	00	00	
10	00	00	00	00	00	00	00	00	
18	00	00	00	00	00	00	00	00	
20	00	00	00	00	00	00	00	00	
28	02	00	33	00	00	00	00	00	
30	00	00	00	00	00	00	00	00	
38	00	00	00	00	00	00	00	00	
40	00	00	00	00	00	00	00	00	

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF		
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	0123456789ABCDEF
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	tQz... (y*uQ.w3..
0020	63																*... (...7./
0030																	c

Na linha 10, o conteúdo da memória de 2Ah é trocado com o acumulador:

	x0	x1	x2	x3	x4	x5	x6	x7	
00	00	2A	02	00	00	00	00	00	
08	00	00	00	00	00	00	00	00	
10	00	00	00	00	00	00	00	00	
18	00	00	00	00	00	00	00	00	
20	00	00	00	00	00	00	00	00	
28	02	00	15	00	00	00	00	00	
30	00	00	00	00	00	00	00	00	
38	00	00	00	00	00	00	00	00	
40	00	00	00	00	00	00	00	00	
48	--	--	--	--	--	--	--	--	

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF	0123456789ABCDEF	
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	Qz... (y*uQ.w3..
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	*... (...7./
0020	63																c
0030																	

Na linha 11, novamente com o MOV, o conteúdo da memória 01h é enviado para a memória de 32:

	x0	x1	x2	x3	x4	x5	x6	x7	
00	00	2A	02	00	00	00	00	00	
08	00	00	00	00	00	00	00	00	
10	00	00	00	00	00	00	00	00	
18	00	00	00	00	00	00	00	00	
20	2A	00	00	00	00	00	00	00	
28	02	00	15	00	00	00	00	00	
30	00	00	00	00	00	00	00	00	
38	00	00	00	00	00	00	00	00	
40	00	00	00	00	00	00	00	00	
	--	--	--	--	--	--	--	--	

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF		
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	0123456789ABCDEF
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	EQz... (y*uQ.w3..
0020	63																*... (...7./
0030																	c

Na linha 12 o valor de TAB é movido para o registrador DPTR:

	x0	x1	x2	x3	x4	x5	x6	x7	
00	00	2A	02	00	00	00	00	00	
08	00	00	00	00	00	00	00	00	
10	00	00	00	00	00	00	00	00	
18	00	00	00	00	00	00	00	00	
20	2A	00	00	00	00	00	00	00	
28	02	00	15	00	00	00	00	00	
30	00	00	00	00	00	00	00	00	
38	00	00	00	00	00	00	00	00	
40	00	00	00	00	00	00	00	00	
	--	--	--	--	--	--	--	--	

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF	0123456789ABCDEF	
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	tQz... (y+uQ.w3..
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	*... (...7./
0020	63																c
0030																	

Mais uma vez o MOV, na linha 13 faz com que o valor 28h seja movido para o acumulador:

	x0	x1	x2	x3	x4	x5	x6	x7	
00	00	2A	02	00	00	00	00	00	
08	00	00	00	00	00	00	00	00	
10	00	00	00	00	00	00	00	00	
18	00	00	00	00	00	00	00	00	
20	2A	00	00	00	00	00	00	00	
28	02	00	15	00	00	00	00	00	
30	00	00	00	00	00	00	00	00	
38	00	00	00	00	00	00	00	00	
40	00	00	00	00	00	00	00	00	
--	--	--	--	--	--	--	--	--	

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	tQz...	(y+uQ.w3..														
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	*...	(...7./														
0020	63																c															
0030																																

Na linha 14, o valor de A é somado com o registrador DPTR e movido para o acumulador:

	x0	x1	x2	x3	x4	x5	x6	x7	
00	00	2A	02	00	00	00	00	00	
08	00	00	00	00	00	00	00	00	
10	00	00	00	00	00	00	00	00	
18	00	00	00	00	00	00	00	00	
20	2A	00	00	00	00	00	00	00	
28	02	00	15	00	00	00	00	00	
30	00	00	00	00	00	00	00	00	
38	00	00	00	00	00	00	00	00	
40	00	00	00	00	00	00	00	00	
	--	--	--	--	--	--	--	--	

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0000	74	51	7A	02	85	02	28	79	2A	75	51	0C	77	33	C4	C5	t	Q	z	.	.	.	.	.	.	.	.	.	.	.	.	.
0010	2A	85	01	20	90	00	1D	E5	28	93	02	00	00	37	0C	2F	*	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0020	63																c															
0030																																

Na sequência, o programa retorna a posição de memória armazenada no label INICIO e comando TAB, linha 16, posiciona 4 bytes ordenados a partir da última posição utilizada. O comando END na última linha do programa, informa ao compilador o encerramento da execução. Por último, segue abaixo o conteúdo do acumulador quando o conteúdo do PC estiver em INICIO pela terceira vez:

	HEX	DEC	BIN	OCT	CHAR
A:	2F	47	00101111	57	/
B:	00	0	00000000	0	

## CONCLUSÕES

O experimento foi realizado com sucesso diante de suas propostas, visto que os modos de endereçamento possuem grande importância para o microcontrolador, bem como seus operandos, constituindo partes imprescindíveis para a compilação ser executada sem erros pelo computador. Cada etapa de sua execução foi entendida e analisada, e considerada então uma prerrogativa básica para o entendimento da linguagem Assembly.

## REFERÊNCIAS BIBLIOGRÁFICAS

SCHMIDT, Gerhard. Introdução para o iniciante à Linguagem Assembly dos Microprocessadores ATMEL-AVR. Dezembro de 2003. Disponível em: [https://www.ic.unicamp.br/~ducatte/mc404/2009/docs/beginner\\_pt.pdf](https://www.ic.unicamp.br/~ducatte/mc404/2009/docs/beginner_pt.pdf). Acesso em: 24 jul. 2021



MORAIS, Misael. Organização e arquitetura de computadores. [S. l.], . 2021. Disponível em:

<https://drive.google.com/drive/folders/0Bwjlecok7TpyfnAtQmx6bE4yZ043amNsbnRxMkF4UFIpWVZhWmRfeVBSHRRVi1xRzNOZnM?resourcekey=0-WmQ6S1i6hLYyCoGBLbMeGw>.

Acesso em: 24 jul. 2021