

Display LCD

01. INTRODUÇÃO

Os módulos LCD são interfaces de saída muito útil em sistemas microprocessados. Estes módulos podem ser gráficos e a caracter. Os módulos LCD gráficos são encontrados com resoluções de 122x32, 128x64, 240x64 e 240x128 dots pixel, e geralmente estão disponíveis com 20 pinos para conexão. Os LCD comuns (tipo caracter) são especificados em número de linhas por colunas e são encontrados nas configurações previstas na Tabela 1.

Tabela 1 - Módulos LCD disponíveis

Número de Colunas	Número de Linhas	Quantidade de pinos
8	2	14
12	2	14/15
16	1	14/16
16	2	14/16
16	4	14/16
20	1	14/16
20	2	14/16
20	4	14/16
24	2	14/16
24	4	14/16
40	2	16
40	4	16

Os módulos podem ser encontrados com *LED backlight* (com uma iluminação de fundo) para facilitar as leituras durante a noite. Neste caso, a alimentação deste led faz-se normalmente pelos pinos 15 e 16 para os módulos comuns e 19 e 20 para os módulos gráficos, sendo os pinos 15 e 19 para ligação ao anodo e os pinos 16 e 20 para o catodo. A corrente de alimentação deste led varia de 100 a 200mA, dependendo do modelo.

Estes módulos utilizam um controlador próprio, permitindo sua interligação com com outras placas através de seus pinos, onde deve ser alimentado o módulo e interligado o barramento de dados e controle do módulo com a placa do usuário. Naturalmente que além de alimentar e conectar os pinos do módulo com a placa do

usuário deverá haver um protocolo de comunicação entre as partes, que envolve o envio de bytes de instruções e bytes de dados pelo sistema do usuário.

A Tabela 2 descreve cada pino do módulo ou do *display* para conexão deste a outras placas:

Tabela 2 - Pinagem dos Módulos LCD

Pino	Função	Descrição
1	Alimentação	Terra ou GND
2	Alimentação	VCC ou +5V
3	V0	Tensão para ajuste de contraste (ver Figura 1)
4	RS Seleção:	1 - Dado, 0 - Instrução
5	R/W Seleção:	1 - Leitura, 0 - Escrita
6	E Chip select	1 ou (1 → 0) - Habilita, 0 - Desabilitado
7	B0 LSB	Barramento de Dados
8	B1	
9	B2	
10	B3	
11	B4	
12	B5	
13	B6	
14	B7 MSB	
15	A (qdo existir)	Anodo p/ <i>LED backlight</i>
16	K (qdo existir)	Catodo p/ <i>LED backlight</i>

Assim como em um rádio relógio todo módulo LCD permite um ajuste na intensidade da luz emitida ou ajuste de contraste, isto é possível variando-se a tensão no pino 3. A Figura 1 mostra um circuito típico e recomendado pela maioria dos fabricantes para efetuar este ajuste. Alguns fabricantes recomenda o uso de um resistor de 4K7 em série com o potenciômetro de 10K.

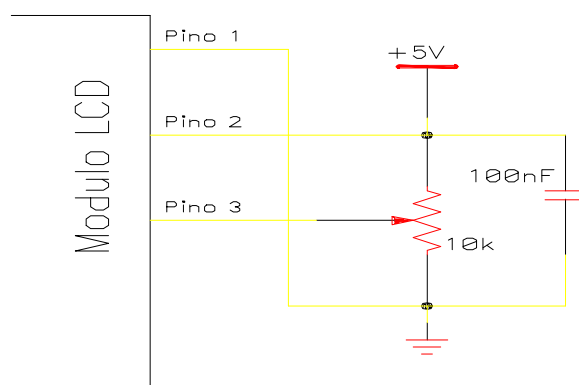
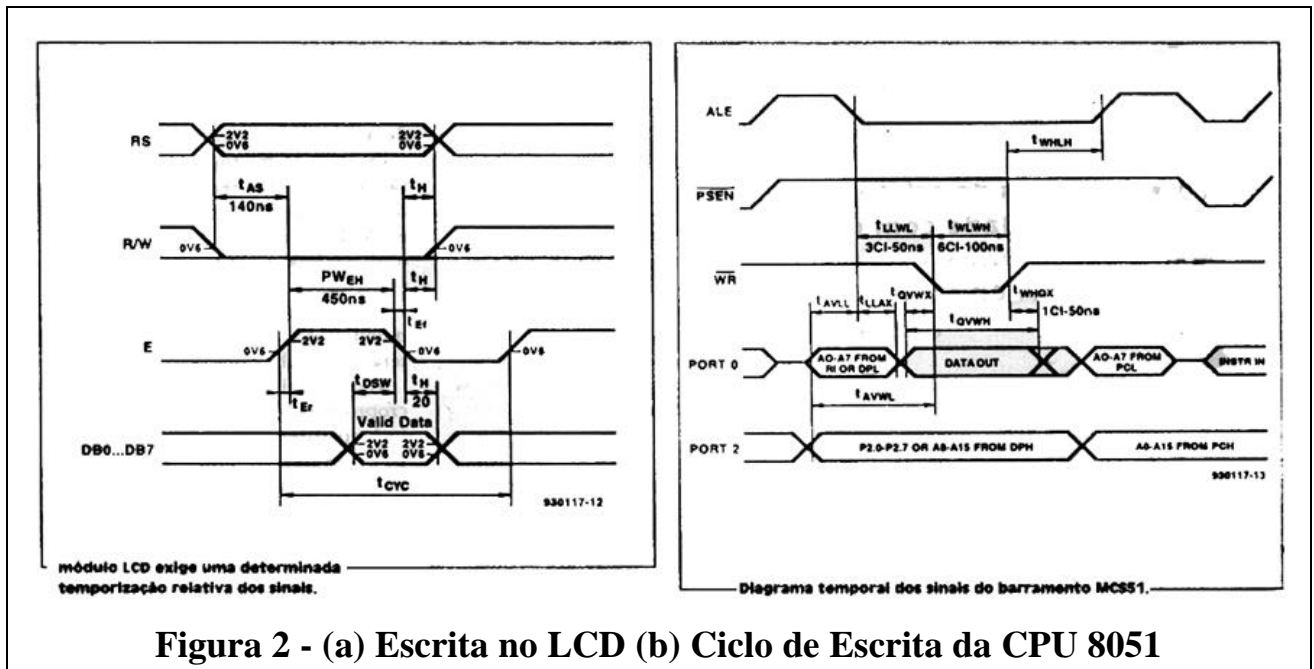


Figura 1 - Detalhe do controle de contraste do módulo LCD

02. INTERFACE COM CPU

Os módulos LCD são projetados para conectar-se com a maioria das CPU's disponíveis no mercado, bastando para isso que esta CPU atenda as temporizações de leitura e escrita de instruções e dados, fornecido pelo fabricante do módulo. A Figura 2 mostra um exemplo de diagrama de tempos típico requeridos para operação de escrita no módulo LCD, estes tempos variam em função do clock da CPU do usuário.



A Tabela 3 a seguir mostra a relação entre a frequência da CPU e a temporização de leitura/escrita da maioria dos módulos LCD. Em geral, podemos conectar o barramento de dados da CPU ao barramento do módulo, mapeando-o convenientemente na placa de usuário, e efetuarmos uma operação normal de leitura/escrita sem mais problemas.

Tabela 3 - Relação clock da CPU x Temporização do Módulo LCD

Clock da CPU	t_{AS} (MHz)	PW_{EH} (nS)	t_H (nS)
08 MHz	325	650	75
10 MHz	250	500	50
12 MHz	200	400	33,3
16 MHz	138	275	12,5

A Figura 3 mostra um exemplo de conexão de uma placa baseada nos microcontroladores da linha Intel de 8 bits (8051), ao módulo LCD. Neste caso como os

sinais A0 e A1 estão conectados aos pinos 4 e 5, teremos então 04 (quatro) endereços distintos para comunicação entre a CPU e o módulo LCD. A Tabela 4 mostra estes endereços.

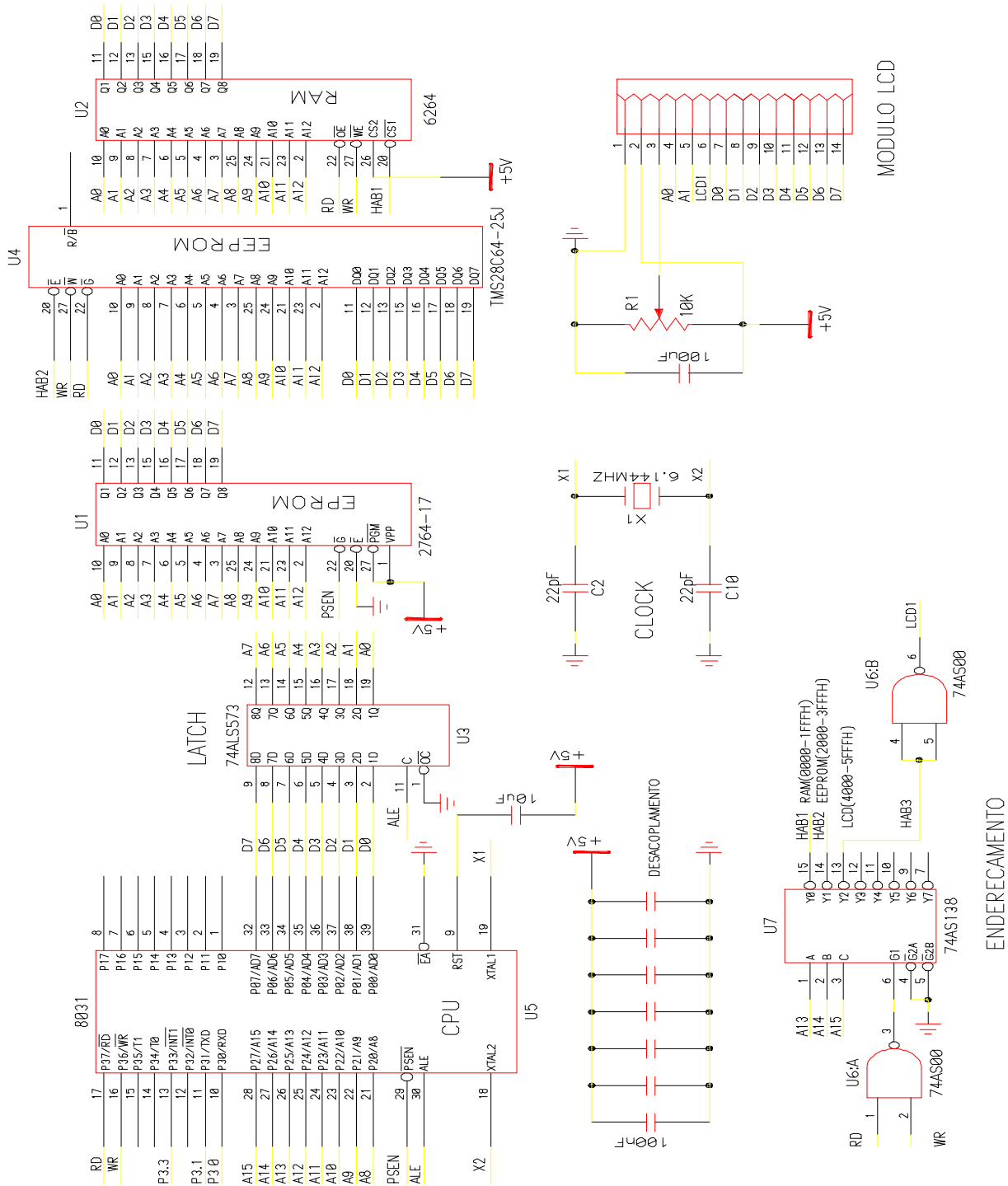


Figura 3 - Sistema baseado na CPU 8051 com módulo LCD

Tabela 4 - Endereçamento do módulo LCD para Figura 3

ENDEREÇO	R/W	RS	DESCRIÇÃO
----------	-----	----	-----------

4000	0	0	Instrução - Escrita no módulo
4001	0	1	Dados - Escrita no módulo
4002	1	0	Instrução - Leitura no módulo
4003	1	1	Dados - Leitura no módulo

O exemplo apresentado na Figura 3 refere-se a conexão do módulo LCD com comunicação/transmissão de 8 bits, mas podemos conectar o módulo com transmissão a cada 4 bits, conforme é mostrado na Figura 4. Neste caso não utilizamos os pinos 7, 8, 9 e 10. Isto é muito útil quando a CPU do usuário possui poucos pinos de I/O, caso típico da linha de microprocessadores PIC, como por exemplo o **Basic Stamp**. Agora surge a dúvida, um mesmo módulo pode conectar-se com 8 ou 4 bits? como isto é possível?

Ocorre que o módulo LCD quando alimentado necessita de algumas instruções de inicialização que identificará qual a forma de transmissão de dados que será estabelecida entre a CPU e o módulo.

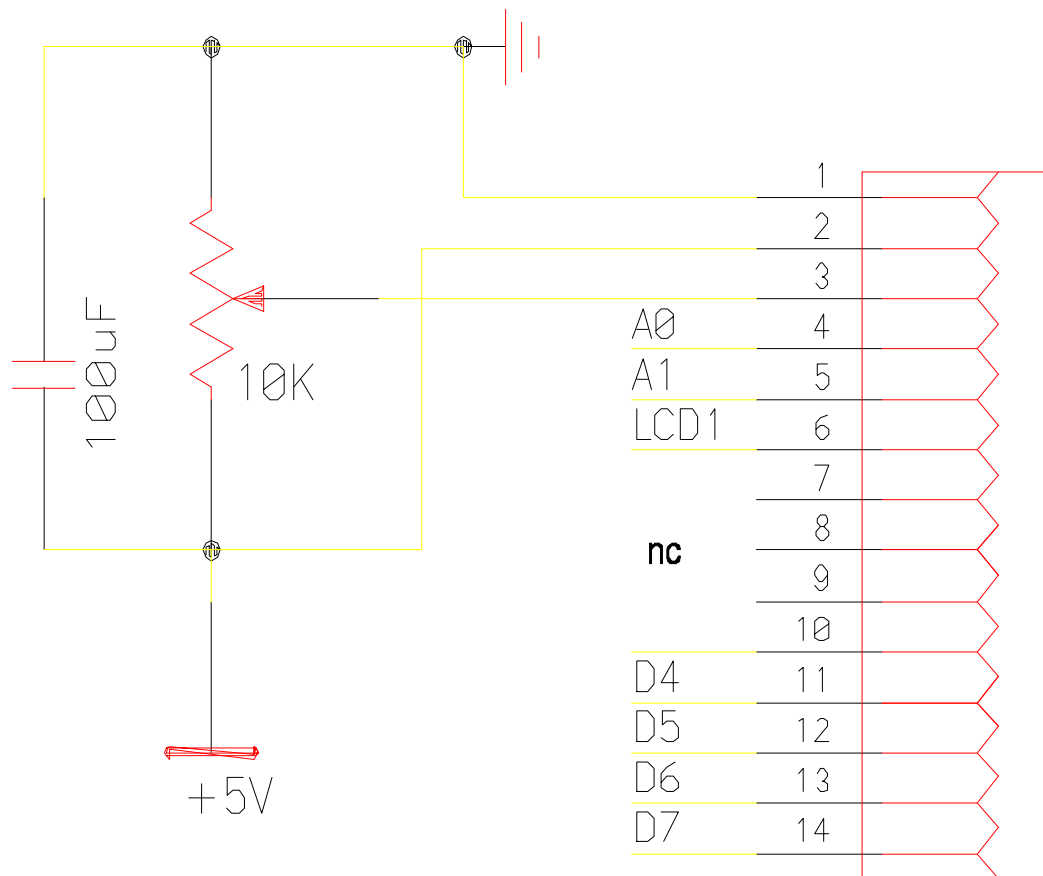


Figura 4 - Modulo LCD comunicando-se com 4 bits

03. PROGRAMAÇÃO / INSTRUÇÕES

Tabela 5 - Conjunto de instruções do módulo LCD

INSTRUÇÃO	RS	R/W	B7	B6	B5	B4	B3	B2	B1	B0	DESCRIÇÃO e tempo de execução (uS)	t
Limpa Display	0	0	0	0	0	0	0	0	0	1	-Limpa todo o display e retorna o cursor para a primeira posição da primeira linha	1.6 mS
Home p/ Cursor	0	0	0	0	0	0	0	0	1	*	-Retorna o cursor para a 1. coluna da 1. Linha -Retorna a mensagem previamente deslocada a sua posição original	1.6 mS
Fixa o modo de funcionamento	0	0	0	0	0	0	0	1	X	S	-Estabelece o sentido de deslocamento do cursor (X=0 p/ esquerda, X=1 p/ direita) -Estabelece se a mensagem deve ou não ser deslocada com a entrada de um novo caracter (S=1 SIM, X=1 p/ direita) -Esta instrução tem efeito somente durante a leitura e escrita de dados.	40 uS
Controle do Display	0	0	0	0	0	0	1	D	C	B	-Liga (D=1) ou desliga display (D=0) -Liga(C=1) ou desliga cursor (C=0) -Cursor Piscante(B=1) se C=1	40 uS
Desloca cursor ou mensagem	0	0	0	0	0	1	C	R	*	*	-Desloca o cursor (C=0) ou a mensagem (C=1) para a Direita se (R=1) ou esquerda se (R=0) - Desloca sem alterar o conteúdo da DDRAM	40 uS
Fixa o modo de utilização do módulo LCD	0	0	0	0	1	Y	N	F	*	*	-Comunicação do módulo com 8 bits(Y=1) ou 4 bits(Y=0) -Número de linhas: 1 (N=0) e 2 ou mais (N=1) -Matriz do caracter: 5x7(F=0) ou 5x10(F=1) - Esta instrução deve ser ativada durante a inicialização	40 uS
Posiciona no endereço da CGRAM	0	0	0	1	Endereço da CGRAM						-Fixa o endereço na CGRAM para posteriormente enviar ou ler o dado (byte)	40 uS
Posiciona no endereço da DDRAM	0	0	1	Endereço da DDRAM						-Fixa o endereço na DDRAM para posteriormente enviar ou ler o dado (byte)	40 uS	
Leitura do Flag Busy	0	1	B F	AC						-Lê o conteúdo do contador de endereços (AC) e o BF . O BF (bit 7) indica se a última operação foi concluída (BF=0 concluída) ou está em execução (BF=1).	0	
Escreve dado na CGRAM / DDRAM	0	1	Dado a ser gravado no LCD								- Grava o byte presente nos pinos de dados no local apontado pelo contador de endereços (<i>posição do cursor</i>)	40 uS
Lê Dado na CGRAM / DDRAM	1	1	Dado lido do módulo								- Lê o byte no local apontado pelo contador de endereços (<i>posição do cursor</i>)	40 uS

A Tabela 5 apresenta o conjunto de instruções, levando-se em consideração que a comunicação com o módulo seja com barramento de 8 bits (fixado durante a inicialização). Para o caso desta comunicação ocorrer com apenas 4 bits (nible), os

dados ou instruções serão enviados por **nible**. sendo enviado o *nible mais significativo primeiro*. Por exemplo para limpar o display, escreve-se o nible 0000 e depois 0001.

A Tabela 6 traz um resumo das instruções mais usadas na comunicação com os módulos LCD.

Tabela 6 - Instruções mais comuns

DESCRIÇÃO	MODO	RS	R/W	Código (Hexa)
Display	Liga (sem cursor)	0	0	0C
	Desliga	0	0	0A / 08
Limpa Display com Home cursor		0	0	01
Controle do Cursor	Liga	0	0	0E
	Desliga	0	0	0C
	Desloca para Esquerda	0	0	10
	Desloca para Direita	0	0	14
	Cursor Home	0	0	02
	Cursor Piscante	0	0	0D
	Cursor com Alternância	0	0	0F
Sentido de deslocamento do cursor ao entrar com caracter	Para a esquerda	0	0	04
	Para a direita	0	0	06
Deslocamento da mensagem ao entrar com caracter	Para a esquerda	0	0	07
	Para a direita	0	0	05
Deslocamento da mensagem sem entrada de caracter	Para a esquerda	0	0	18
	Para a direita	0	0	1C
End. da primeira posição	primeira linha	0	0	80
	segunda linha	0	0	C0

3.1 - DESCRIÇÃO DETALHADA DAS INSTRUÇÕES

3.1.1- Limpa Display

	R	R/W	B7	B6	B5	B4	B3	B2	B1	B0
	S									
CÓDIGO	0	0	0	0	0	0	0	0	0	1
			MSB				LSB			

Esta instrução escreve o caracter ASCII 32 que corresponde ao branco ou barra de espaço em todos os endereços da DDRAM apagando a mensagem que estiver escrita. O cursor retorna ao endereço “zero”, ou seja, à posição mais a esquerda da primeira linha.

3.1.2 - Cursor Home

	R	R/W	B7	B6	B5	B4	B3	B2	B1	B0
CÓDIGO	S		0	0	0	0	0	0	1	*
	0	0	MSB				LSB			

Faz retornar o cursor para a posição mais a esquerda da primeira linha e faz voltar à posição original mensagens previamente deslocadas. O conteúdo da DDRAM permanece inalterado.

3.1.3 - Fixa o modo de operação

	R	R/W	B7	B6	B5	B4	B3	B2	B1	B0
CÓDIGO	S		0	0	0	0	0	1	X	S
	0	0	MSB				LSB			

Esta instrução tem efeito somente durante a leitura ou escrita de dados, portanto, deve ser ativada na inicialização.

- Estabelece o sentido de **deslocamento do cursor** (X=0 p/ esquerda, X=1 p/ direita)
- Estabelece se a **mensagem** deve ou não ser deslocada com a entrada de um novo caracter

S=1 SIM, **S=0** NÃO. Exemplo: X=1 e S=1 => mensagem desloca p/ direita.

3.1.4 - Controle do Display

	R	R/W	B7	B6	B5	B4	B3	B2	B1	B0
CÓDIGO	S		0	0	0	0	0	D	C	B
	0	0	MSB				LSB			

A mensagem fica aparente quando D=1 e desaparece quando D=0, porém o conteúdo da DDRAM fica inalterado. O cursor fica aparente quando C=1 e desaparece quando C=0, porém as propriedades de escritas vigentes permanecem inalteradas. O cursor quando aparente liga a última linha que compõem o caracter, exceto quando B=1, que apresenta em alternância com uma matriz com todos os pontos negros em intervalos de 0,4 segundos. Quando B=1 e C=0, obteremos a ativação intermitente de uma matriz completa (todos os pontos da matriz).

3.1.5 - Deslocamento do Cursor ou da Mensagem

	R S	R/W	B7	B6	B5	B4	B3	B2	B1	B0
CÓDIGO	0	0	0	0	0	1	C	R	*	*
			MSB				LSB			

Desloca o cursor ou a mensagem sem que para isso tenha que escrever ou ler dados do display. Utilizado para posicionamento dos dados no display.

C	R	FUNÇÃO
0	0	Desloca o cursor para a esquerda e decrementa o contador de endereço.
0	1	Desloca o cursor para a direita e incrementa o contador de endereço.
1	0	Desloca a mensagem e o cursor para a esquerda.
1	1	Desloca a mensagem e o cursor para a direita

3.1.6 - Estabelece o modo de utilização do Módulo LCD

	R S	R/W	B7	B6	B5	B4	B3	B2	B1	B0
CÓDIGO	0	0	0	0	1	Y	N	F	*	*
			MSB				LSB			

Y estabelece o modo de comunicação. Se $Y=1$ estabelece 8 bits e quando $Y=0$ será 4 bits, enviados em duas operações, com os 4 bits (Nibble) mais significativos sendo enviados primeiro. **N** fixa o número de linhas: $N=0$ para uma linha e $N=1$ para duas ou mais linhas. **F** fixa o tipo da matriz: $F=0$ para matriz 7x5 ou 8x5 e $F=1$ para matriz 10x5 (somente possível quando apresentando em uma linha).

3.1.7 - Endereçamento da CGRAM

CGRAM é uma região da memória RAM destinada para criação de caracteres especiais, como por exemplo: ç, é, Ê, etc.

	R S	R/W	B7	B6	B5	B4	B3	B2	B1	B0
CÓDIGO	0	0	0	1	A	A	A	A	A	A
			MSB				LSB			

Estabelece o endereço da CGRAM no contador de endereços (AC) como um número binário AAAAAA e após isto os dados serão escritos ou lidos pela CPU neste endereço. Cada caracter especial ocupa 8 endereços na CGRAM.

3.1.8 - Endereçamento da DDRAM

	R	R/W	B7	B6	B5	B4	B3	B2	B1	B0
CÓDIGO	S									
	0	0	1	A	A	A	A	A	A	A
			MSB				LSB			

Estabelece o endereço da DDRAM no contador de endereços (AC) como um número binário AAAAAAA e após isto os dados serão escritos ou lidos pela CPU neste endereço. Para os display de uma linha AAAAAAA varia de 80H a CFH. Já para todos os display de duas linhas varia de 80H a A7H para a primeira linha e de C0H a E7H para a segunda linha.

3.1.9 - Busy Flag (BF)

	R	R/W	B7	B6	B5	B4	B3	B2	B1	B0
CÓDIGO	S									
	0	1	BF	A	A	A	A	A	A	A
			MSB				LSB			

Busy Flag ou o bit 7 indica ao sistema onde está conectado o módulo LCD, se o controlador do módulo está ocupado com alguma operação interna (BF=1), e neste caso, não aceita nenhuma instrução até que BF volte para 0.

Além disso, permite a leitura do conteúdo do contador de endereços (AC) expressa por AAAAAAA. O contador de endereços pode conter tanto endereço da CGRAM como da DDRAM, depende neste caso, da instrução anterior.

3.1.10 - Escrita de dados na DDRAM ou CGRAM

	R	R/W	B7	B6	B5	B4	B3	B2	B1	B0
CÓDIGO	S									
	1	0	A	A	A	A	A	A	A	A
			MSB				LSB			

Escreve o byte AAAAAAAA tanto na CGRAM como na DDRAM, dependendo da instrução anterior (que define o endereço). Após a escrita, o endereço é automaticamente incrementado ou decrementado de uma unidade dependendo do modo escolhido (ver item 3.1.3).

3.1.11 - Leitura de dados na DDRAM ou CGRAM

	R	R/W	B7	B6	B5	B4	B3	B2	B1	B0
	S									
CÓDIGO	1	1	A	A	A	A	A	A	A	A
			MSB				LSB			

Faz uma leitura na CGRAM ou na DDRAM, dependendo da instrução anterior (que define o endereço). É importante que precedendo a esta leitura seja executado a instrução de estabelecimento do endereço da CGRAM ou DDRAM, pois caso contrário o dado lido é inválido.

3.2 - TABELAS DE ENDEREÇOS DOS CARCTERES NA DDRAM

A seguir resumiremos os endereços da DDRAM (em hexadecimal) dos caracteres da maioria dos módulos LCD disponíveis no mercado.

		caracter							
Módulo 8x1	1	2	3	4	5	6	7	8	
Endereço(hexa)	80	81	82	83	84	85	86	87	

LCD 16x1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F

LCD 16x1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	80	81	82	83	84	85	86	87	C0	C1	C2	C3	C4	C5	C6	C7

LCD 16x2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
linha 1	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
linha 2	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF

LCD 20x2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
linha 1	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	90	91	92	93
linha 2	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF	D0	D1	D2	D3

LCD 20x4	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
linha 1	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	90	91	92	93

linha 2	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF	D0	D1	D2	D3
linha 3	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	90	91	92	93
linha 4	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF	D0	D1	D2	D3

LCD 24x1 LCD 24x2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
linha 1	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	90	91	92	93	94	95	96	97
linha 2	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF	D0	D1	D2	D3	D4	D5	D6	D7

LCD 40x1 a LCD 40x4	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
linha 1	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	90	91	92	93
linha 2	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF	D0	D1	D2	D3
linha 3	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	90	91	92	93
linha 4	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF	D0	D1	D2	D3
Cont.	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
linha 1	94	95	96	97	98	99	9A	9B	9C	9E	9D	9F	A0	A1	A2	A3	A4	A5	A6	A7
linha 2	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF	E0	E1	E2	E3	E4	E5	E6	E7
linha 3	94	95	96	97	98	99	9A	9B	9C	9E	9D	9F	A0	A1	A2	A3	A4	A5	A6	A7
linha 4	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF	E0	E1	E2	E3	E4	E5	E6	E7

OBS:

- Para os módulos de 04 linhas estamos considerando que existe um outro pino de habilitação (como o pino 6) para as duas últimas linhas, portando outros endereços de hardware.
- Antes de enviar uma instrução para escrita de dados no display, enviar antes uma de endereçamento na DDRAM, com o endereço onde deve ser escrito o caracter, tipo um *gotoxy()*.

3.3 - TABELAS DE ENDEREÇOS DOS CARCTERES NA CGRAM

Os caracteres especiais previamente programado, durante a inicialização, podem ser utilizados a qualquer tempo como se fossem caracteres normais, lembrando que os **endereços bases** em *hexadecimal* para gravação dos caracteres especiais, na maioria dos módulos LCD, são respectivamente: 40, 48, 50, 58, 60, 68, 70 e 78. Cada caracter especial ocupa 8 (oito) endreços.

Tabela 7 - Caracter especial {ç} na CGRAM

	RS	R/W	Dado em Binário								HEXA
Endereço 50	0	0	0	1	0	1	0	0	0	0	50
50	1	0					X	X	X		0E
51	1	0					X			X	11
52	1	0					X				10
53	1	0					X				10

54	1	0		x		x		x	15
55	1	0			x	x	x		0E
56	1	0		x					10
57	1	0							00*

Isto significa que para utilizarmos o caracter gravado no endereço base 50, durante a inicialização ou reset do sistema, teremos que escrevermos 8 bytes entre os endereços 50 e 57, para construirmos o caracter. Para ilustrar este procedimento, supor que queiramos construir o caracter {ç} no endereço base 50. Neste caso, devemos construir o mapa deste caracter especial como mostrado na Tabela 7 (supondo estar trabalhando com matriz 7x5 e com auto incremento de endereço a cada escrita). Observe que o último endereço sempre será 00, pois esta posição é sempre ocupada pelo cursor.

4 - INICIALIZAÇÃO DOS MÓDULOS LCD

Toda vez que alimentamos o módulo LCD deve ser executado o procedimento de inicialização, que consiste no *envio de uma seqüência de instruções* para configurar o modo de operação para execução de um dado programa de interfaceamento. Em muitos display este procedimento ocorre automaticamente, dentro de condições específicas que envolve temporizações mínimas referente a transição do nível lógico 0 para 1, ao ligarmos a fonte. Em caso de dúvidas, recomendamos o envio destas instruções após o reset do sistema.

1 - Inicialização para sistemas 8 bits de dados (5 instruções)

Entre as duas primeiras instruções recomendamos um *delay de 15 mS*. As demais instruções podem ser escritas após checar o **Busy Flag**.

		Instruções em Hexadecimal (8 bits)							
MÓDULO LCD	1		2		3		4		5
1 linha - Matriz 7x5 e 8x5	30	15mS	30	15mS	06	BF	0E	BF	01
1 linha - Matriz 10x5	34	15mS	34	15mS	06	BF	0E	BF	01
2 linha - Matriz 7x5 e 8x5	38	15mS	38	15mS	06	BF	0E	BF	01

3.4.2 - Inicialização para sistemas 4 bits de dados (5 instruções)

Entre as quatro primeiras instruções recomendamos um *delay de 15 mS*. As demais instruções podem ser escritas após checar o **Busy Flag**. Estes bits (*nible*) devem estar conectados aos pinos 11, 12.13 e 14.

Instruções em Hexadecimal (4 bits)

MÓDULO LCD	1	2	3	4	5	6	7	8	9	10	11	12
1 linha - Matriz 7x5 e 8x5	3	3	3	2	2	0	0	8	0	1	0	1
1 linha - Matriz 10x5	3	3	3	2	2	4	0	8	0	1	0	1
2 linha - Matriz 7x5 e 8x5	3	3	3	2	2	8	0	8	0	1	0	1

04. ROTEIRO PARA PROGRAMAÇÃO

A seguir passaremos a descrever um resumo dos procedimentos para utilização de um módulo ou display LCD:

- 1 Ao energizar o módulo ajuste o potenciômetro de controle do brilho ou contraste até obter a visualização da matriciação na primeira linha para módulo de duas linhas ou até a matriciação de meia linha para módulos de uma linha.
- 2 Alguns módulos de uma linha só funcionam com a instrução 38 ao invés de 30, conforme instruções de inicialização.
- 3 O sinal de **Enable** (pino 6) deverá ser gerado conforme a temporização mostrada na Figura 2. Os códigos de dados ou de instruções só serão processados pelo processador do módulo após a descida do sinal do **Enable**.
- 4 Para ajustar a velocidade de comunicação entre a CPU do usuário e o módulo LCD existem duas possibilidades:
 - Intercalar uma rotina de atraso de aproximadamente 15 mS entre as instruções.
 - Fazer a leitura do **Busy Flag** antes do envio de cada instrução e só enviar quando o mesmo for 0. Neste caso, a única exceção será durante a inicialização.
- 1 Durante a inicialização enviar a seqüência correta das instruções de inicialização conforme item 3.4
- 2 Para programar caracteres na CGRAM, faça inicialmente o endereçamento da mesma.
- 3 Após a escrita de dados na CGRAM envie a instrução 01, para posicionar o cursor.
- 4 Para escrever os caracteres especiais previamente gravados na CGRAM, utilize os códigos de 00 até 07 correspondente aos *endereços bases de 40, 48 até 78* em hexa.
- 5 Comandos úteis:

FIXAÇÃO DAS CONDIÇÕES DE UTILIZAÇÃO	Instrução
1 linha 5x7 (8 bits)	30H
2 linha 5x7 (8 bits)	38H
1 linha 5x10 (8 bits)	34H
1 linha 5x7 (4 bits)	20H
2 linha 5x7 (4 bits)	28H

1 linha 5x10 (4 bits)	24H
-----------------------	-----

CONTROLE DISPLAY	Instrução
Display aceso c/ cursor fixo	0EH
Display aceso c/ cursor intermitente	0FH
Display aceso sem cursor	0CH
Display apagado	08H

MODO DE OPERAÇÃO	Instrução
Escreve deslocando a mensagem para esquerda (cursor fixo)	07H
Escreve deslocando a mensagem para a direita (cursor fixo)	05H
Escreve deslocando o cursor para a direita	06H
Escreve deslocando o cursor para a esquerda	04H

OUTROS COMANDOS ÚTEIS	Instrução
Limpa display e retorna o cursor para o inicio	01H
Retorna o cursor para o inicio (sem alterar a DDRAM)	02H
Desloca somente o cursor para a direita	14H
Desloca somente o cursor para a esquerda	10H
Desloca o cursor + mensagem para a direita	1CH
Desloca o cursor + mensagem para a esquerda	18H
Desloca o cursor para posição inicial da segunda linha	C0H
Desloca o cursor para posição inicial da primeira linha	80H

CGRAM (caracteres especiais)	Instrução
Endereço inicial para construir caracteres especiais	40H
Para escrever o primeiro caracter (previamente construídos)	00H
Para escrever o último caracter (previamente construídos)	07H

Obs:

Após o endereçamento da CGRAM, o cursor se desloca para a primeira posição da segunda linha (ou metade), portanto é recomendado enviar a instrução 01 ou “limpa display e cursor home”.

05. CUIDADOS ESPECIAIS COM MÓDULOS LCD

1 - MANUSEIO

- Somente retire o módulo de sua embalagem protetora imediatamente antes de sua instalação
- Não guarde os módulos em recintos de alta temperatura e alta umidade. A temperatura de armazenamento deverá estar compreendida entre 5 e 30 °C.
- O LCD é coberto por uma lâmina plástica polarizada a qual não pode ser riscada. Cuidado em seu manuseio. Para a limpeza da lâmina utilize cotonetes embebido em benzina. Não utilize outros tipos de solventes.
- Observe cuidadosamente os procedimentos de controle anti-estático quando manusear os módulos. Eles incorporam circuitos integrados CMOS LSI os quais são sensíveis à descarga eletrostática. Não toque nos terminais do conector, trilhas do circuito impresso e/ou terminais do CI.

2 - INSTALAÇÃO

- Nunca desmonte o módulo
- Use uma estação de solda aterrada para soldagem de conectores ou terminais.
- montador deverá também ser convenientemente aterrado.
- Sempre que o projeto o permita, instale o módulo atrás de uma janela protetora de plástico ou vidro.
- Somente retire a fita adesiva que protege a lâmina plástica frontal imediatamente antes de seu uso.

5.3 - OPERAÇÃO

- Nunca instale ou desconecte o módulo com sua alimentação ligada.
- Sempre opere os módulos respeitando sua gama de temperatura de operação.
- Observe cuidadosamente os valores das tensões de alimentação e os níveis dos sinais de controle.
- Ajuste a tensão no pino 3 (V0) para obter o contraste mais conveniente para uma dada aplicação.

06. EXEMPLO DE PROGRAMAÇÃO

Baseado no hardware da Figura 3, elaborar um programa em assembler para o 8031 que escreva no módulo LCD a seguinte mensagem: “Vila Nova”.

```
=====
; Programa:      Exemplo para manuseio de display ou modulo LCD      =
; Data:         Out/98                                             =
; Descricao:                                         =
;               Este programa grava a mensagem "Kit Didatico"      =
;               no display LCD 2x20 (duas linha de 20 caracteres). =
;               Os pinos de dados do modulo LCD estao conectados   =
;               diretamente no barramento de dados da CPU, conforme =
;               Figura 3 da apostila sobre LCD.                    =
=====
$mod51
lcd_wr_inst equ 4000H
lcd_wr_dado equ 4001H
lcd_rd_inst equ 4002H
lcd_rd_dado equ 4003H
=====
; Programa principal
=====
    ORG 0H
    MOV SP,#50H
    MOV TMOD,#11H          ; TIMER1 / TIMER0 no modo 1
    CLR EA                 ; desabilita todas as interrupcoes
    CALL inicia_lcd
    MOV R7,#0FFH
loop:
    INC R7
    MOV A,R7
    MOV DPTR,#mensagem
    MOVC A,@A+DPTR        ; le caracter da mensagem
    CJNE A,#0FFH,cont     ; testa se fim da mensagem
    JMP fim
cont:
    CALL wr_dado           ; escreve o conteudo de acumulador no LCD
    CALL espera           ; aguarda busy flag
    JMP loop
fim:
    JMP fim               ; fim do programa
=====
; Rotinas chamadas pelo Programa Principal
=====
    ORG 100H
inicia_lcd:                ; rotina de inicializacao
    MOV A,#38H            ; 2 linhas / matriz 7x5
    CALL wr_inst
    CALL tempo            ; delay de 15mS
    MOV A,#38H            ; 2 linhas / matriz 7x5
    CALL wr_inst
    CALL tempo            ; delay de 15mS
    MOV A,#06H            ; cursor com autoincremento para direita
    CALL wr_inst
    CALL espera           ; busy flag
    MOV A,#0EH            ; liga display / cursor
```

```

CALL wr_inst
CALL espera           ; busy flag
MOV A,#01H           ; limpa display / cursor home
CALL wr_inst
CALL espera           ; busy flag
RET

espera:               ; rotina para aguardar busy flag
CALL rd_inst         ; ler busy flag e retorna em A
RLC A                ; bit7=BF vai para o carry
JC espera            ; se BF=1 => verifica novamente
RET

wr_inst:              ; escreve a instrucao em A no LCD
MOV DPTR,#lcd_wr_inst
MOVX @DPTR,A
RET

wr_dado:              ; escreve o dado em A no LCD
MOV DPTR,#lcd_wr_dado
MOVX @DPTR,A
RET

rd_inst:              ; le o Busy Flag e o contador de endereco
MOV DPTR,#lcd_rd_inst
MOVX A,@DPTR         ; retorna em A
RET

rd_dado:              ; le o dado / de qual endereco?
MOV DPTR,#lcd_rd_dado ; depende da ultima operacao no LCD
MOVX A,@DPTR         ; retorna em A
RET

;=====
; Rotina de tempo: 15 mS / clock 12 MHz => 50.000 ciclos
;=====
tempo:
MOV DPTR,#15536      ; (65536-50000)
MOV TH1,DPH
MOV TL1,DPL
SETB TR1             ; dispara timer1
JNB TF1,$            ; aguarda estouro => 50.0000
CLR TR1
CLR TF1              ; prepara para proxima chamada
RET

mensagem:            ; mensagem a ser escrita no LCD
DB 'Kit didatico', 0FFH
END                  ; obrigatorio

```