



UFPEL

Microcontroladores

Aula 8 – Comunicação com display de LCD

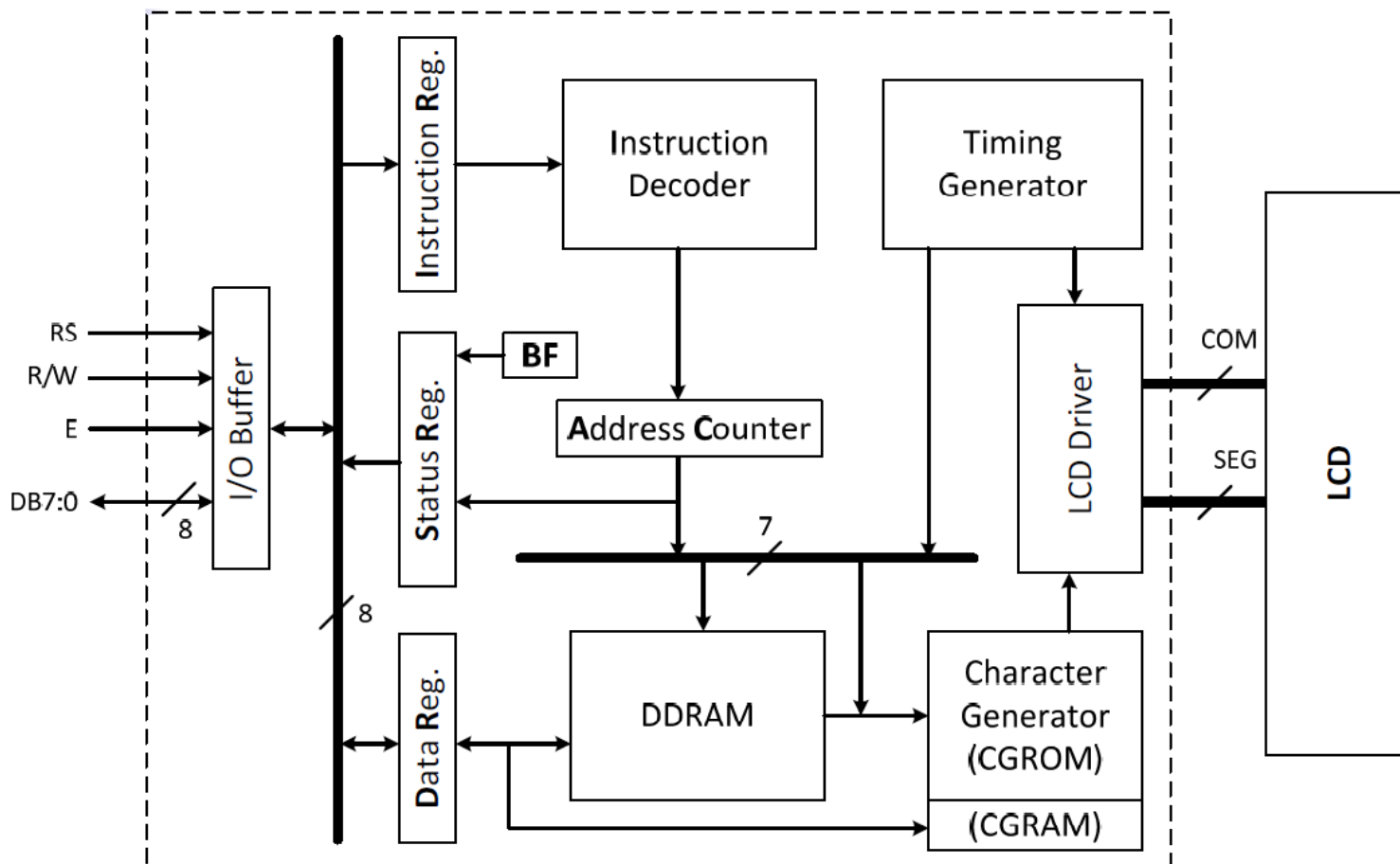
Prof. Dr. Alan Carlos Junior Rossetto

- Um display de LCD (*liquid crystal display*) é um exemplo de periférico gráfico que pode ser conectado externamente ao microcontrolador;
- Este tipo de dispositivo não gera interrupção, mas possui os registradores típicos de um periférico;
- O uC escreve na memória interna do dispositivo tanto os comandos de controle quanto os dados a serem exibidos;
- Esta memória configura o controlador e armazena os caracteres que devem ser mostrados no LCD.



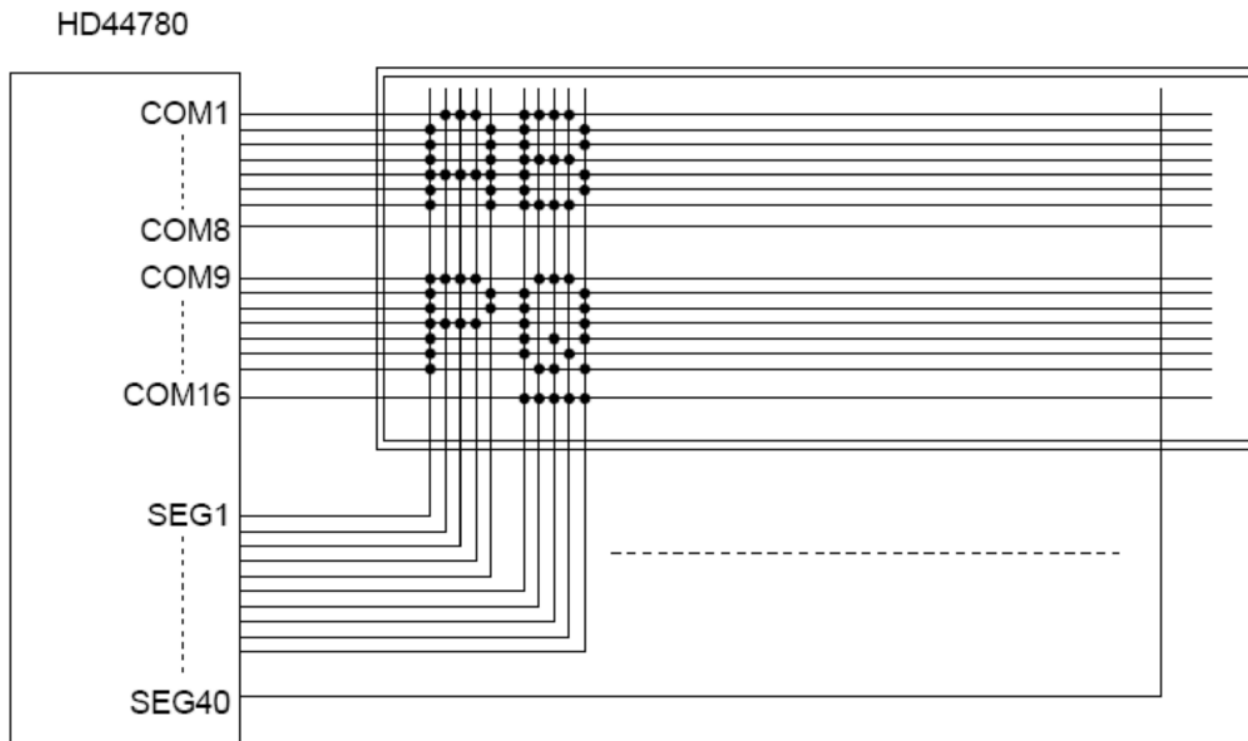
Esquemático elétrico interno

- O diagrama de blocos interno de um display de LCD é mostrado a seguir:



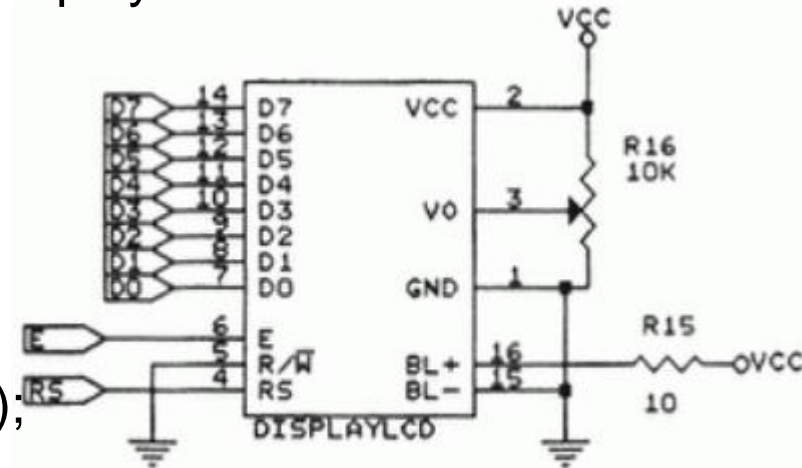
Esquemático elétrico interno

- A representação gráfica na tela do LCD é realizada pelo controlador (driver) que aciona posições individuais dentro de uma célula do display de forma a representar um caractere;
- Uma célula pode conter 5 x 7 ou 5 x 10 dessas posições individuais;
- Por padrão, os caracteres e símbolos que são enviados ao display devem estar codificados em ASCII.

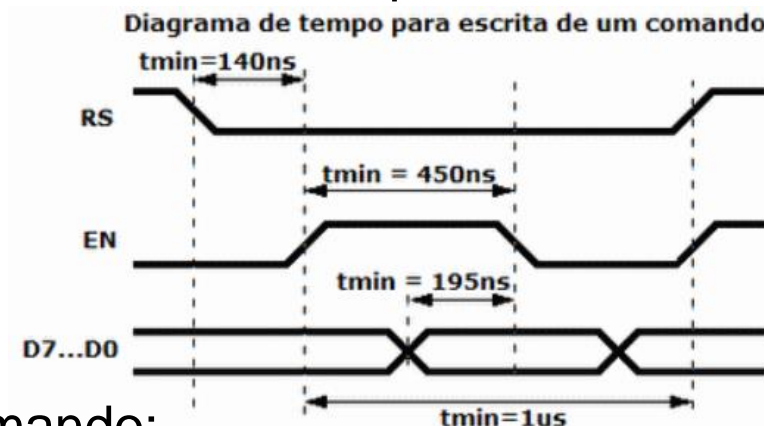


Esquemático elétrico externo

- A interface de conexão elétrica de um display de LCD é mostrada a seguir e composta por:
- Pinos (barramento) de dados:
 - D7...D0;
- Pinos de controle:
 - EN: Pino de ativação (dado pronto);
 - RS: Pino de diferenciação entre comando e caractere;
 - R/W: Pino de leitura ou escrita.
- Pinos de alimentação:
 - VCC e GND;
- Pinos de iluminação:
 - BL+ e BL-: backlight;
 - VO: contraste.



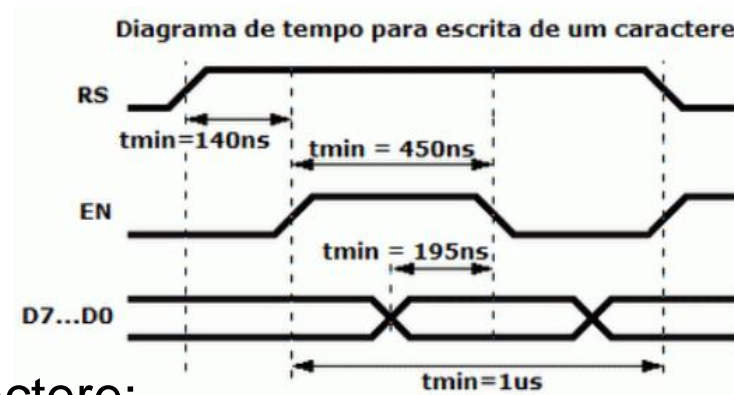
- Como o display recebe dados e sinais de controle pelo mesmo barramento, é preciso utilizar alguns sinais elétricos para diferenciação de um ou outro;
- Ademais, um protocolo de temporização precisa ser observado, a fim de que a comunicação entre uC e LCD ocorra sem problemas;
- Para a escrita de um comando, temos:
 - $R/\!W = 0$;
 - $RS = 0$;
 - Pulso no EN.



- Exemplo de função para escrita de comando:

```
WR_CMD:      CLR RS          ; RS = 0.
              NOP             ; Espera 1 us.
              SETB EM         ; EN = 1.
              LCALL DELAY_5US ; Espera 5 us.
              CLR EM          ; EN = 0.
              LCALL DELAY_5MS ; Espera 5 ms.
              RET              ; Retorna da função.
```

- Como o display recebe dados e sinais de controle pelo mesmo barramento, é preciso utilizar alguns sinais elétricos para diferenciação de um ou outro;
- Ademais, um protocolo de temporização precisa ser observado, a fim de que a comunicação entre uC e LCD ocorra sem problemas;
- Para a escrita de um caractere, temos:
 - $R/\!W = 0$;
 - $RS = 1$;
 - Pulso no EN.
- Exemplo de função para escrita de caractere:



```
WR_CHAR:      SETB RS
               NOP
               SETB EN
               LCALL DELAY_5US
               CLR  EM
               LCALL DELAY_5MS
               RET
```

Caracteres em ASCII

- Como dito anteriormente, os caracteres a serem mostrados no display devem estar codificados em ASCII. A seguir, é mostrada a representação destes em um display de células 5x7.

Lower 8 Bits	Upper 4 Bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000	CG RAM (1)			0	@	P	`	P					-	9	E	o	p
xxxx0001	(2)			!	1	A	Q	a	q				.	7	4	ã	q
xxxx0010	(3)			"	2	B	R	b	r				「	イ	ウ	×	þ
xxxx0011	(4)			#	3	C	S	c	s				」	ウ	テ	ε	ω
xxxx0100	(5)			\$	4	D	T	d	t				、	I	ト	μ	α
xxxx0101	(6)			%	5	E	U	e	u				*	オ	ナ	ι	ü
xxxx0110	(7)			&	6	F	V	f	v				ヲ	カ	ニ	ヨ	ρ
xxxx0111	(8)			'	7	G	W	g	w				フ	キ	ヌ	ラ	π
xxxx1000	(1)			(8	H	X	h	x				イ	ク	ホ	リ	Σ
xxxx1001	(2))	9	I	Y	i	y				お	ク	ル	リ	γ
xxxx1010	(3)			*	:	J	Z	j	z				エ	コ	ル	レ	7
xxxx1011	(4)			+	;	K	L	k	l				オ	サ	ヒ	ロ	π
xxxx1100	(5)			,	<	L	#	1	l				ハ	シ	フ	ワ	φ
xxxx1101	(6)			-	=	M	J	m	}				ユ	ズ	ン	ン	÷
xxxx1110	(7)			.	>	N	^	n	÷				ヨ	セ	ホ	ン	π
xxxx1111	(8)			/	?	O	_	o	+				ッ	リ	マ	ン	π

Configuração do *display*

- Antes da utilização, é necessário configurar o display de LCD de acordo com as suas características e/ou com o formato de exibição desejado através do envio de alguns comandos ao display;
- Quanto às características físicas, podemos configurar:

Instrução	D7	D6	D5	D4	D3	D2	D1	D0
Configurações físicas	0	0	1	DL	N	F	0	0

- DL: número de bits no barramento de dados
 - DL = 0: 4 bits (MSBs da porta) | DL = 1: 8 bits.
- N: número de linhas do display
 - N = 0: Uma linha | N = 1: Duas linhas.
- F: número de pontos na matriz de caracteres
 - F = 0: 5x7 pontos | F = 1: 5x10 pontos.
- Tipicamente DL = 1; N = 1; F = 0 -> (00111000)b = 38h

Configuração do *display*

- Antes da utilização, é necessário configurar o display de LCD de acordo com as suas características e/ou com o formato de exibição desejado através do envio de alguns comandos ao display;
- Quanto às características gráficas, podemos configurar:

Instrução	D7	D6	D5	D4	D3	D2	D1	D0
Configurações gráficas	0	0	0	0	1	D	C	B

- D: Controle do display
 - D = 0: Display desligado | D = 1: Display ligado.
- C: Controle do cursor
 - C = 0: Cursor desligado | C = 1: Cursor ligado.
- B: Cursor piscante
 - B = 0: Cursor não piscante | B = 1: Cursor piscante.
- Normalmente: D = 1; C = 1; B = 0 -> (00001110)b = 0Eh

Configuração do *display*

- Antes da utilização, é necessário configurar o display de LCD de acordo com as suas características e/ou com o formato de exibição desejado através do envio de alguns comandos ao display;
- Quanto às características do cursor, podemos configurar:

Instrução	D7	D6	D5	D4	D3	D2	D1	D0
Deslocamento do Cursor	0	0	0	0	0	1	I/D	S

- I/D: Deslocamento do cursor
 - I/D = 0: direita para a esquerda (decremento);
 - I/D = 1: esquerda para a direita (incremento).
- S: Função scroll
 - S = 0: Scroll desligado;
 - S = 1: Scroll ligado.
- Normalmente: I/D = 1; S = 0 -> (00000110)b = 06h

Configuração do *display*

- Antes da utilização, é necessário configurar o display de LCD de acordo com as suas características e/ou com o formato de exibição desejado através do envio de alguns comandos ao display;
- Quanto às características do cursor, podemos configurar:

Instrução	D7	D6	D5	D4	D3	D2	D1	D0
Deslocamento do Cursor	0	0	0	0	0	1	I/D	S

- Outras configurações úteis:
 - 01h: limpa o display;
 - 02h: vai para a primeira posição do display.

- Exemplo de rotina para configuração do display:

```
CONFIG_LCD:  MOV LCD, #38h ; Define 8 bits, duas linhas e
               ; matriz de 5x7 pontos.
               LCALL WR_CMD ; Escreve comando.
               MOV LCD, #06H ; Deslocamento do cursor E->D.
               LCALL WR_CMD ; Escreve comando.
               MOV LCD, #0EH ; Cursor fixo.
               LCALL WR_CMD ; Escreve comando.
               MOV LCD, #01H ; Limpa o display.
               LCALL WR_CMD ; Escreve comando.
               MOV LCD, #02H ; Vai para a primeira linha
               ; (opcional).
               LCALL WR_CMD ; Escreve comando.
               RET
```

Endereçamento de linha / coluna

- A tabela a seguir mostra os endereços base de cada posição em um display de LCD 16x4, sendo que as duas primeiras linhas equivalem aos endereços de um display 16x2:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F

- Dessa forma, é possível acessar diretamente qualquer posição do display com base no endereço da posição desejada, sendo o endereço final dado por:
 - Endereço final = 80h + Endereço base
- Exemplo: posição dada pela linha 1 e coluna 7
 - Endereço final = 80h + 06h = 86h

Endereçamento de linha / coluna

- A tabela a seguir mostra os endereços base de cada posição em um display de LCD 16x4, sendo que as duas primeiras linhas equivalem aos endereços de um display 16x2:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F

- Dessa forma, é possível acessar diretamente qualquer posição do display com base no endereço da posição desejada, sendo o endereço final dado por:
 - Endereço final = 80h + Endereço base
- Exemplo: posição dada pela linha 2 e coluna 1
 - Endereço final = 80h + 40h = C0h

Endereçamento de linha / coluna

- A tabela a seguir mostra os endereços base de cada posição em um display de LCD 16x4, sendo que as duas primeiras linhas equivalem aos endereços de um display 16x2:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F

- Exemplo: função para “pular” para a segunda linha:

```

LINE_2:      MOV LCD, #0C0h      ; Move a posição desejada
                                   ; para o barramento.
              LCALL WR_CMD        ; Escreve um comando.
              RET                 ; Retorna da função.
  
```


- Utilizando a placa de desenvolvimento v0.7 e o arquivo exemplo disponíveis na página da disciplina, elabore um programa que escreva a seguinte mensagem estática em um display de LCD 16x2:

Disciplina de:

μControladores

- Requisito: utilize a diretiva DB para armazenar as mensagens e o DPTR para varrer a lista.

- Utilizando a placa de desenvolvimento v0.7 e o arquivo exemplo disponíveis na página da disciplina, modifique o programa para que a mensagem a seguir fique “rolando” (função scroll) no display indefinidamente:

Disciplina de μ Controladores / UFPEL 2023/1

Tabela ASCII

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]