Computação Digital

ENG 1448

Laboratório 07

Objetivos:

- Fazer o display LCD funcionar
- Se familiarizar com um novo padrão que requer uma sincronização

Descrição:

Você configurará o nosso kit para fazer o display LCD funcionar, LCD este que possui padronização similar a vários outros displays do mercado. Escreverá uma frase (única por grupo) composta por pelo menos 3 palavras, podendo ser: seu grupo, nome de alguém, qualquer coisa que faça sentido e que tenha sido intencional.

Implementação:

Antes de começarmos, saiba que esta parte é praticamente cópia do *User Guide* (Cap. 5), mas escrito com minhas palavras e em uma ordem que acho de mais fácil entendimento. Tenha liberdade de ler por aqui ou por lá.

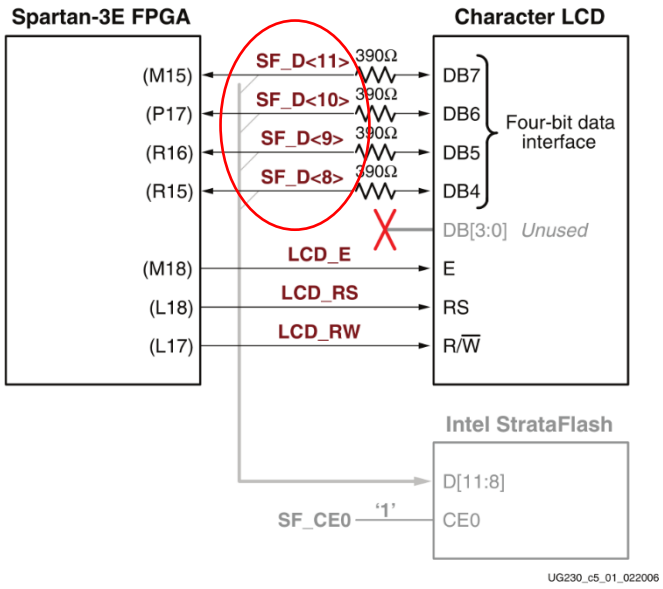
O LCD permite ser configurável com o barramento de dados de 8 ou 4 bits. A Digilent, fabricante do nosso kit, preferiu conectar apenas 4 terminais (“bits”) do LCD à FPGA. Desta forma, se quisermos enviar dados de 8 bits, somos forçados a enviar duas mensagens de 4 bits cada.

A segunda consideração importante é que, no nosso kit, o barramento de dados é compartilhado com a memória *Strata Flash*. Garanta que a *Strata Flash* não seja habilitada quando estiver interagindo com o LCD.

Na figura abaixo, observe tanto este compartilhamento, como o fato de só serem usados os 4 bits mais significativos do LCD, um leve complicador.

Inicialmente, vamos às memórias existentes no LCD. Existem 3 memórias, cada uma com uma função específica. São elas:

- DD RAM
- CG ROM
- CG RAM



DD RAM

A notação DD RAM vem de Display Data RAM, e é a RAM que armazena os dados a serem mostrados no display.

Esta memória possui posições equivalentes a um display de 2 linhas com 40 caracteres por linha (80 posições ao todo). Só que o display em si só reproduz 2 linhas de 16 caracteres cada. O motivo de possuir 40 caracteres por linha é que o display LCD pode rolar a “janela de exibição” para a esquerda ou para a direita para revelar os dados adicionais armazenados na DD RAM.

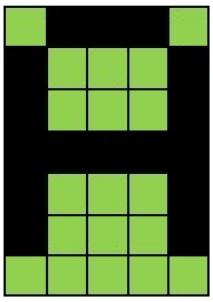
	Character Display Addresses																Undisplayed Addresses		
1	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	…	27
2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	…	67
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	…	40

Figure 5-3: **DD RAM Hexadecimal Addresses (No Display Shifting)**

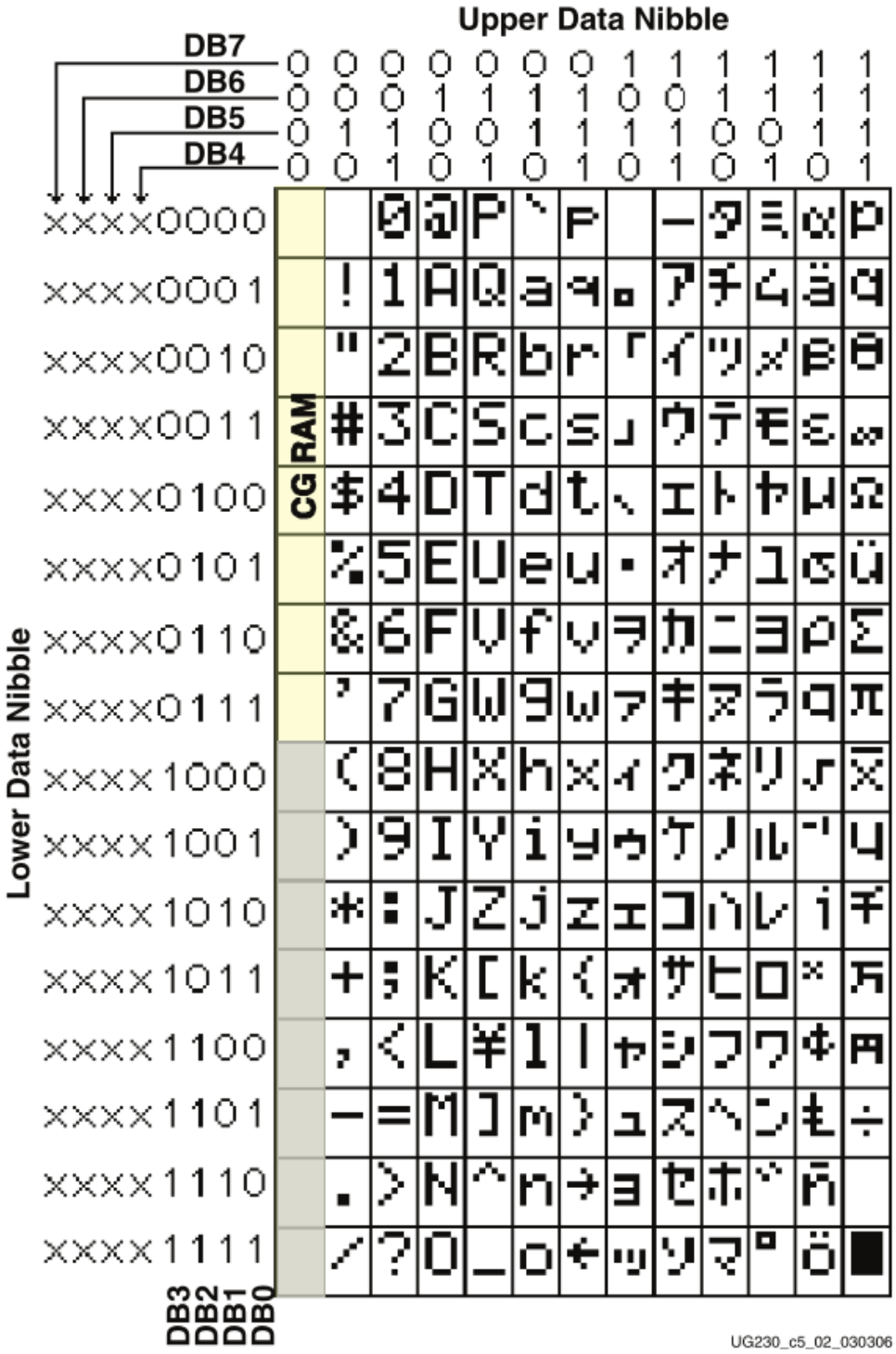
CG ROM

A CG ROM vem de Character Generator ROM, ou seja, é uma memória ROM que traz já o código de cada caractere que será traduzido no desenho gráfico desejado, através da sensibilização dos pontinhos (pixels) que formam um caractere. Apenas para ilustrar, os pontinhos que formam a letra “A” maiúscula precisam estar memorizados em uma ROM, de tal forma que o tal código relativo à “A” se traduza no seguinte desenho...

(Nota: no nosso display, há 8 linhas em cada caractere. Este desenho com 7 é só para exemplificar)



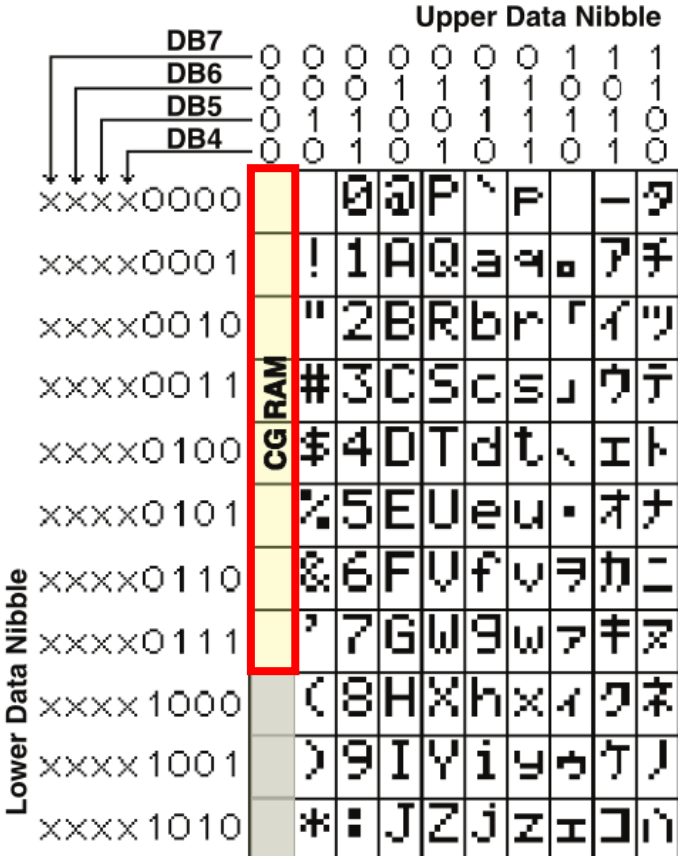
Assim, temos o seguinte mapa de caracteres para o nosso LCD específico:



CG RAM

A CG RAM (Character Generator RAM), é a memória RAM de 8 posições, onde se pode definir caracteres customizados, como, por exemplo:

É possível gravar 8 caracteres personalizados, como pode ser visto na figura abaixo:



Agora vamos aos controles do LCD. Há 3 pinos importantes, o LCD_E, LCD_RW e o LCD_RS.

- ✓ O LCD_E é o enable (E) do LCD. Quando em “0”, todas as entradas do LCD são ignoradas. Para conversar com o LCD, é primordial que o LCD_E esteja em “1”. A grande dica é usar este controle para a correta temporização.
- ✓ O LCD_RW quando em “0” indica ao LCD que será uma operação de escrita, do contrário (“1”), de leitura.
- ✓ O LCD_RS quando em “1” indica que o comando é de manipulação de dados, do contrário (“0”), comando ou endereçamento.

É recomendável que vocês leiam o que cada um dos comandos abaixo faz (a descrição de cada um dos comando apresentados na figura abaixo estão nas páginas 48-52 do *User Guide*).

Table 5-3: **LCD Character Display Command Set**

Function	LCD_RS	LCD_RW	Upper Nibble				Lower Nibble			
			DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Clear Display	0	0	0	0	0	0	0	0	0	1
Return Cursor Home	0	0	0	0	0	0	0	0	1	-
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S
Display On/Off	0	0	0	0	0	0	1	D	C	B
Cursor and Display Shift	0	0	0	0	0	1	S/C	R/L	-	-
Function Set	0	0	0	0	1	0	1	0	-	-
Set CG RAM Address	0	0	0	1	A5	A4	A3	A2	A1	A0
Set DD RAM Address	0	0	1	A6	A5	A4	A3	A2	A1	A0
Read Busy Flag and Address	0	1	BF	A6	A5	A4	A3	A2	A1	A0
Write Data to CG RAM or DD RAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0
Read Data from CG RAM or DD RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0

A maneira formal de se conversar com o LCD é através de 3 etapas:

- ✓ Inicialização
- ✓ Configuração
- ✓ Escrita

Antes de iniciar o projeto, recomendo a leitura do modo de operação de 4 bits da interface do LCD, páginas 52 e 53 do manual do kit.

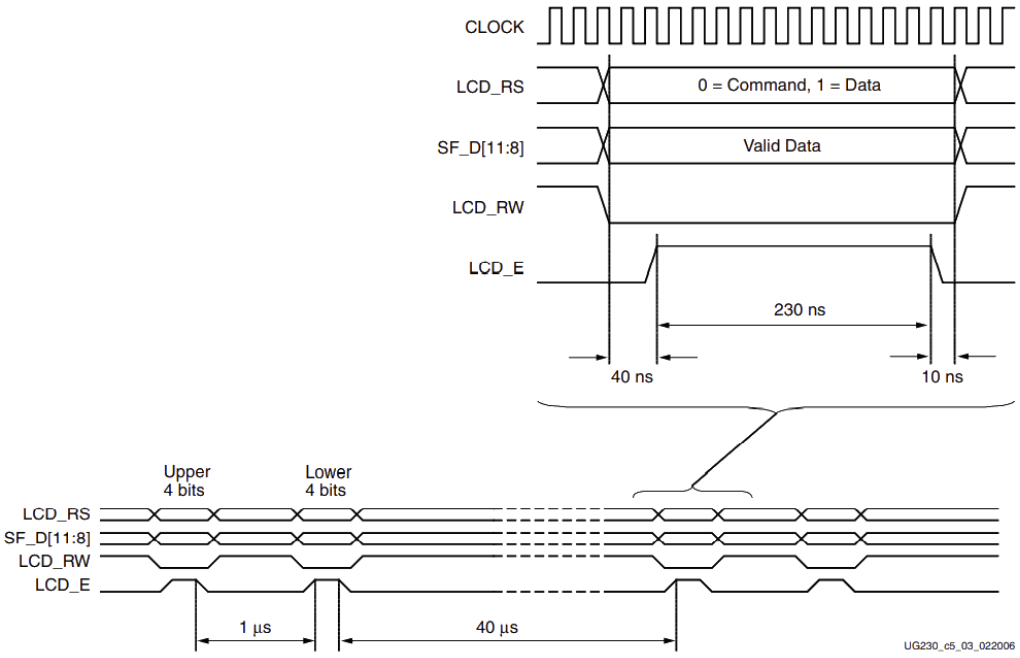


Figure 5-6: **Character LCD Interface Timing**

Resumidamente, temos:

Inicialização

Nesta etapa, devemos realizar a seguinte sequência.

1. Wait 15 ms or longer, although the display is generally ready when the FPGA finishes configuration. The 15 ms interval is 750,000 clock cycles at 50 MHz.

2. Write SF_D<11:8> = 0x3, pulse LCD_E High for 12 clock cycles.

3. Wait 4.1 ms or longer, which is 205,000 clock cycles at 50 MHz.

4. Write SF_D<11:8> = 0x3, pulse LCD_E High for 12 clock cycles.

5. Wait 100 µs or longer, which is 5,000 clock cycles at 50 MHz.

6. Write SF_D<11:8> = 0x3, pulse LCD_E High for 12 clock cycles.

7. Wait 40 µs or longer, which is 2,000 clock cycles at 50 MHz.

8. Write SF_D<11:8> = 0x2, pulse LCD_E High for 12 clock cycles.

9. Wait 40 µs or longer, which is 2,000 clock cycles at 50 MHz.

Traduzindo, temos...

Passo	Dado	Enable	Ciclos de clock
0			750.000
1	D=3h	E=1	12
2		E=0	205.000
3	D=3h	E=1	12
4		E=0	5.000
5	D=3h	E=1	12
6		E=0	2.000
7	D=2h	E=1	12
8		E=0	2.000

Configuração

Para a configuração, a partir de certas escolhas, como se o cursor incrementa automaticamente, se ele fica piscando ou não etc., temos:

1. Emitir um comando Funcion Set com 0x28 (outra notificação para 28h) para a operação no nosso Kit
2. Emitir um comando Entry Mode Set com 0x06, para que o cursor automaticamente seja incrementado

3. Emitir um comando Display On/Off com 0x0C, tanto para tornar o LCD visível como para desabilitar a visualização do cursor, bem como o pisca-pisca
4. Emitir um comando Clear Display. Segundo o manual, “Allow at least 1.64 ms (82,000 clock cycles) after issuing this command”

Escrita

Esta etapa claramente é a parte mais fácil, mas também sensível em timing.

Uma vez o endereço inicial estipulado (com Set DD RAM address), basta ir escrevendo os caracteres com Write Data. Lembre só do “timing”.

Relatório

Descreva detalhadamente cada etapa de inicialização, configuração e escrita no display LCD. Faça um testbench e detalhe o envio dos dados, assim como o timing.

Dica:

Faça antes um roteiro da ordem e dos dados que vocês precisam mandar para o LCD, e os tempos mínimos aceitos. Provavelmente para grande parte dos timings, você poderá ter tempos maiores que já resolvem a temporização de vários comandos – e você ganhará assim uma padronização mínima.