# Lab 7: Character LCD Control

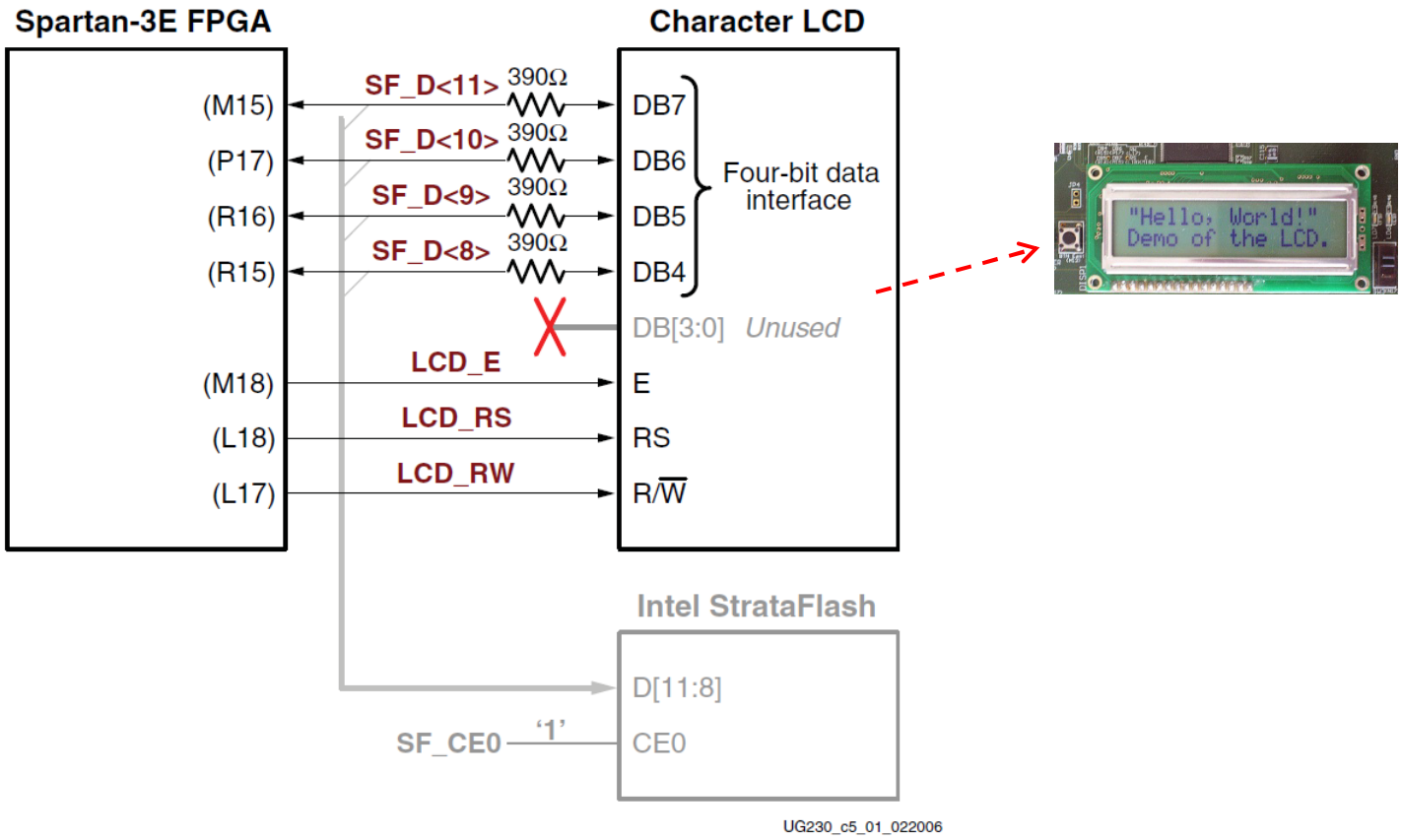

National Chiao Tung University
Chun-Jen Tsai
11/18/2016

# Lab 7: Character LCD Control

- In this lab, you will learn how to use the 1602 character LCD module
  - When you circuit is configured into the FPGA, it will compute the first 25 Fibonacci numbers and store them in SRAM
  - When the computation is done, the LCD display will begin to scroll through the numbers cyclically
  - Then, the user can use the WEST buttons to change the direction of scrolling
- You will demo the design to your TA during the lab hours on 11/29

# 1602 Character LCD Display

- There is a 2×16 character LCD display on the board:



# Memory Map of the LCD

- The LCD device can be treated as a 32-byte memory
  - Each 8-bit cell corresponds to a character on the display
  - Writing an ASCII code to a cell will display the character on the corresponding location on the LCD:


Note: the LCD is slow, you should not update the screen faster than every 0.5 seconds!

- Character display memory (DD RAM) addresses:

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F |
| 2 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 4A | 4B | 4C | 4D | 4E | 4F |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |

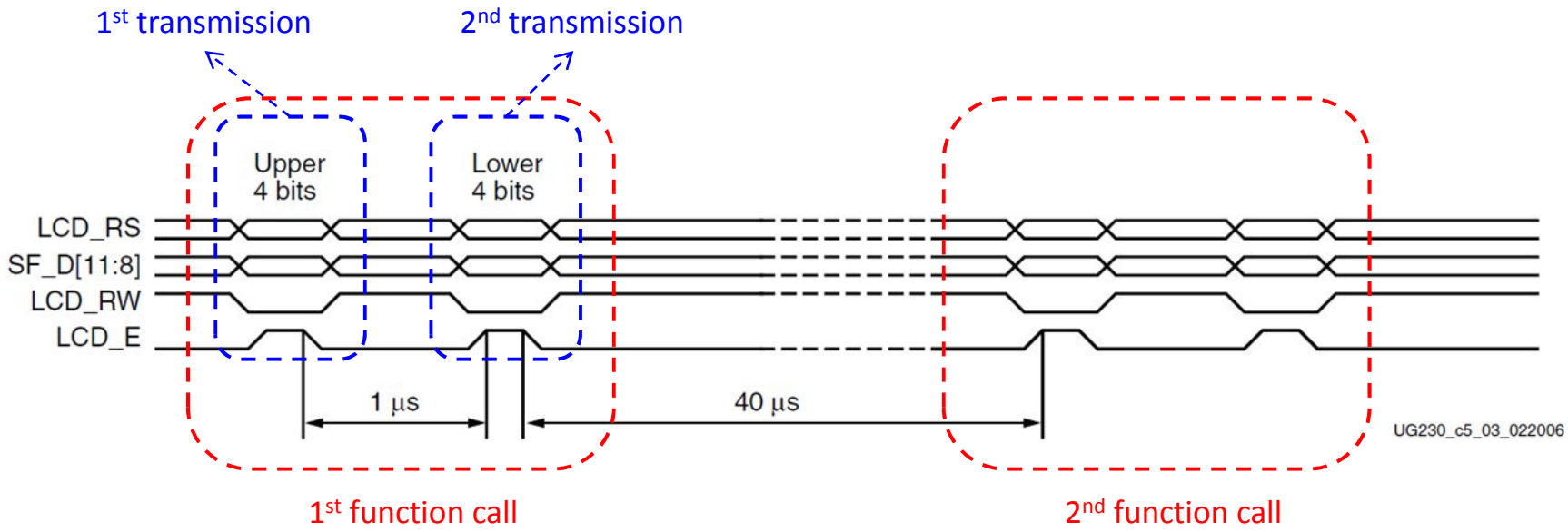
# Character LCD Interface (1/2)

- The LCD interface has 8 data wires (DB0 ~ DB7) and 3 control wires (LCD_E, LCD_RS, LCD_RW):
  - LCD_E enable/disable the inputs to the LCD module
  - The rest of the wires are defined depending on the functions:

| 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 |

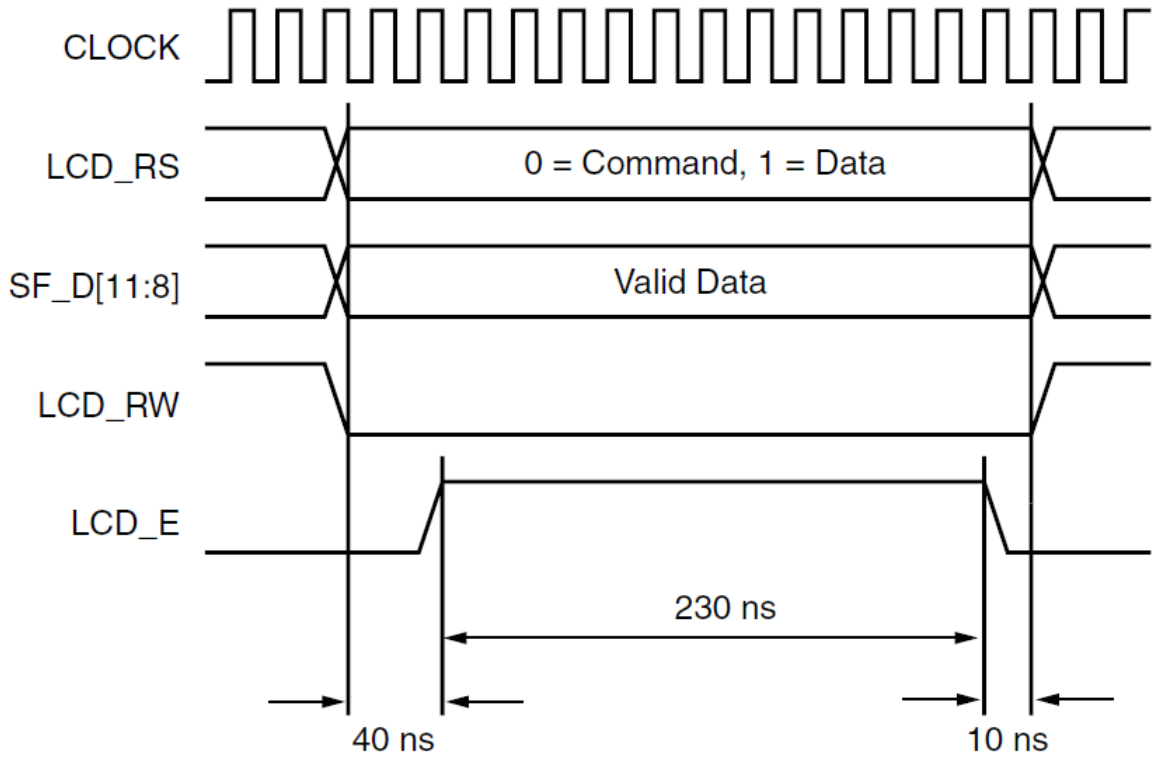
# Character LCD Interface (2/2)

- However, the Spartan-3E Starter Board uses the 4-bit operating mode of the LCD device, that is, only DB4~DB7 are connected to the FPGA
  - Execution of a function will need two transmissions, using only LCD_E, LCD_RS, LCD_RW, and DB4~DB7:



# Timing Diagrams for Transmission

- The timing diagram for one transmission in four-bit mode is as follows:
  - Note that execution of a function requires two transmissions



# The Sample LCD Module

- A ISE workspace that contains an LCD module that handle all the timing issues will be provided for you to use; the workspace also has a “Hello, World!” circuit that shows you how to use the LCD module

```
module LCD_module(
    input clk,
    input reset,
    input [127:0] row_A,
    input [127:0] row_B,
    output reg LCD_E,
    output reg LCD_RS,     //register select
    output reg LCD_RW,     //read / write
    output reg [3:0]LCD_D //data
    );
```

top-row of text

bottom-row of text

# Two Phases of the LCD Controller

- The LCD controller is composed of two FSMs:
  - The FSM of the initialization phase has 12 states and runs at 23.84 Hz; once the initialization is done, it will trigger the text screen refreshing FSM
  - The text screen refreshing FSM has 68 states and runs at 381.47 Hz; Among the 68 states, 64 states are used to send ASCII codes (4-bit per state) and 4 states are used to set the cursor positions
- The LCD screen will be refreshed at 5.96 Hz since 381.47Hz / 68 = 5.6 Hz

# Lab 7 Sample Package

- For this lab, the sample circuit shows the following text on the LCD module:


- When the WEST button is pressed, the text lines will be flipped

# The Fibonacci Numbers

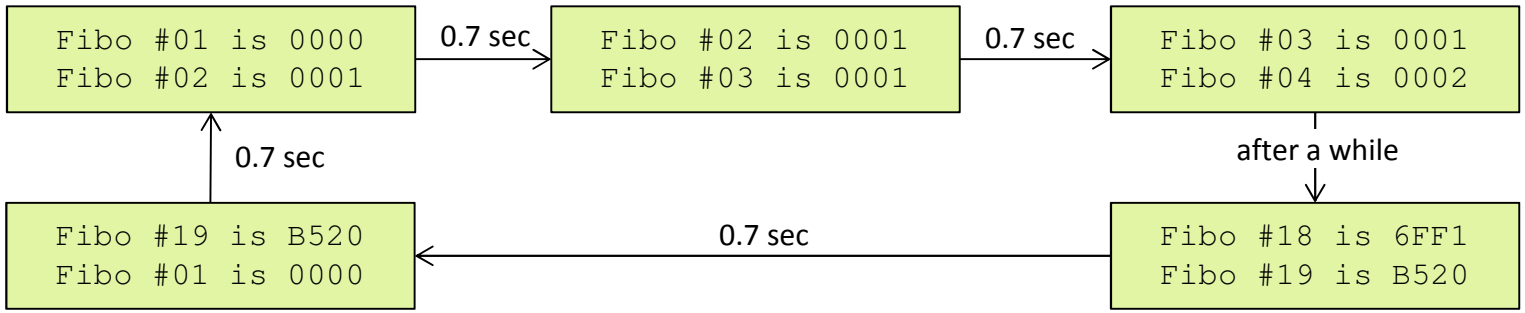
- The first two Fibonacci numbers are 0 and 1. Each remaining number is the addition of the previous two numbers.
- A short C-model that computes the first 25 Fibonacci numbers is as follows:

```
int fibo[25], idx;

fibo[0] = 0, fibo[1] = 1;
for (idx = 0; idx < 25; idx++)
{
    if (idx >= 2)
    {
        fibo[idx] = fibo[idx-1] + fibo[idx-2];
    }
    printf("Fibo #%02x is %04x.\n", idx+1, fibo[idx]);
}
```

# What to Do in Lab7

- In Lab 7, it is mandatory for you to do following things:
  - Use a 16-bit SRAM to store the array `fibo[25]`
  - Once the first 25 Fibonacci numbers are found, the LCD will start to display these numbers as follows:
    - Roughly every 0.7 sec, the LCD scrolls up one number
    - If the WEST button is pressed, the scrolling direction will be reversed (scroll-up becomes scroll-down, and vice versa)
  - Example display: cyclic scroll-up (numbers are in hexadecimal)



# References

- Chapter 5 of *Spartan-3E Starter Kit Board User Guide, UG230 (v1.0)*.
- Xilinx, *Using Block RAM in Spartan-3 FPGAs, Xilinx Application Note, XAPP 463 v1.1.2*, July 23, 2003.