Lab #7: Design Example: Keypad Scanner and Encoder - Part 2 (120 pts)

Objective

The objective of lab assignments 6 through 10 are to systematically design and implement an FPGA-based keypad scanner. The keypad scanner will utilize a FIFO for data storage and retrieval, a display mux, and the seven-segment displays, slide switches, and LEDs of the Digilab XLA prototyping board.

The top level block diagram of the system is shown in Figure 1 and the I/O in Table 1. When a button of the hex keypad is pressed the system must decode the button and store the data in an internal FIFO. The *read* button will be used to read data from the FIFO and display the data on the seven-segment displays. The *mode_toggle* input will be used to toggle between display states so that more than eight signals can be presented for view on the LED's. The LEDs will display the status of the FIFO and other information. The hardware prototype will be verified to operate with the Grayhill 072 hex Keypad. A system partition is shown in Figure 2. As input *mode_toggle* is depressed output *DGrp* will toggle between DGrp1, DGrp2, and DGrp3 as denoted in Table 2.

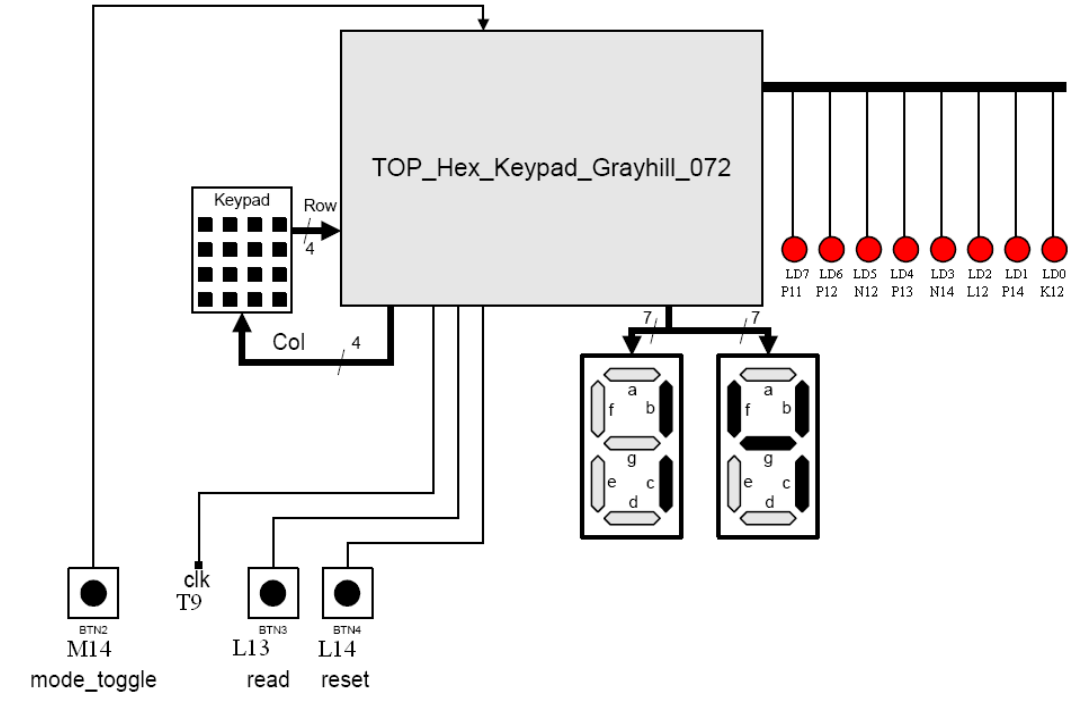

Figure 1: Keypad scanner top level

Name	Direction	Width	Purpose
clk	input		1MHz clock
reset	input		Asynchronous active high reset
read	input		Read from the fifo.

mode_toggle	input		Toggle between LED display states
Row	input	[3:0]	Keypad row
Col	output	[3:0]	Keypad column
seven_seg	output	[6:0]	7-segment character - active low
seven_seg_0_en	output		7-segment character 0 enable – active low
seven_seg_1_en	output		7-segment character 1 enable – active low
seven_seg_2_en	output		7-segment character 2 enable – active low
seven_seg_3_en	output		7-segment character 3 enable – active low
DGrp	output	[7:0]	8 LED's

Table 1: Keypad scanner I/O

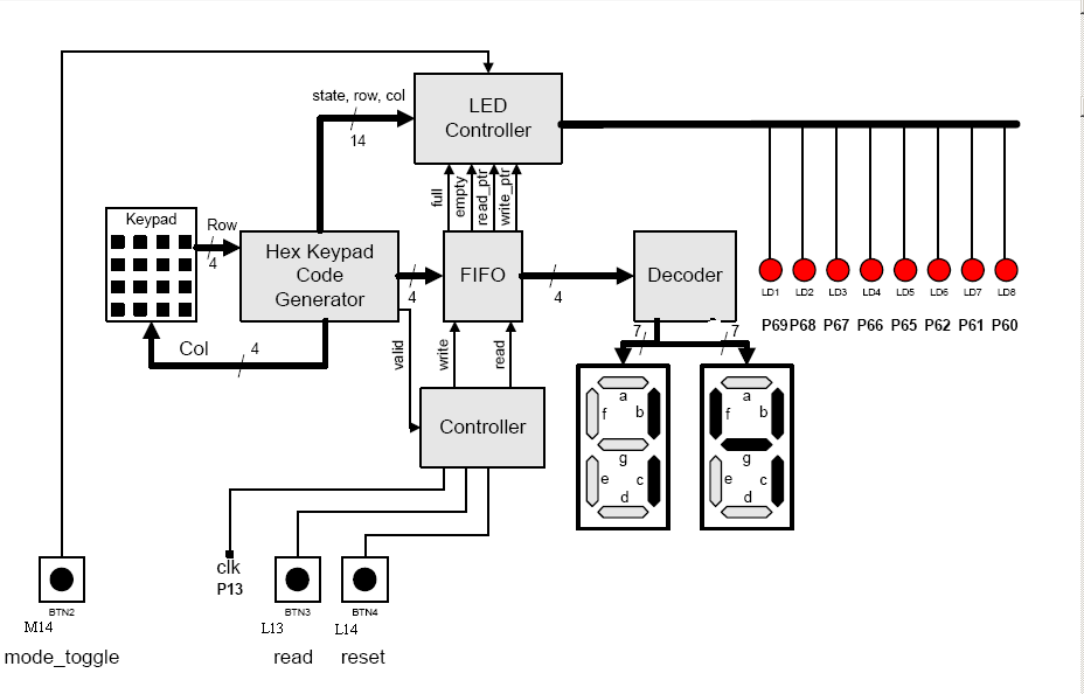

Figure 2: Partition of keypad scanner

The objective of this lab is to design and test the FIFO and decoder units.

Data Group	**System Debug Info**
DGrp1	{1'b0, *read_ptr*[2:0], 1'b0, *write_ptr*[2:0]
DGrp2	{*<your choice>, empty, full*}
DGrp3	{*Row, Col*}

Table 2: LED System Debug info

FIFO

The text, *Advanced Digital Design with the Verilog HDL*, presents a parameterized Verilog model of a FIFO. A FIFO (first-in-first-out) buffer is a dedicated memory stack consisting of a fixed array of registers. The FIFO that is to be used in this lab is shown in Figure 3 with stack_width=4 and stack_height=8. The registers of the stack operate synchronously (rising edge) with a common clock, subject to reset. The stack has two pointers (addresses), one pointing to the next word to which data will be written, and another pointing to the next word that will be read, subject to write and read inputs, respectively. The FIFO has input and output data-paths, and two bit-lines serving as flags to denote the status of the stack (full or empty). Output *Data_out* should only change if input *read_from_stack* is asserted and is not simply the data pointed to by the read pointer. See the text for additional details about the FIFO.

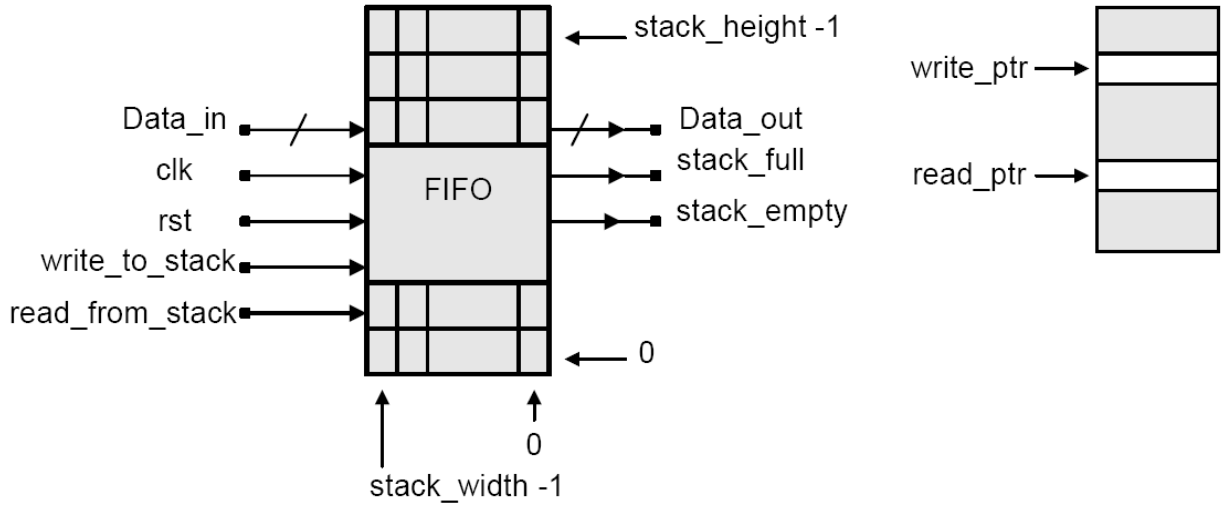

Figure 3: Fifo design

Decoder

A decoder must be designed to (1) decode a word read from the FIFO and (2) create driving signals for the two active-low, seven-segment displays on the Digilab board. The decoder must generate display signals in BCD for the 16 codes of the *Grayhill 072* keypad. Referring to Figure 2 the MSB of the 7-segment input is mapped to the "*a*" segment of the display and the LSB is mapped to the "*g*" segment of the string "abcdefg." Each seven segment display has an active low enable which must be driven. These are called seven_seg_0_en and seven_seg_1_en.

However, only one set of 7 display signals is available for the display and must be time multiplexed between the displays. See page 17 of the Spartan-3 Starter Kit Board Users guide for more details. These display signals are active low.

Testbench

You may verify the fifo/decoder together or separately. Be sure to fully test the fifo. Self-checking verification of the fifo is required.

Visual verification of the decoder is all that is required.

Deliverables

1. I/O table for the Fifo and Decoder
2. Source code of Fifo and Decoder
3. Test plan
4. Source code of final self-checking testbench
5. Demonstration of the operation of the test-bench to the lab instructor.

Grading

1. Demonstration - 60pts
    a. Correct operation of test-bench(s)
    b. Correct operation of Fifo/Decoder
2. Final report - 60pts