**EE 316 Computer Engineering Junior Lab**

**Design Project 4**

**Spring 2023**

**Specification**:  **Digital Hangman using FPGA**

**Due Dates: Thursday, March 23 - lab demo**

**Monday, March 25, written report due.**

**Parts List:**

1. one Digilent's Zedboard
2. one 2-line/16 character I2C LCD display module
3. one Sparkfun 7-segment display (with I2C)
4. one FTDI USB to UART 5V/3.3V adapter
5. one PS/2 keyboard
6. one PS/2 socket

Your task is to design an electronic system for playing hangman using a PS2 keyboard and an LCD panel connected to Digilent's Zedboard, which communicates with a remote PC via a USB to UART adapter. The PS/2 board will be connected to the Zedboard thoguth a PMOD port JA1. The LCD will use PMOD port JB1. Upon power-up, the LCD should display only a single cursor (an underline located at the left side of the screen). For the USB-UART serial port, use a minimum baud rate of 9600, parity none, data bits 8, and stop bits 1.

On the PC, the software should have a graphical user interface (GUI) that displays the game's status and an image of a hangman's noose. The software, written in C++, C#, Python, JAVA, or any other language, should read a text file containing a set of 5-10 words with variable lengths not exceeding 16 letters per word. The text file should be loaded at the beginning of the game.

At the start of each game, the software should display the message "New Game?" on the GUI and LCD module. The player must respond by pressing the "Y" key on the PS/2 keyboard connected to the hardware. The keyboard should recognize, at a minimum, the English alphabet from A through Z. Pressing any other key should have no effect. The software should randomly select one of the puzzle words from the text file and display the positions of each letter in the word using underscores "\_" symbols. On the LCD module, the word should be left-justified.

Once the puzzle is displayed, the player can guess a letter using the external PS/2 keyboard, with a maximum of 6 guesses allowed per puzzle. If the player guesses a letter correctly, the system should display every instance of the letter in the word. If the player guesses wrong, the "picture" on the GUI should change, and the number of remaining guesses should be displayed on the 7-segment display, connected to the PMOD port JA1.

If the player successfully guesses every letter in the word, the PC should send a congratulatory message to the LCD saying, "Well done! You have solved N puzzles out of M." If the player fails to guess the word, the message should be "Sorry! The correct word was XXXXX. You have solved N puzzles out of M." Since these messages exceed 16 characters, they should appear at the right edge of the LCD and scroll across the screen at a readable rate.

After displaying the result message, the system should display the message "New Game?" and the player can request a new puzzle by typing "Y," unless all the words from the text file have been used up. If the player wants to end the game, they can type the letter "N." Once the game is over, the LCD should display the final score ("4 correct out of 10") and then display the message "GAME OVER" after a short duration.

**Documentation:**

Digitent Zedboard

<https://digilent.com/shop/zedboard-zynq-7000-arm-fpga-soc-development-board/>

Help for PS/2 protocol

<http://www.burtonsys.com/ps2_chapweske.htm>

AT Keyboard Scan Codes

<https://webdocs.cs.ualberta.ca/~amaral/courses/329/labs/scancodes.html>

PS/2 Keyboard to ASCII Converter (VHDL)

<https://forum.digikey.com/t/ps-2-keyboard-to-ascii-converter-vhdl/12616>

VHDL Sample UART (make sure you have the permission to use this code)

<http://www.asic-world.com/examples/vhdl/uart.html>

Learn VHDL by example (search for Structural Modeling of HD-6402 on the page)

<http://esd.cs.ucr.edu/labs/tutorial/>

**Teams:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Team 1** | **Team 2** | **Team 3** | **Team 4** | **Team 5** | **Team 6** | **Writer** |
| Kubicka | LaBue | Morris | Skinner | Ernesto | Mathew |  |
| Shawn | Kaili | Isabelle | Graham | Pamela | Cameron |  |
| Zander | Keith | Ella | Jacob | Nelson | Nathan |  |
|  | Macy |  |  |  |  |  |