Pacman User's Guide

Mechanics of Pacman

User provides input through an NES controller. Input from the controller is used begin a new game and to navigate the Pac-man character through the map.

Assembly code is used in performing continual checks to the controller for user input. When a direction arrow is pressed on the direction pad (d-pad), code is then used to check if the asserted direction is a valid direction in which Pac-man can move. Invalid directions include those which point to walls. If the direction is determined valid then Pac-man will appear to move by 3 pixels at a time. He continues in the asserted, valid direction until a new, valid direction is asserted or he hits a wall.

Along with Pac-man are 4 other characters we refer to, generally, as ghosts. The ghosts also move around on the map. However, their movements do not depend on user input via the NES controller. Rather, the ghosts choose their next movement based on a random generator which chooses the next direction to assert. If the direction is valid, the ghost moves. Just as with Pac-man, however, if the asserted direction is a wall and thereby invalid, the ghost will continue in its previously asserted direction until the random direction generator asserts a new, valid direction or the ghost hits a wall.

VGA graphics are used to display the menu, map, score, lives, characters, and results of user input.

Running the Application

To run the application, we assume no changes have been made to either the original assembly file or its binary equivalent .bin file. If changes have been made you will need to reassemble the original .asm file using the assembler application. See Assembler User Guide for details.

Otherwise, we are ready to begin synthesis on the Spartan 3E FPGA using Xilinx 12.2. Before synthesizing ensure the exmem.v file reads from the bin file which contains the application code you would like to run. Initially you will want to read from "___MAIN_ASSEMBLY_CODE___.bin" Make sure this file is saved in the CR16 folder. This is where Xilinx will look for your .bin file. Although it is not recommended, if you happen to make changes to the application code, ensure the same file location convention is followed.

Once your files are up to date and in the right locations feel free to synthesize by right-clicking "Map Target Device" and selecting "run." This assumes your hardware is set up properly. See Spartan 3E User Manual for more information.