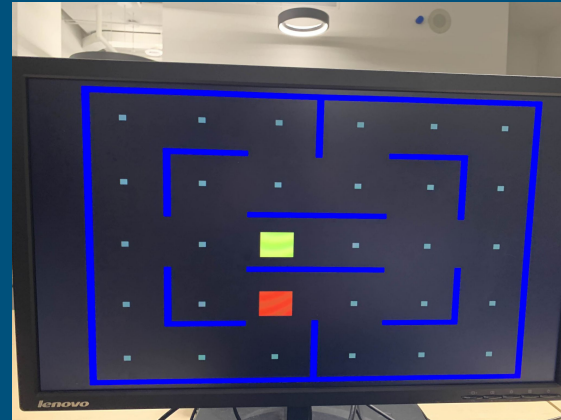


# FPGA-Based Board Pacman Game with Integrated AI

Minhyeok An 217078668  
Kevin Nguyen 217228255  
Prithviraj Keswani 215649817  
Vincent Mai 217990136  
Humza Inam 218144865

# Introduction

In this project, created a Pac-Man style game using VHDL on DE10 Lite boards and AI. The goal was to create an interactive game that combines real-time graphics, player controls, and smart AI behavior. Used a DE10 Lite board for the Pac-Man map and Pac-Man player and Raspberry Pico Pi 2W board for the ghost AI.



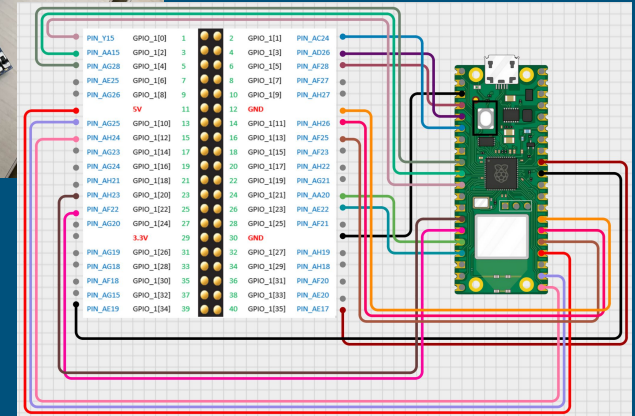
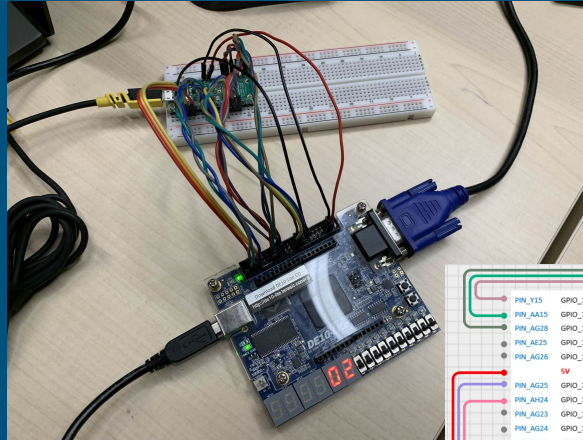
# Implementation Details

---

- The Pac-Man maze would be created by coding individual pixels.
- For the border and walls on the map we draw lines of blue pixels to create a maze for the Pac-Man and Ghost to roam around.
- The Pac-Man uses color yellow and Ghost uses color red.
- Pac-Man moves up/down using KEY0 and KEY1, and switches to left/right when SW0 is off.
- The Ghost AI will be controlled by Raspberry Pico Pi 2W.
- The Ghost will move based on the movement and coordinates of the player.
- Ghost movement will be based on the Euclidean Distance Algorithm.

# Hardware & Software Configuration

- DE10 Lite board
- Raspberry Pico Pi 2W board
- Bread board
- Jumper wires
- VGA wire
- Micro USB
- Quartus Prime
- MicroPython



# Code Explanation

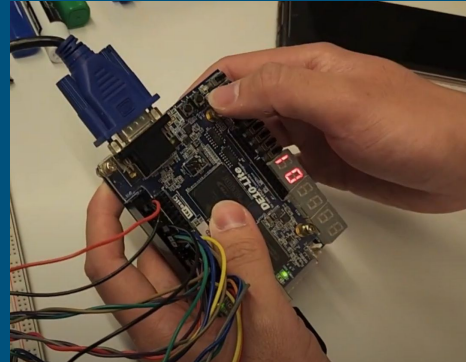
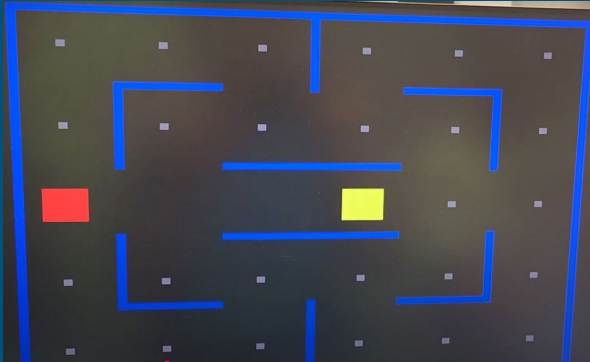
---

- image\_gen.vhd: handles image generation in VGA
- maze\_rom.vhd: defines the maze layout of the pacman map
- start\_screen.vhd: generates the start screen
- start\_screen\_pkg: contains the constants and parameters used in start\_screen
- vga\_controller.vhd: generates necessary timings for VGA display
- Wall\_mapping.vhd: contains the arrays and parameters needed for the walls
- Mapping\_pkg.vhd: contains the parameters for the map
- PACMANGHOST.py: handles the movement and AI of the ghost
- pacman\_top: top level which reroutes the process of the code
- pacman\_common: defines basic parameters for the pacman code

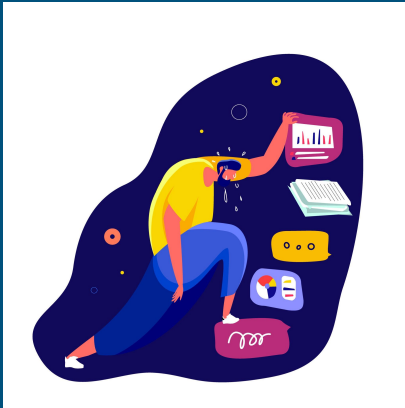
# Testing & Debugging

---

- User Testing was used to test wiring and the implementation of the code and display
- Gameplay Testing was used to configure the potential issues and mapping of the system



# Challenges



- Displaying the graphics on the VGA and finding the connection.
  - The movement of the pacman using 4 buttons.
  - Connecting the DE10 Lite board with the Raspberry Pico Pi board using GPIO.
-

# Concluding Remarks

---

- This project aimed to cover the important aspects taught in EECS 3216.
- Previous knowledge from EECS 3201 help us developed the FPGA VHDL code.
- New knowledge was required to implement the GPIO connections between the Raspberry Pico Pi 2W and the DE10 Lite FPGA board.
- We worked as a team together to solve these problems.





# Resources

---

- “VGA Controller (VHDL)”: <https://forum.digikey.com/t/vga-controller-vhdl/12794>