FLAPPY BIRD

VHDL FPGA

Brikena D, Roman H, Alex R, Peyton S

PRESENTATION OVERVIEW

BACKGROUND

Group motivation behind project

OVERVIEW

Top-level block diagram and overall functionality

BREAKDOWN

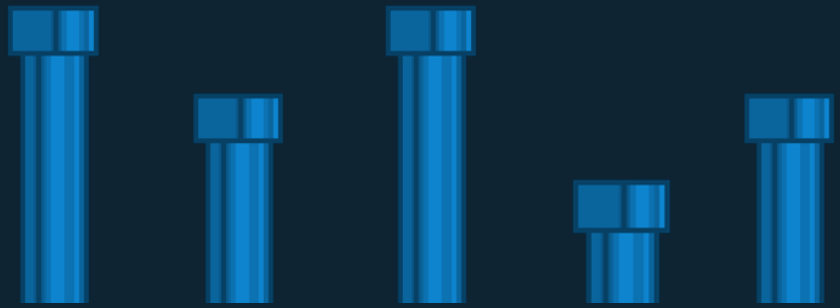
Individual program block analysis

PERFORMANCE

How well the design performs overall

LESSONS LEARNED

Takeaways from project and future improvements

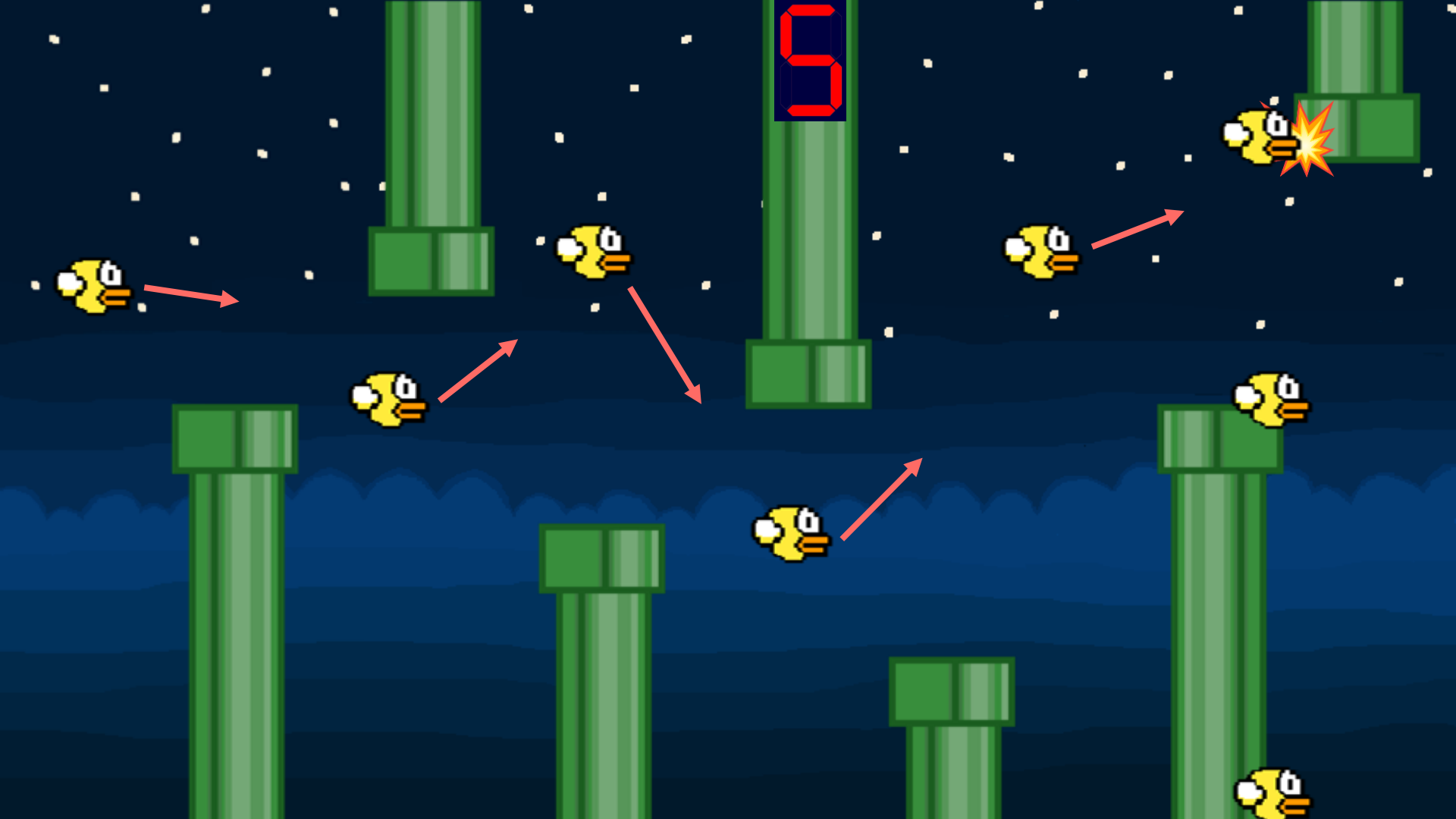


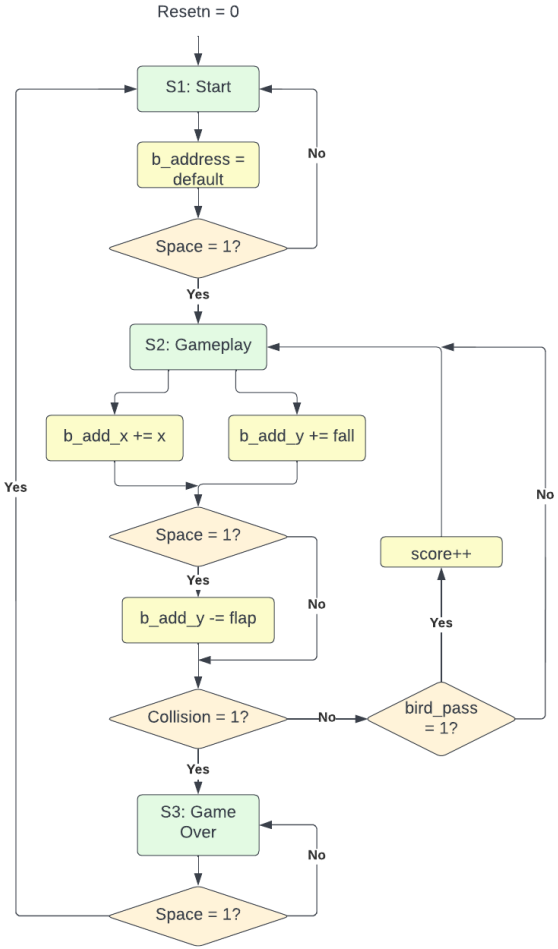
BACKGROUND

What motivated us to do this project?

- Common enjoyment of games
- VGA and PS/2 interfacing

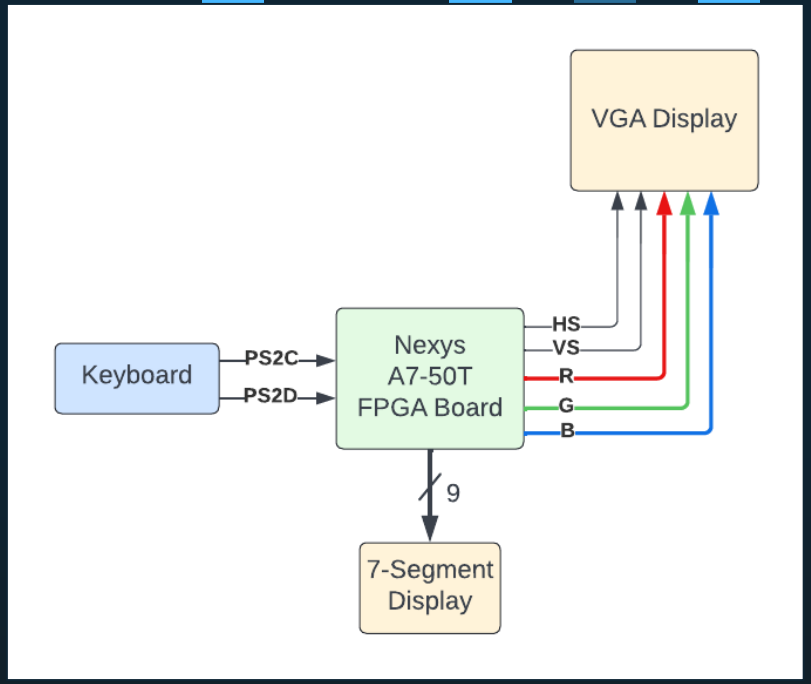
PROGRAM CONCEPT


5



PROGRAM FLOWCHART

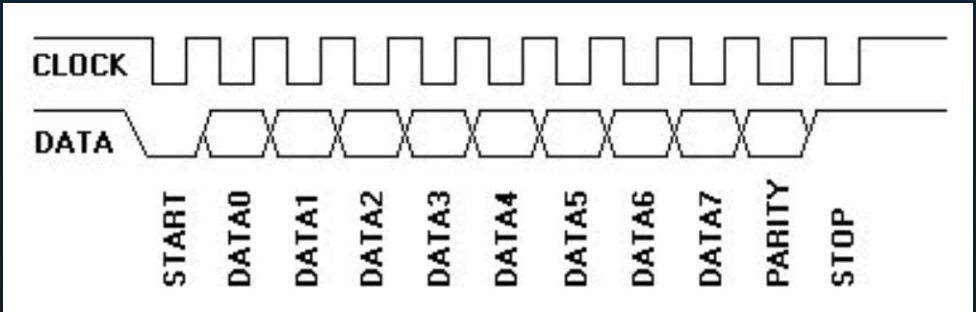
PERIPHERAL CONNECTIONS



PROGRAM BREAKDOWN

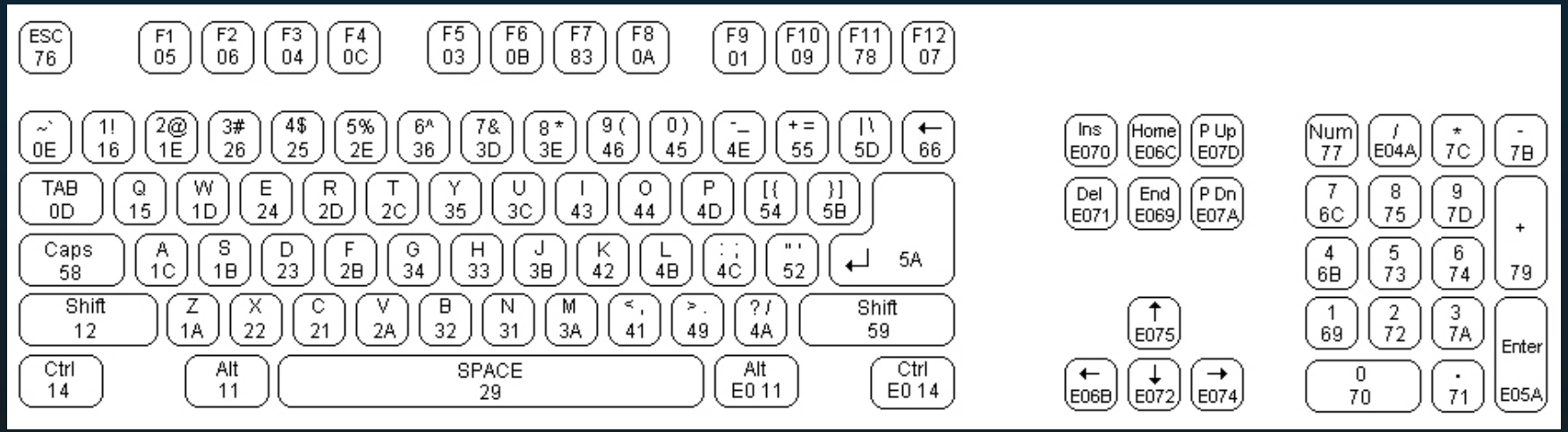
PS/2 PROTOCOL

- User game control
- Bidirectional synchronous serial protocol
- Data transmission = 11-12 bits
  - Start bit = ‘0’
  - 8 data bits
    - Little endian
  - Parity bit
    - Odd
  - Stop bit = ‘1’

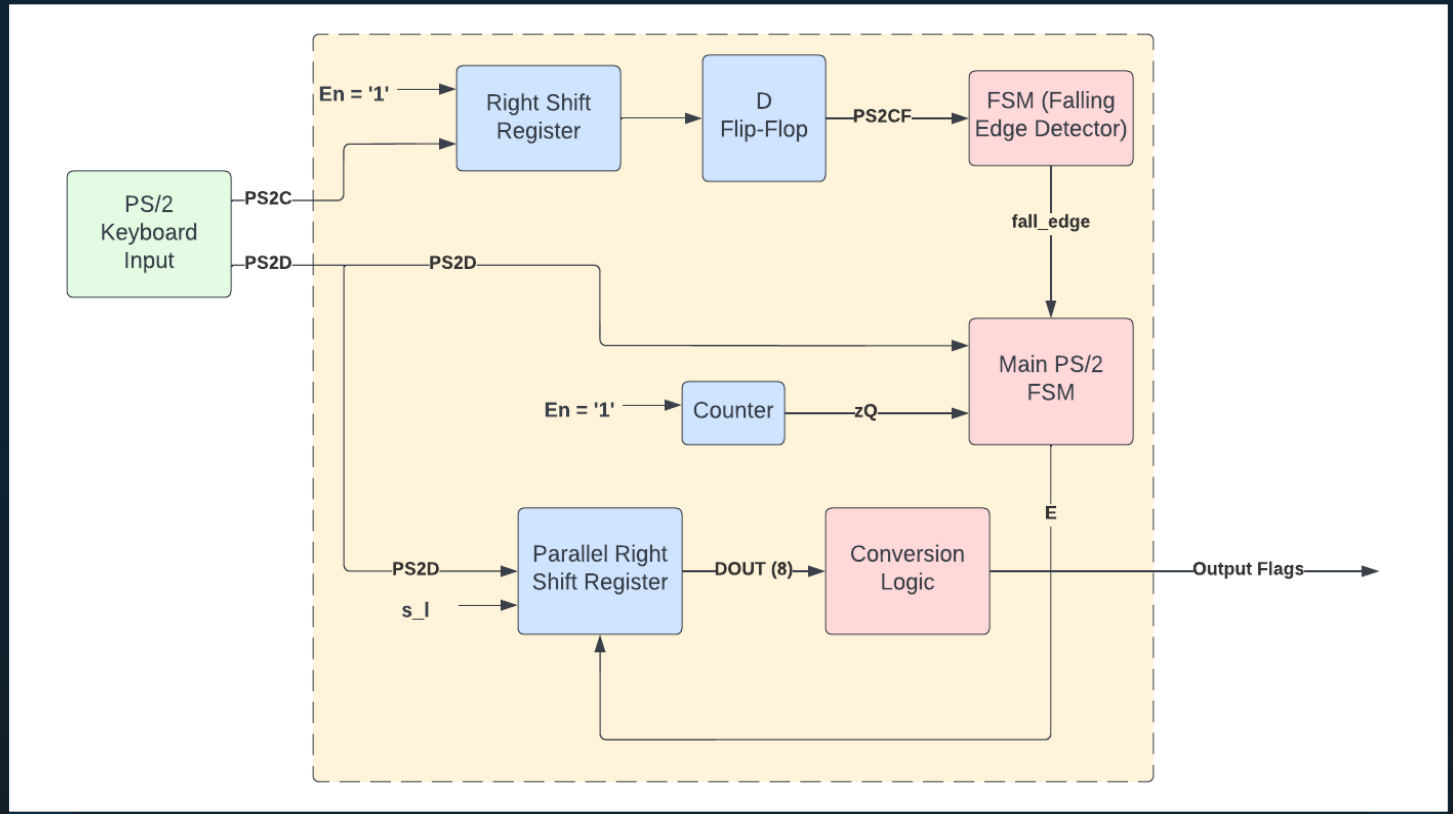


PS/2 PROTOCOL

- Strategy:
  - Decode ‘DATA’ signal
  - Set flags when desired keys pressed



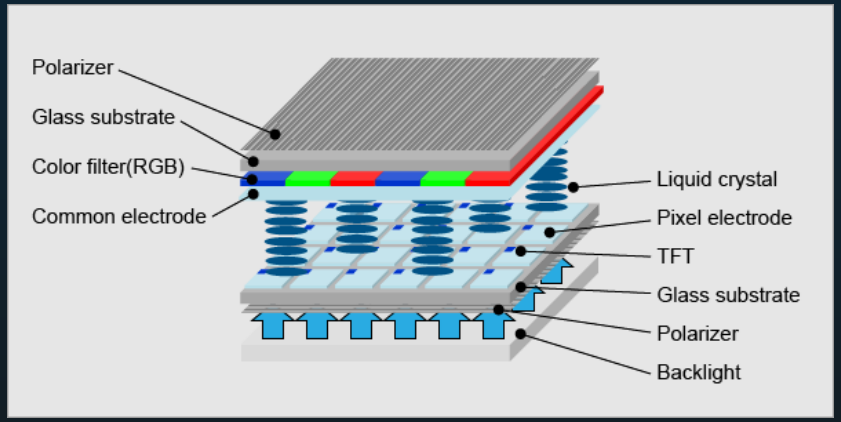
PS/2 PROTOCOL



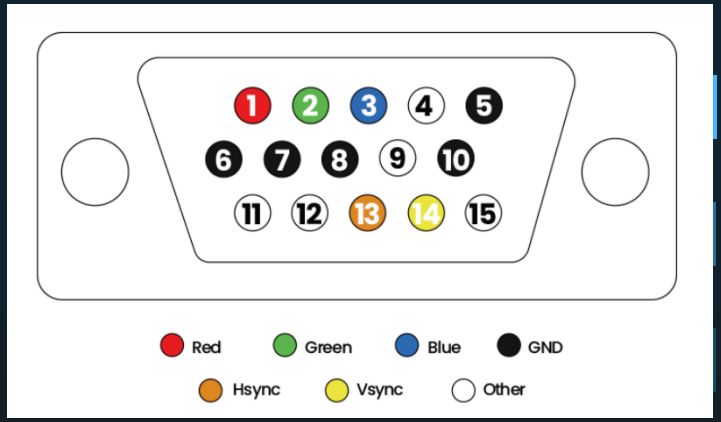
VGA DISPLAY PROTOCOL

Video Graphics Array

- 640 x 480 resolution (pixels)
- 256 color combinations
- 60 Hz refresh rate
- 15-pin connector



LCD Display

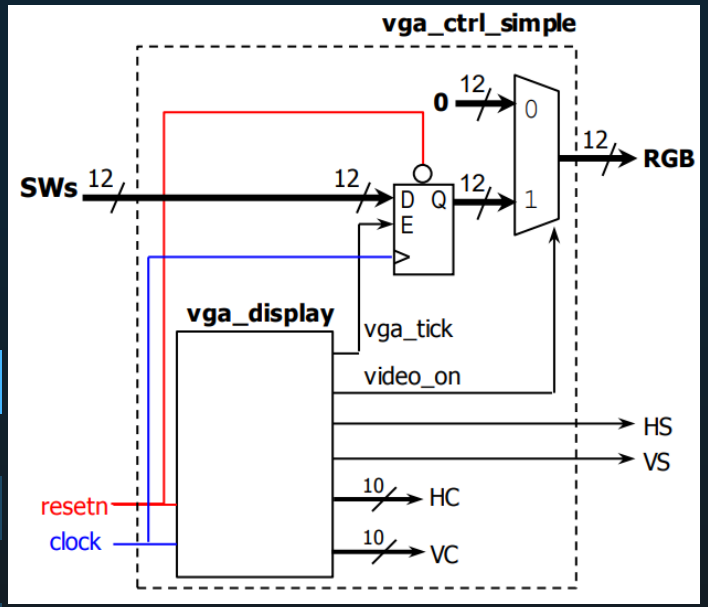


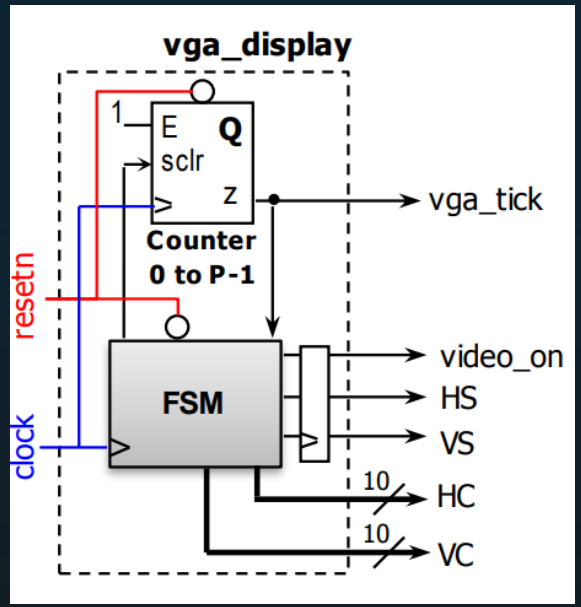
VGA Connector

VGA DISPLAY PROTOCOL

Video Graphics Array

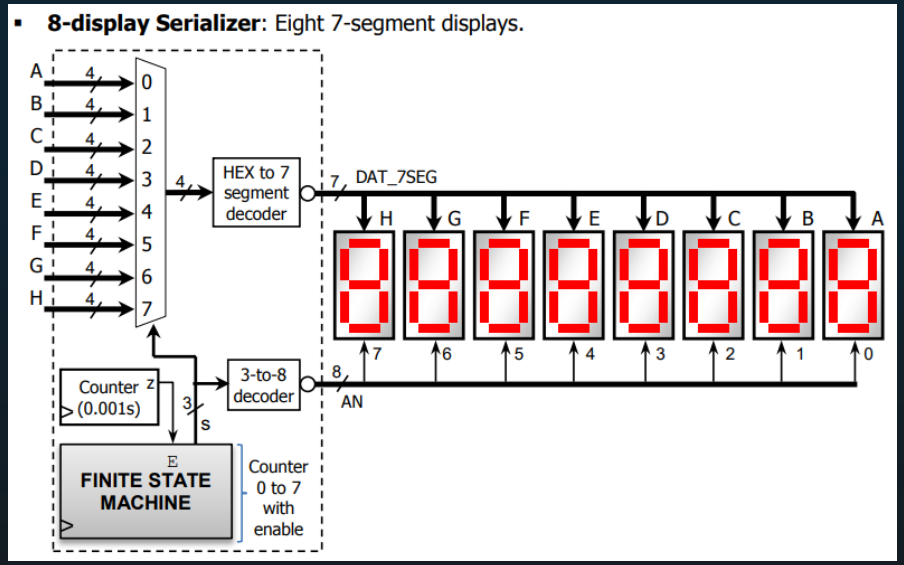
- Use HC,VC,HS, and VS to select pixel location
- RGB signal = color output for selected signal
- Use logic to dictate which pixel is displayed where

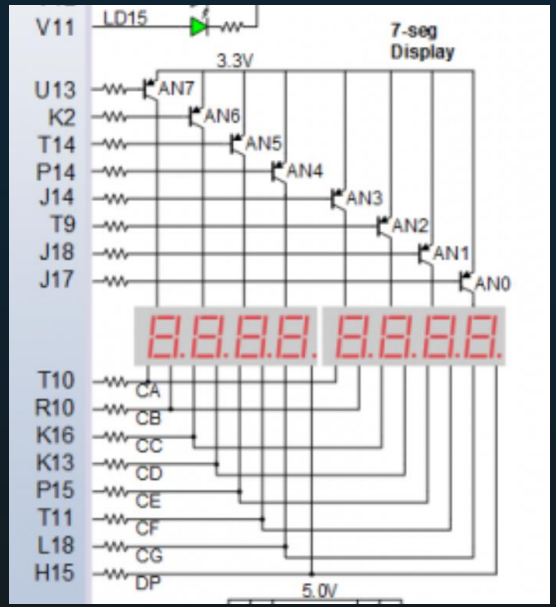




7-SEGMENT DISPLAY

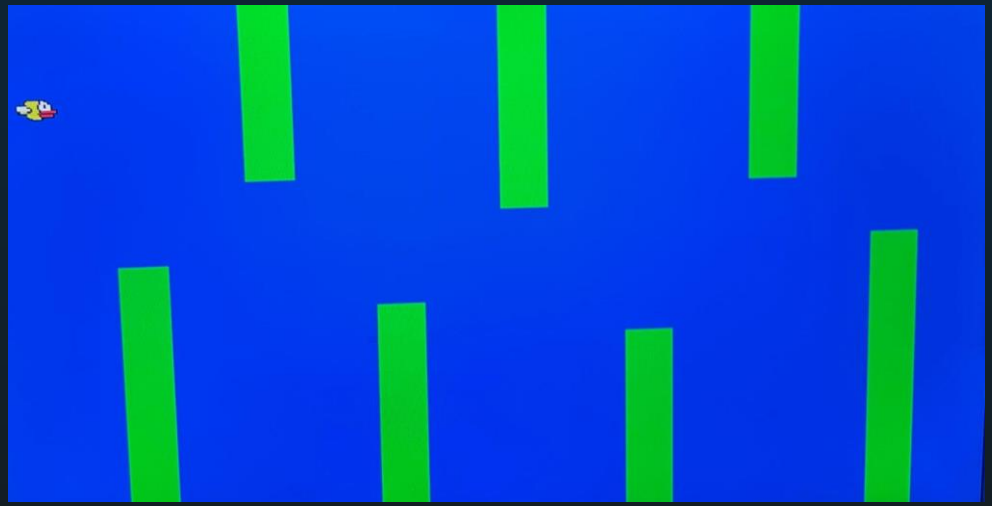
- On-board seven segment display
- Depicts game score
  - Score increments when bird passes obstacle





PROGRAM PERFORMANCE

- Quick response time
- Large sprite step
  - Simpler implementation
- Basic graphics
- Small playing field



PROGRAM DEMO

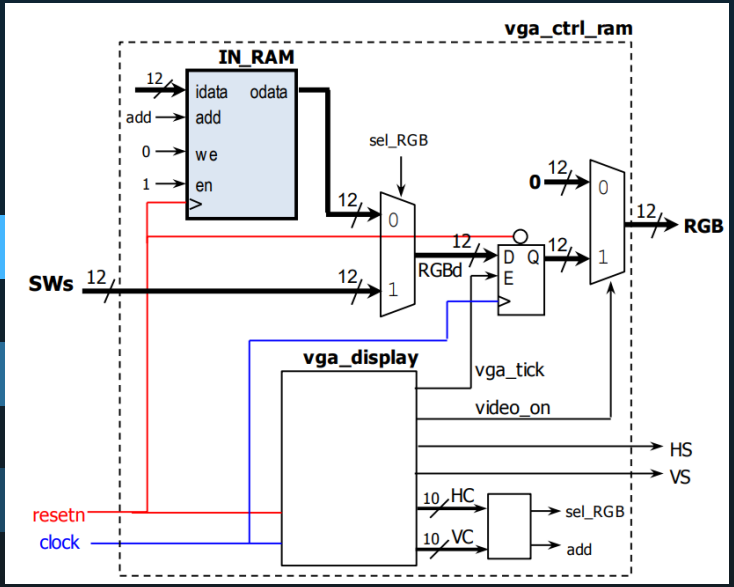
https://youtube.com/shorts/JBdozbKbibE

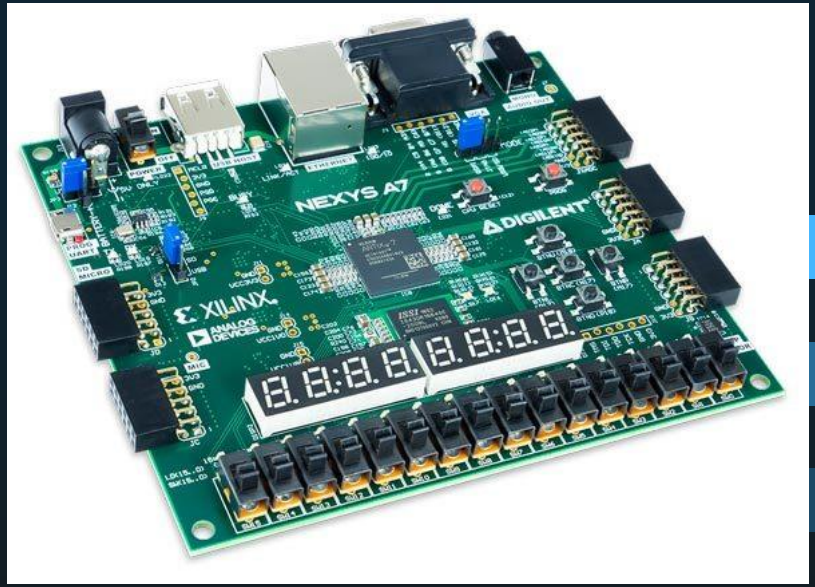
Another version of it:

https://streamable.com/6ecfd2

LESSONS LEARNED

- Initial plan: import graphics to FPGA
  - Nexys A7-50T RAM cannot handle extensive graphics
- Simple VGA control without using RAM has better performance





FUTURE IMPROVEMENTS

- Precise movement control of sprite
  - Bird movements confined to about 10 pixels either direction
- Variable level difficulty
- Use different board with larger RAM to have
  - Increase image quality
- Variable pipe size
  - Currently using pre-set pattern

THANK YOU!
ANY QUESTIONS?

We'd also like to give a special thank you to Professor Llamocca for all of his help

WORKS CITED

- U. Zoltán, "Nexys-A7-50T-OOB" *GitHub*, 2006. [Online]. Available: https://github.com/Digilent/Nexys-A7-50T-OOB/blob/master/src/hdl/Ps2Interface.vhd. [Accessed: 01-Apr-2022].
- D. Llamocca, "VHDL Coding for FPGAs," *Reconfigurable Computing Research Laboratory*. [Online]. Available: http://www.secs.oakland.edu/~llamocca/VHDLforFPGAs.html. [Accessed: 1-Apr-2022].
- S. K, "VGA Display Controller," *Digilent Reference*. [Online]. Available: https://digilent.com/reference/learn/programmable-logic/tutorials/vga-display-congroller/start. [Accessed: 01-Apr-2022].
- A. Brown, "Nexys A7 Reference Manual," *Digilent Reference*. [Online]. Available: https://digilent.com/reference/programmable-logic/nexys-a7/reference-manual. [Accessed: 1-Apr-2022].