Project Proposals

Project Proposal Details

- Due: Fri, Mar 6th, 11:59pm
  - One submission *per team* (not per person)
- Personal details: Your name, your partner's name, email addresses, etc.
- Your project: Project name, what you're making, why it connects to the course.
- Milestones: What will you do in Week #1, Week #2 & Week #3 (final demo)
  - Make sure each week is a lab's worth of work!

Project Partners FAQ

- Yes, you still need to work in pairs.
- No, it doesn't have to be the same person you did your labs with.
- Yes, it could be with somebody in another room or another section…
  - **BUT**…if any disagreement with your milestone marks arises, we need to verify this with the TA.
  - If you don't remember the TA who marked you or the TA doesn't know who you are, we can't guarantee a mark change.

Looking for inspiration?

- Think of electronic devices or simple games.
- Look at electronic hardware websites:
  - e.g. Creatron Inc.
- Remember, the project needs to be three labs' worth of work!
  - Project ideas that we will **not** allow:
    - Clocks / Stopwatches
    - Pianos
    - Tic-Tac-Toe (unless you add a smart AI)
    - Other ideas at this basic level.
  - Rejected ideas will have to be resubmitted, which results in less time to work on the first milestone.

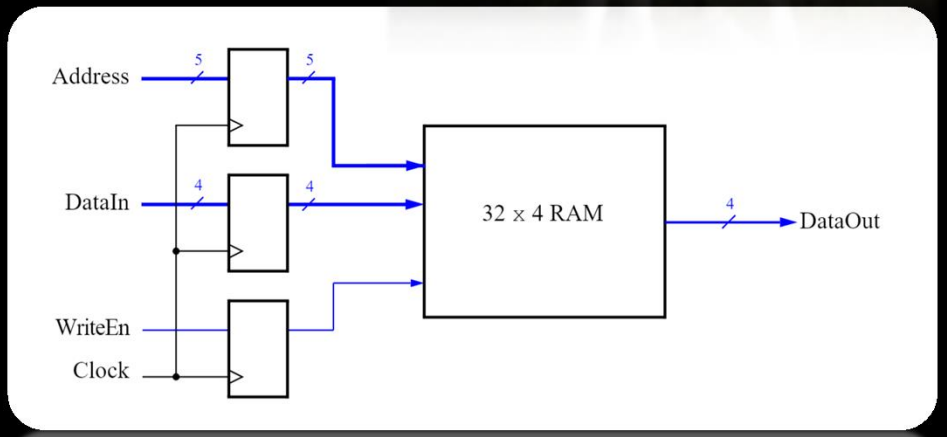
Lab 7 Preparation

Lab 7 Components

- Part I: Create a memory unit
- Part II: Interface with the VGA display
- Part III: VGA animation (bonus)

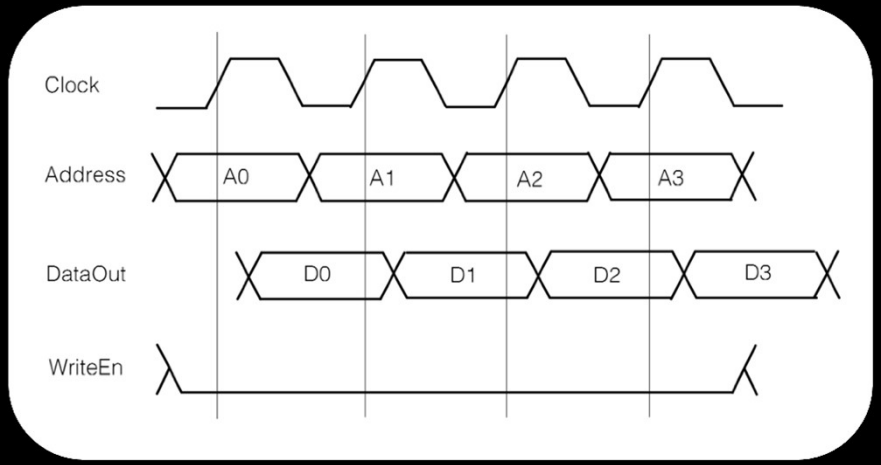
Part I: Memory Unit

- Creating a mini-RAM unit.
- Make use of the IP Catalog built into Quartus.
  - Follow lab instructions to create a 4-bit RAM unit with 32 words.
- Once created, connect this RAM to the switches, keys and HEX.

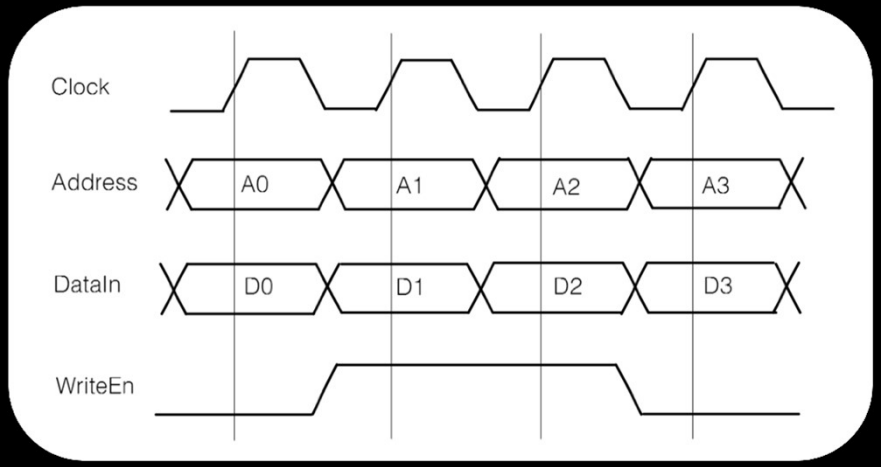


Part 1: Read & Write Timing

- Read:
  - Note slight delay after clock signal, before data appears.



- Write:
  - Note that only `D1` and `D2` are written (because of the `WriteEn` signal).

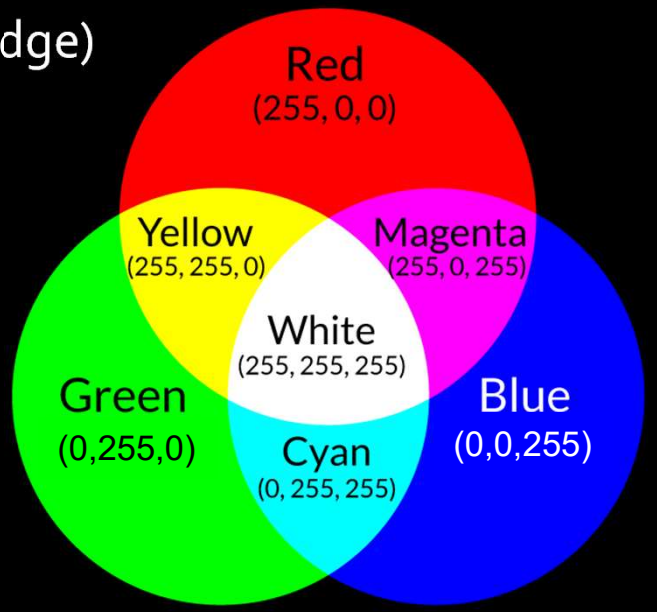


Part II: VGA Display

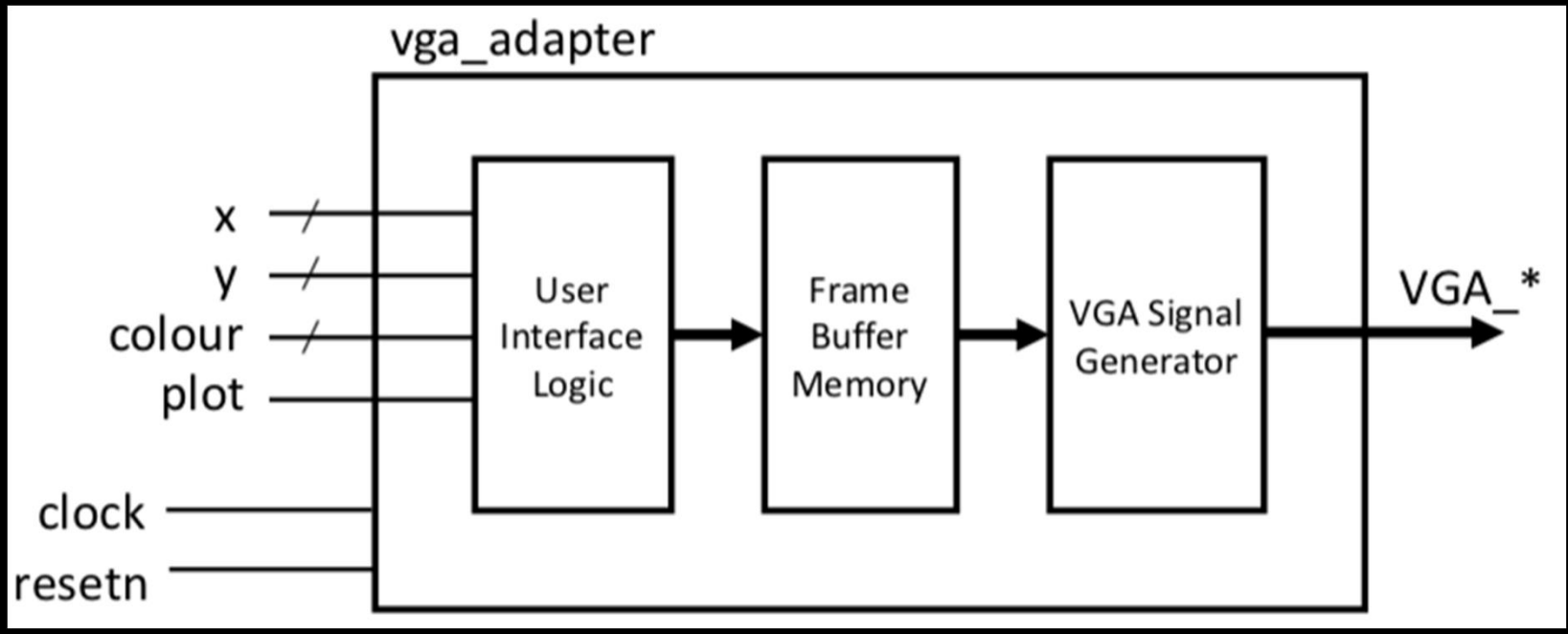
- Draw pixels on the screen, given a VGA adaptor that takes in the following values:
  - `X` (horizontal position of pixel)
  - `Y` (vertical position of pixel)

  } Where (`0`, `0`) is top left corner of screen

  - `colour` (three values: **R**, **G**, **B**)
  - `plot` (signals to write at next clock edge)
  - `clock`, `resetn`
- Colours are additive!

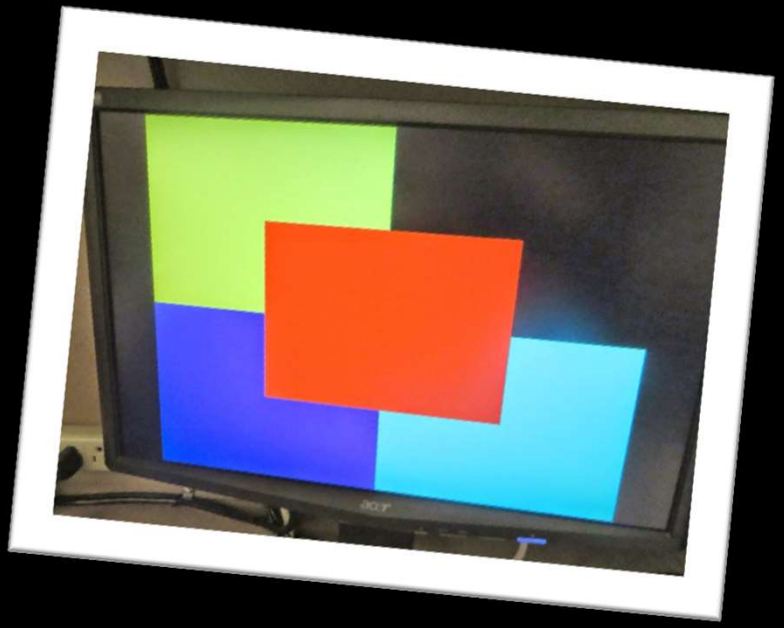


VGA adapter module schematic



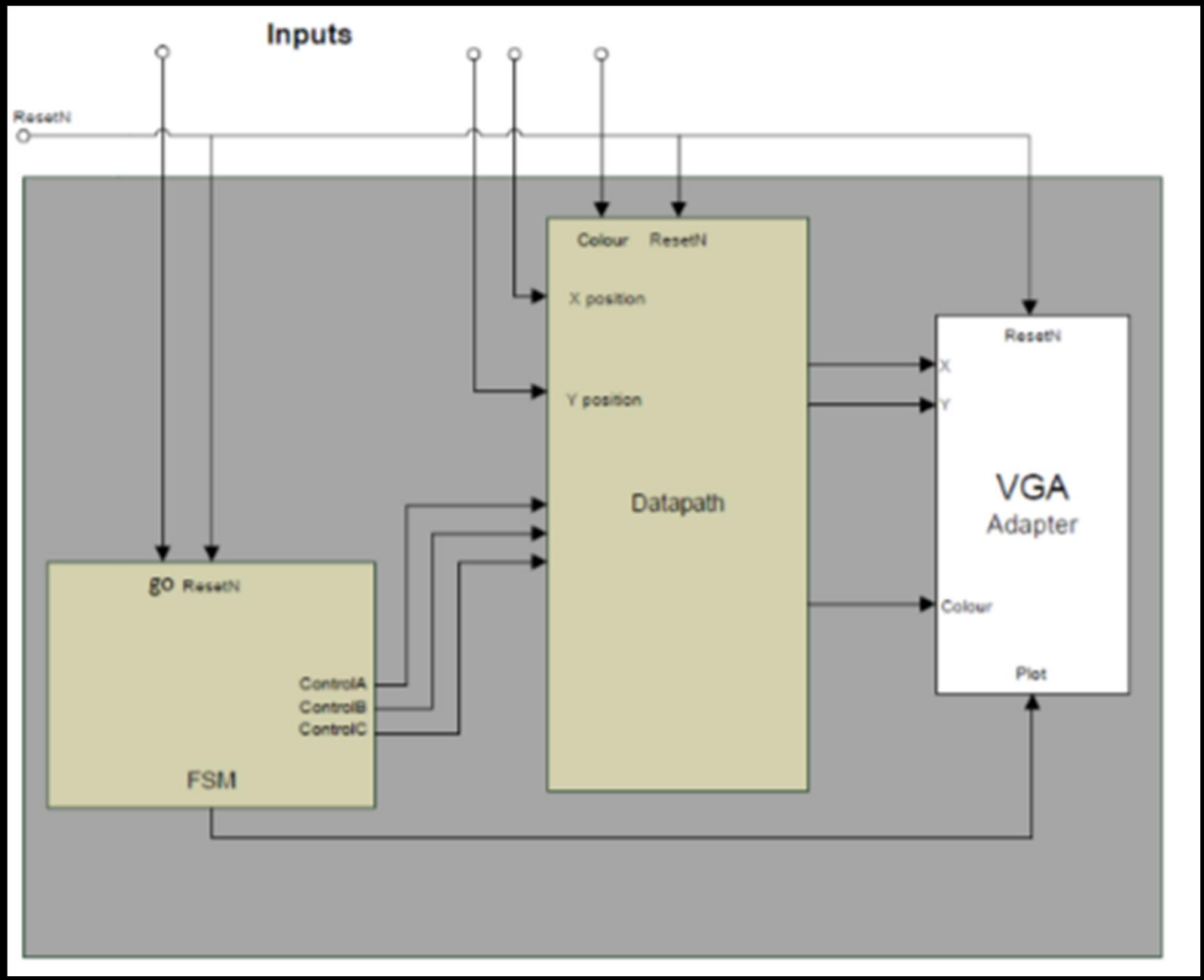
Part II: VGA Display

- Specifying the inputs to the VGA adaptor will set a single pixel to a single colour.
  - How would you make a box on the screen?
- Given input coordinates `X` and `Y`, make a `4x4` box of coloured pixels, using `X` and `Y` as the top left corner of the box.



Part II: VGA Display

- Components needed:
  - VGA adaptor (provided by us)
  - Datapath that takes in:
    - X and Y (through switches)
    - control signals (from KEYs, clock and FSM)
  - FSM:
    - Controls datapath to load X and Y values, and iterate through the pixel locations that need to be updated (relative to X and Y).


Inputs
ResetN
Colour
ResetN
X position
Y position
Datapath
ResetN
X
Y
VGA
Adapter
Colour
Plot
go
ResetN
ControlA
ControlB
ControlC
FSM

Part II: VGA Display

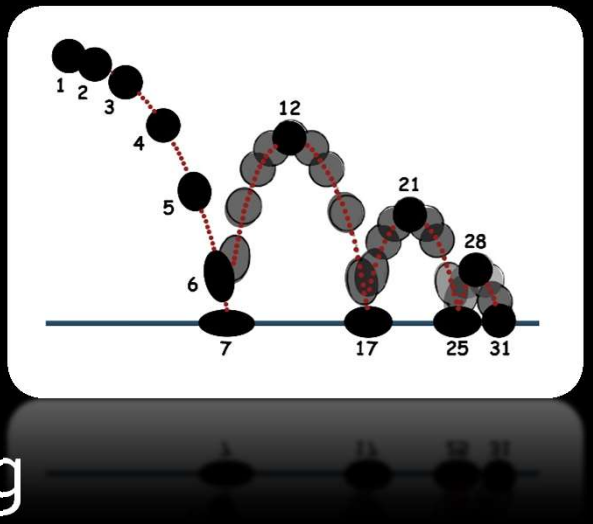
- Hints:
  - Have tests to verify that each component works on its own.
    - Try using the VGA adaptor to draw a single pixel, make sure the datapath works on its own, verify that the FSM is moving from state to state as expected.
  - Consider using counters to store the offsets from `X` and `Y` that need to be displayed.
  - Background is black by default, so test with pixel colour values other than (`0`,`0`,`0`)

Part II: VGA Display

- When testing your VGA code in the lab, look for this switch:

- This will swap the VGA screen between the workstation and the FPGA board.

Part III: Animation (bonus)



- Note: This part is optional, but can be done for bonus marks in the course.
- Animate a box by drawing it, then waiting, then drawing another at a different location, then waiting…
- Many projects will use animation in some form, so you should try this part out!
  - Also…bonus marks! ☺