

CSEE 4840 Embedded systems Design

Light Saber generator- *Return of the Jedi*

Anusha Dachepally (ad2657)

Raghu Binnamangalam (rsb2145)

Devesh Dedhia (ddd2121)

Roopa Kakarlapudi (rk2489)

Overview

- Motivation
- Goals
 - ▣ Real Time Video Display
 - ▣ Color detection
 - ▣ Light saber generation
- Real time video processing
 - ▣ Processing at 60 fps

Motivation

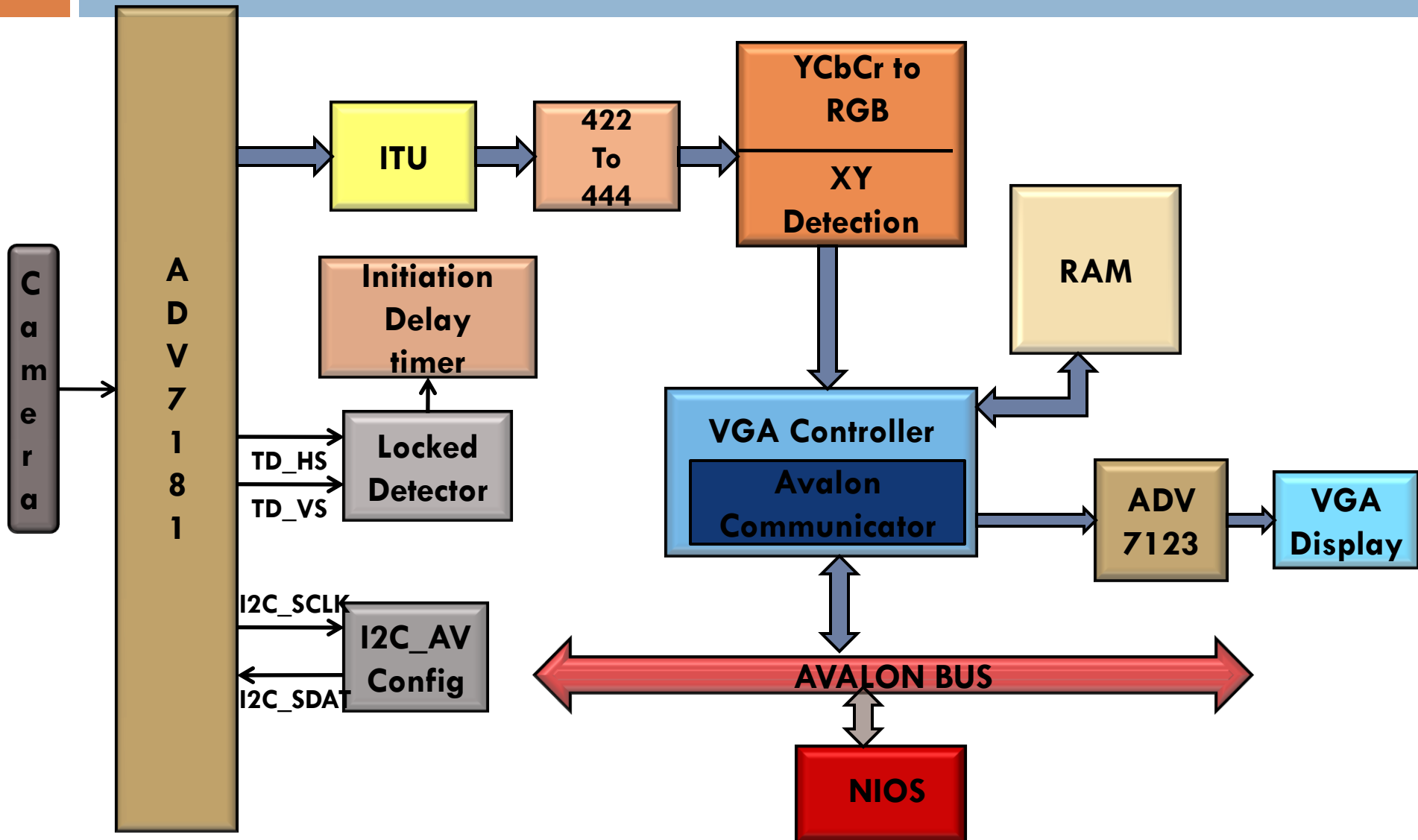
- To learn integration of hardware and software
- Real time video processing on the FPGA board
- Inspired by
 - ▣ Luke Skywalker (Star wars!!)



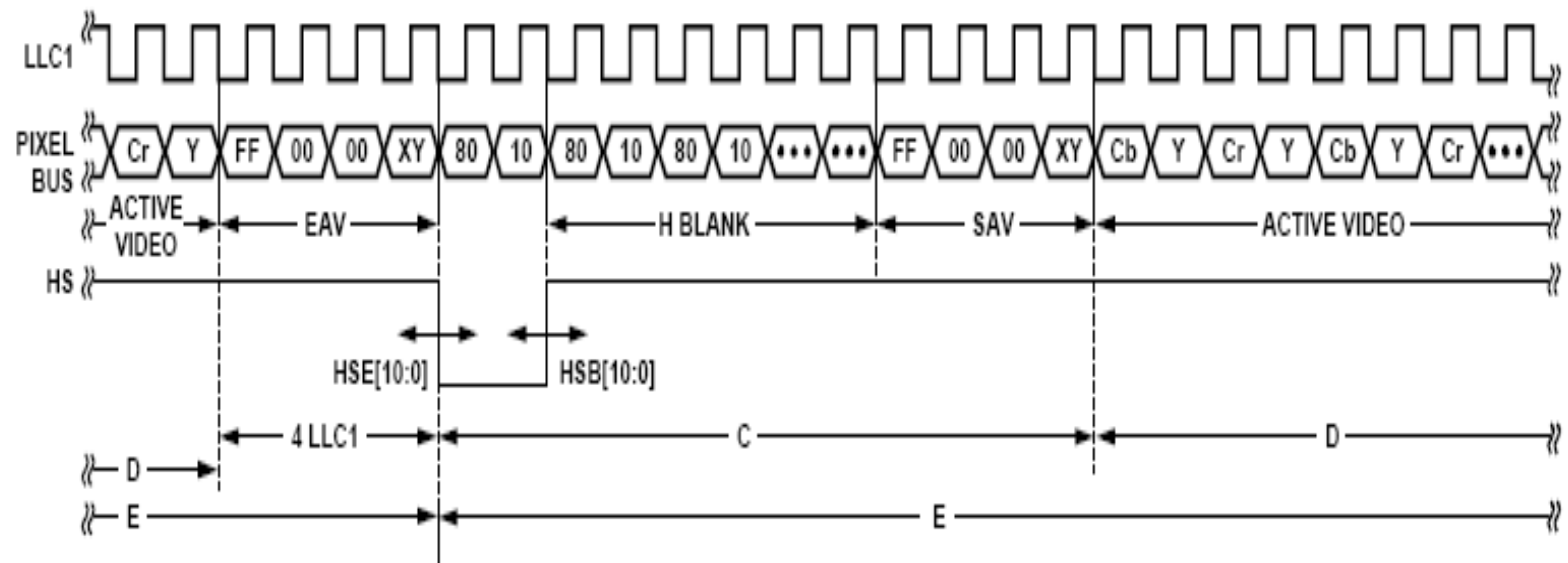
Objectives

- To capture video in real time of a person holding a sword
- Do real time video processing at 60 fps
- Replace the sword with a light saber on the VGA display

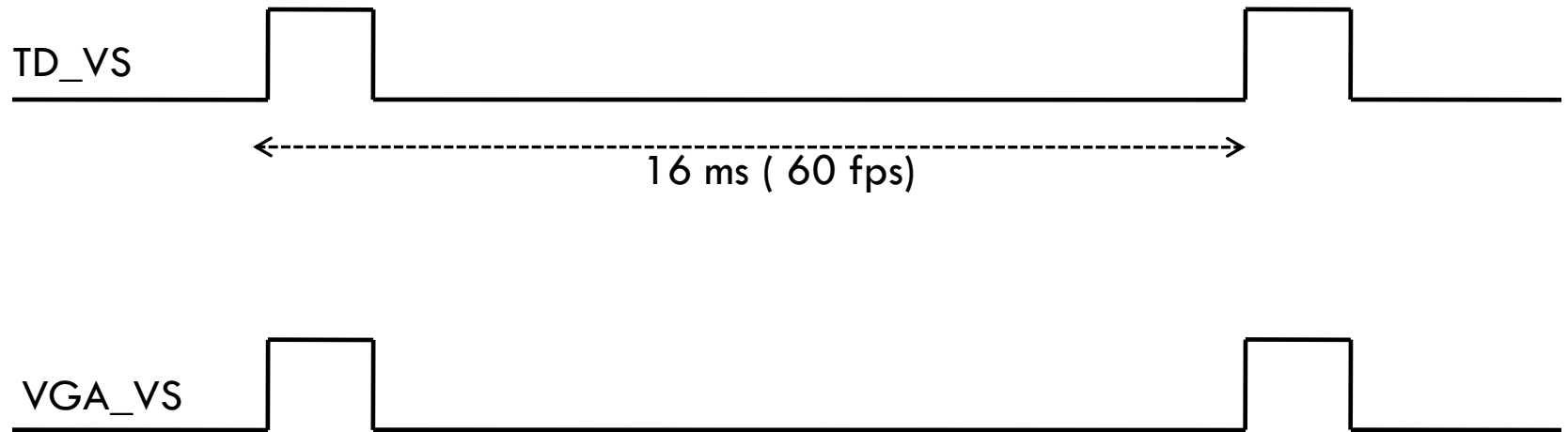
System Architecture



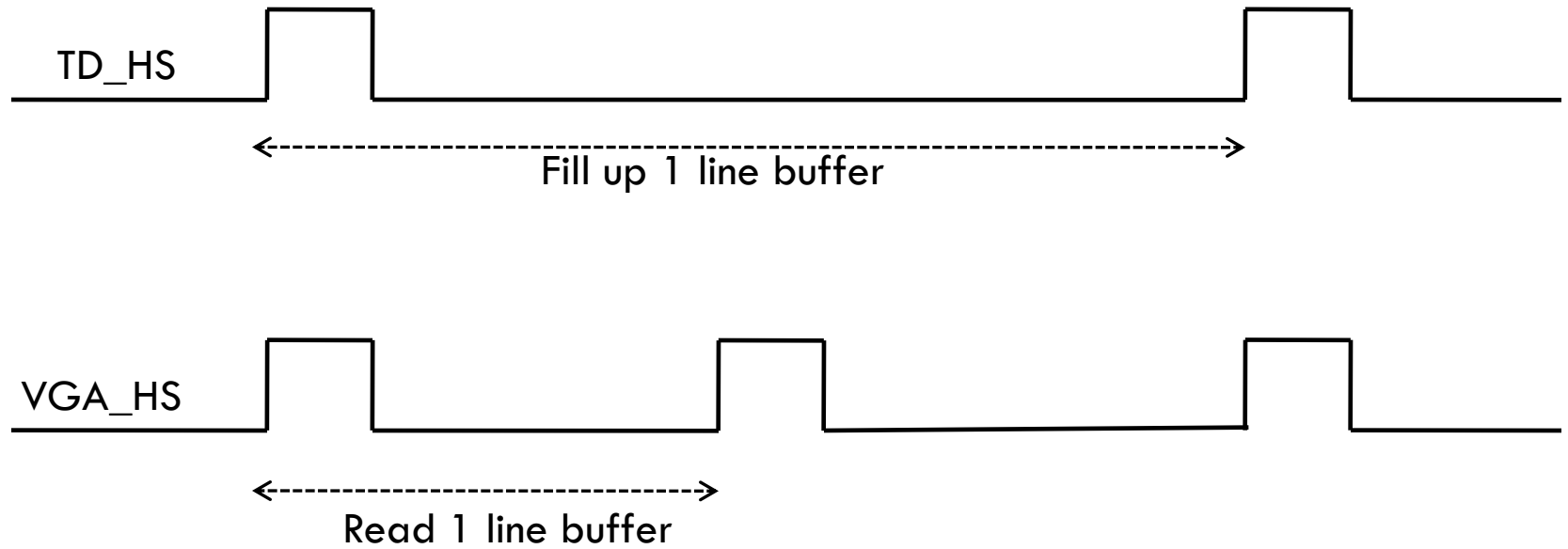
Timing diagrams



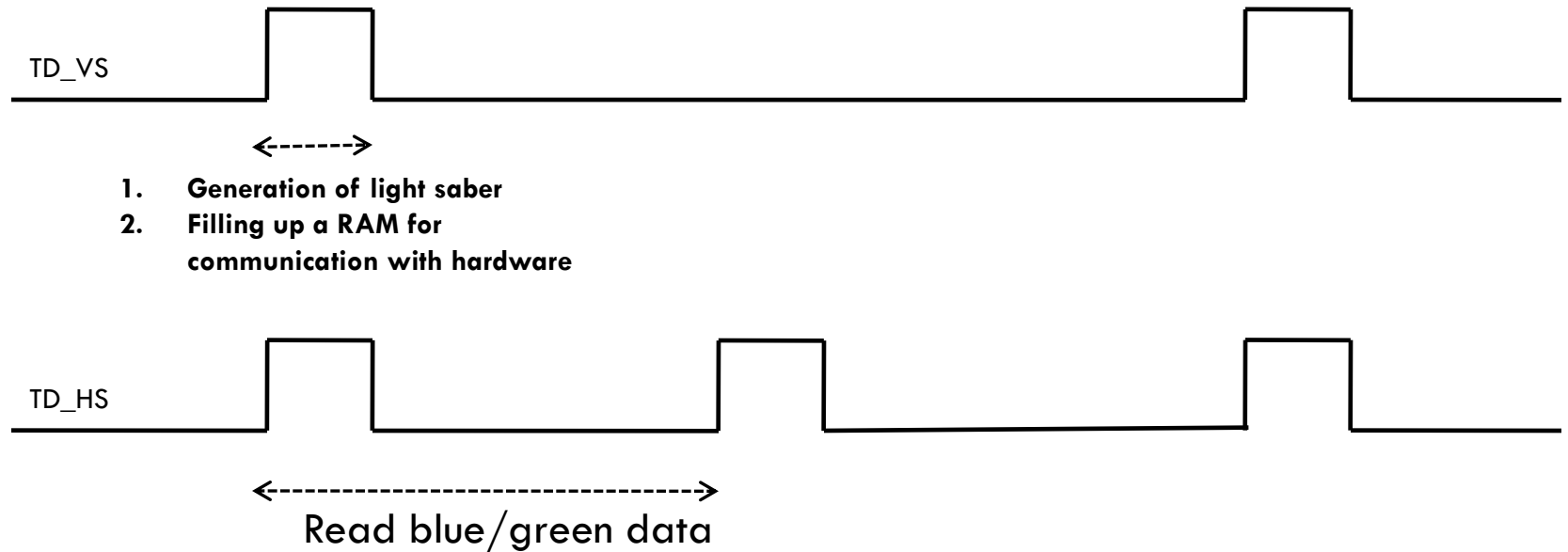
Timing diagram 1



Timing Diagram 2



Timing diagram 3 (Software)



Real time Video Display

- 60 fps display
 - ▣ To emulate real time video
- No frame storage
 - ▣ SRAM not dual ported
 - ▣ SDRAM is a time deterrent
 - ▣ Real time display on the fly
- Usage of 2 line buffers
 - ▣ Swap between the 2 line buffers
 - ▣ Avoids loss of information
- No interlacing
 - ▣ replicating even/odd frame

Color detection

- Conventional approach
 - ▣ Euclidean distance
 - ▣ Intense computations hence delay
- Our approaches
 - ▣ RGB vs YCrCb
 - RGB has varying threshold with varying intensity of light
 - YCrCb has greater tolerance

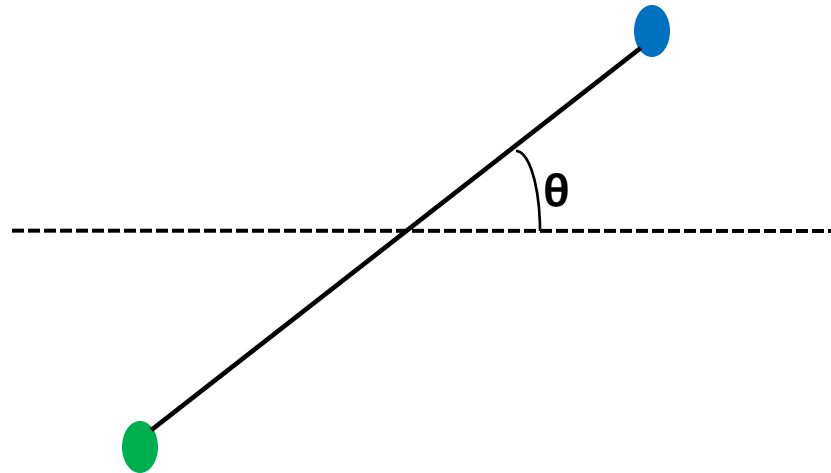
Light saber generation



- Find centre of mass for ends of sword

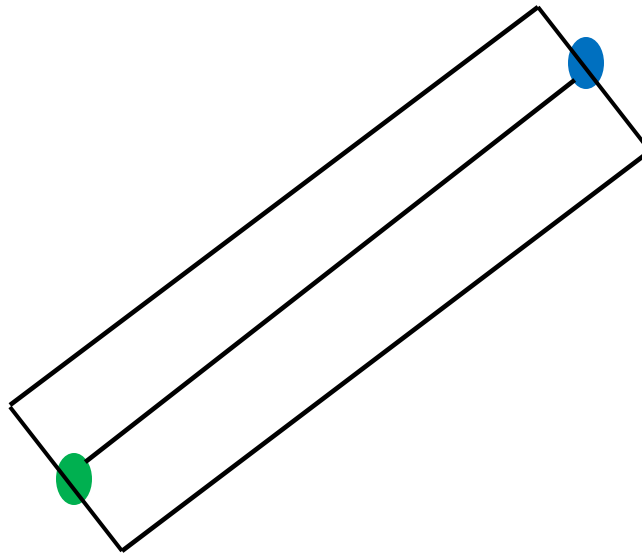
Light saber generation

- Calculate the slope of the line



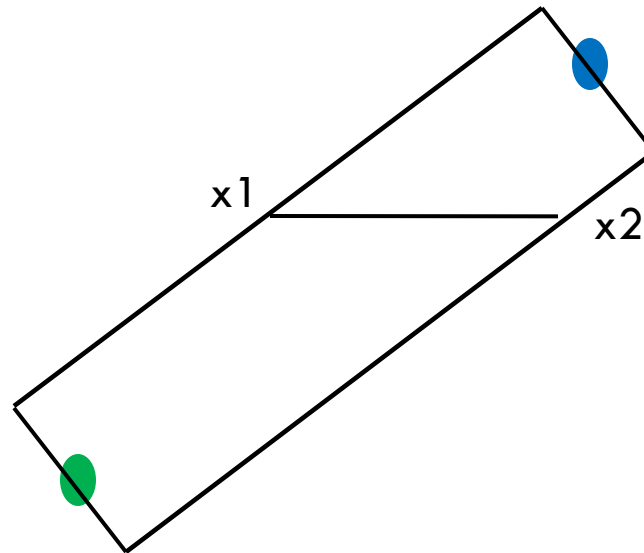
Light saber generation

- Use a predefined sword width and halo width



Light saber generation

- Filling the coordinate table



Software Design

- Centre of mass calculation
- Using theta calculations find the four edge points of the light saber for a predefined width.
- Use a table to store the X1 and X2 for each line displayed by the VGA.

Design challenges

- For real time video display
 - ▣ Horizontal sync and Vertical sync w.r.t incoming video from ADV 7181
- Achieving 60 fps
- Making the system work without frame buffers
- Deal with multiple clock domains

Design challenges(contd..)

- Floating point calculations
- Line drawing algorithms
- Synchronization with the VGA module in hardware
- Hardware signals are sampled more than once

Lessons learnt

- Start early
- DE2 Terasic real time video code is all junk
- Design timing diagrams before implementation
- Conventional measuring techniques like CRO are Life savers
- More time for debugging

May the force be with you!!!





Thank You