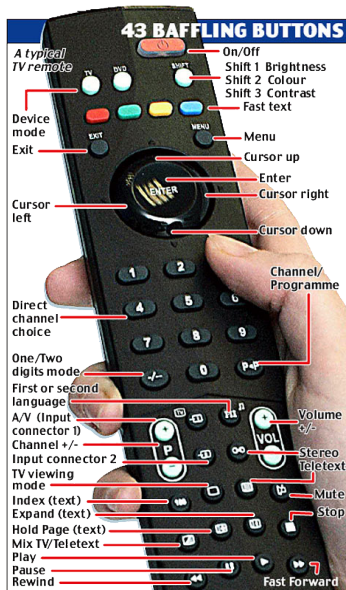


TEAM#1

Gestural Remote Control Using FPGA

11/15/12

Motivation



Proliferation? [2]
[3]

Helping the Elderly [4]
[5]

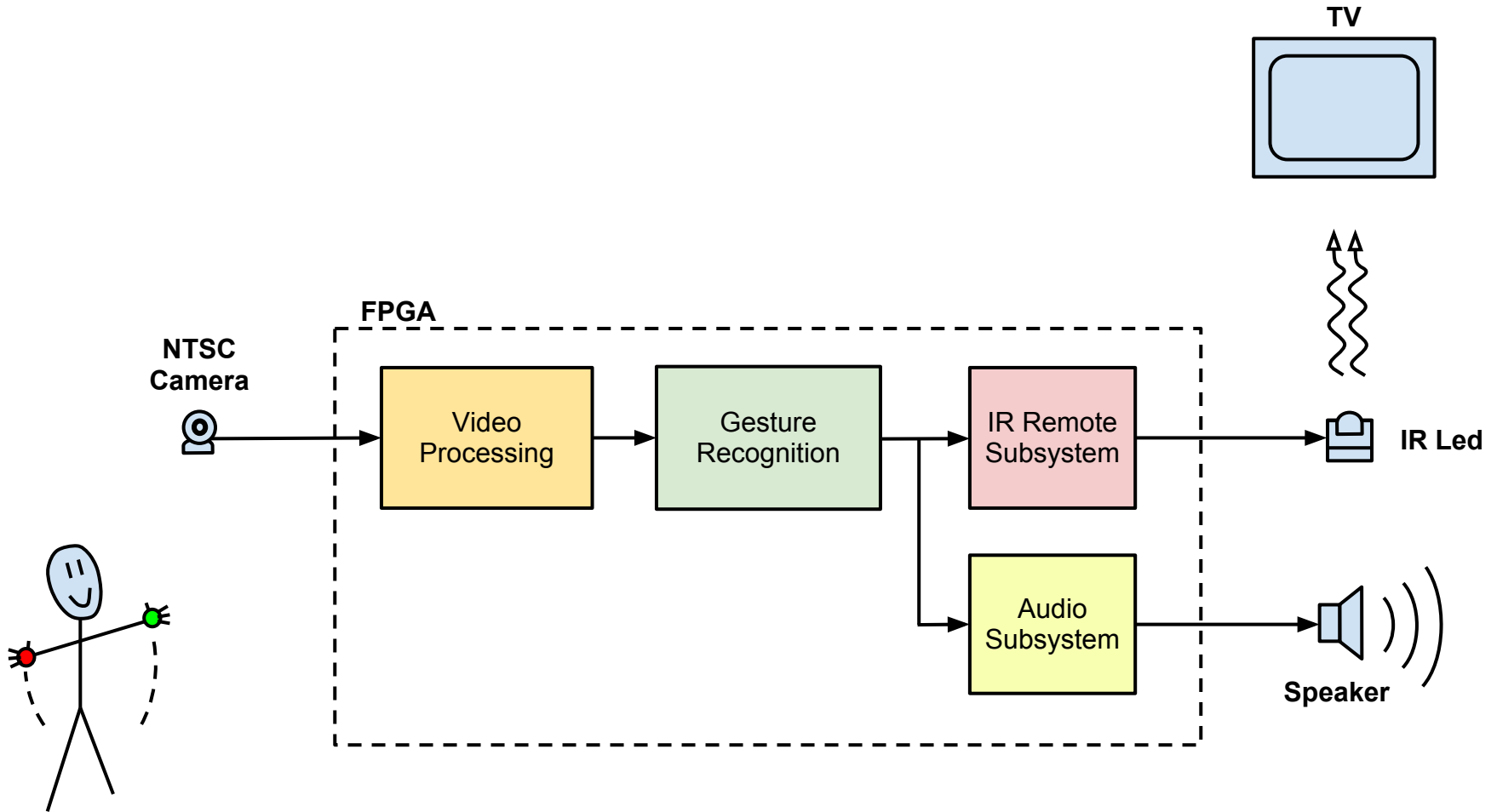
- | | | |
|-----|---------------|--|
| [1] | DailyMail | goo.gl/GHPeM |
| [2] | InvertorSpot | goo.gl/S29Pv |
| [3] | Helixsoft | goo.gl/afoG7 |
| [4] | Potvin Newsly | goo.gl/DDRLl |
| [5] | Gizmodo | goo.gl/laUcG |

Confusion? [1]

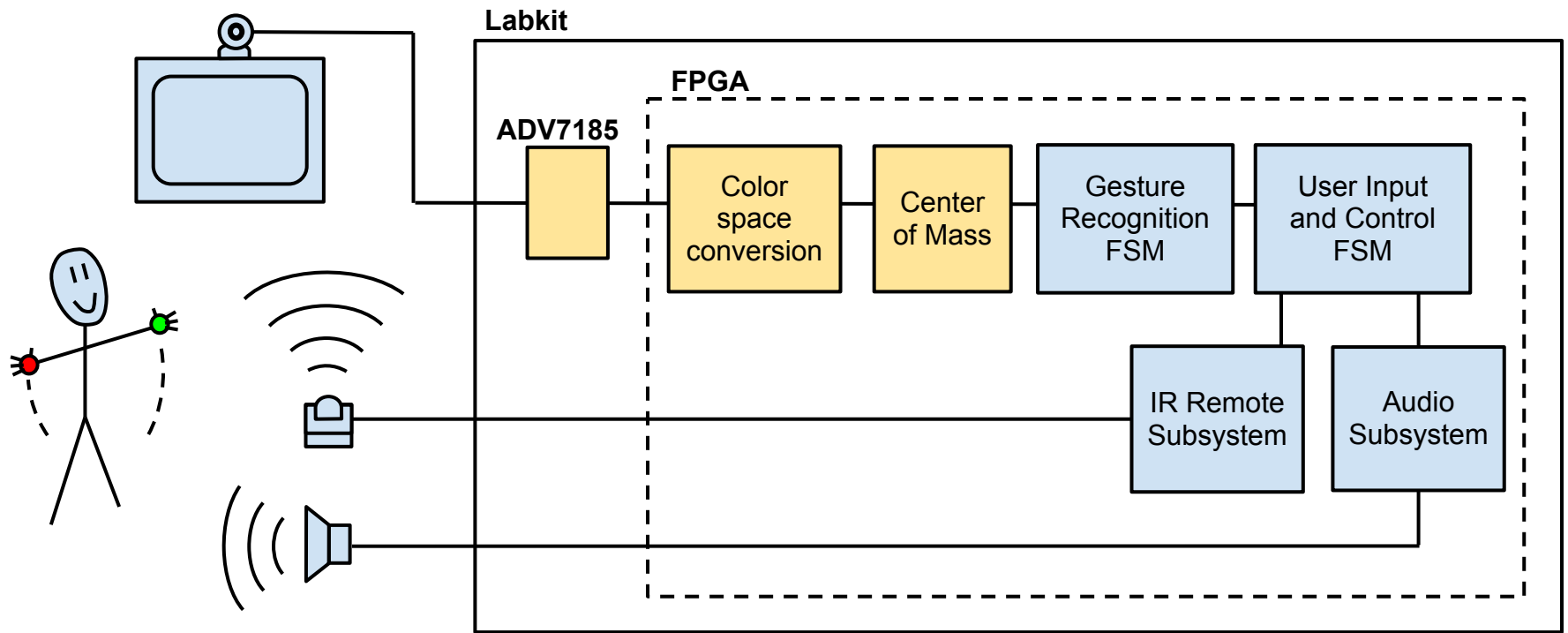
- Gestural Remote Control
 - Control your TV using simple gestures
 - Touch-free interaction
- Advantages
 - Easy learning
 - Controls multiple devices
 - No input device
(can't lose the remote)
- Technical overview
 - NTSC camera
 - User hands are recognized by red and green gloves
 - Audio feedback using wave files stored in flash chip

Gesture	Function
Waving to the camera	TV Power
Swiping left and right	TV Channel -/+
Holding right hand up	TV Volume +
Holding left hand up	TV Volume -

Block Diagram - Top Level



Video Processing



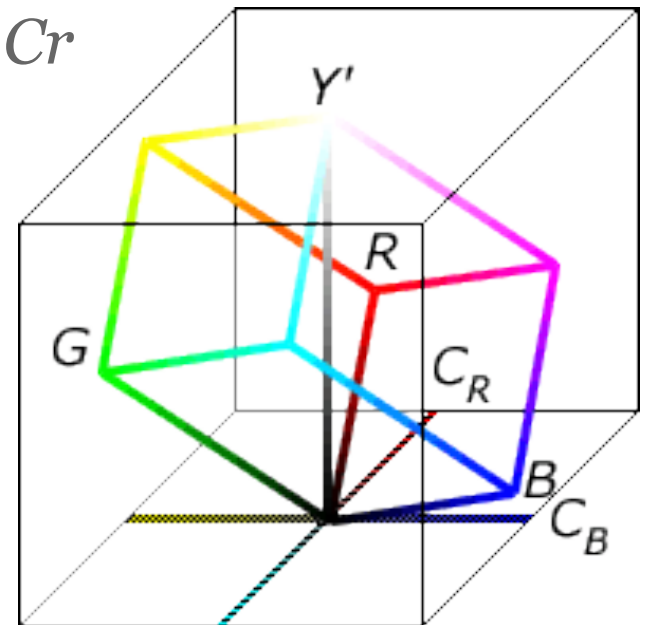
Video Processing - Color spaces

- Color space conversion: YCbCr to RGB
- YCbCr is:
 - the output format of the video decoder chip (ADV7185)
 - a way of encoding RGB information:

$$R = Y + Cr$$

$$G = Y - (K_{by} / K_{gy}) \cdot Cb - (K_{ry} / K_{gy}) \cdot Cr$$

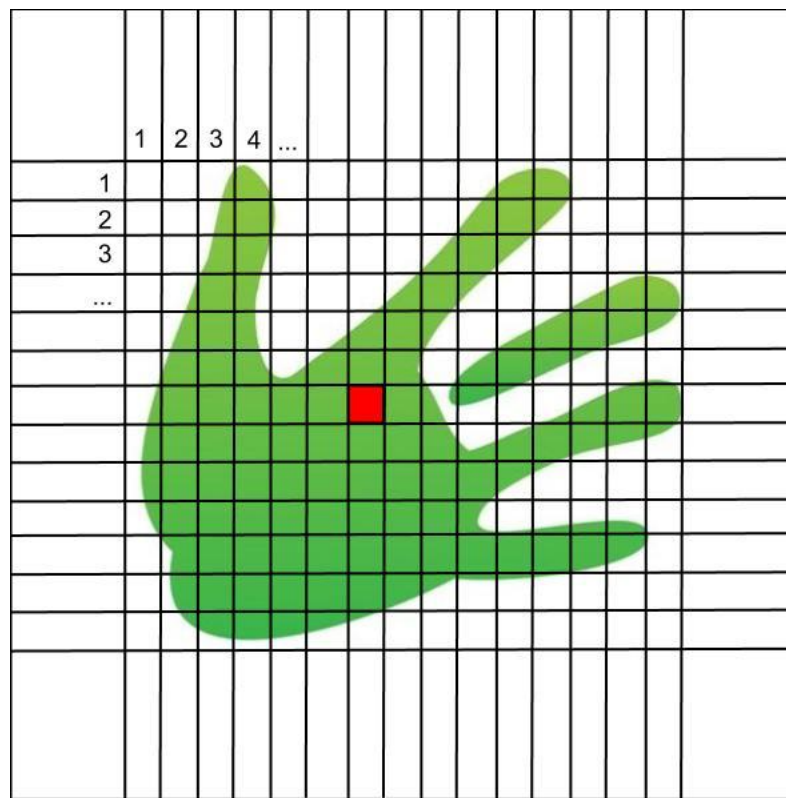
$$B = Y + Cb$$



Video Processing - Center of mass

Algorithm:

- if (green > threshold) accumulate X and Y coordinates
- divide the sum of coordinates by the number of captured pixels
- send X and Y coordinates as well as ready signals for both hands

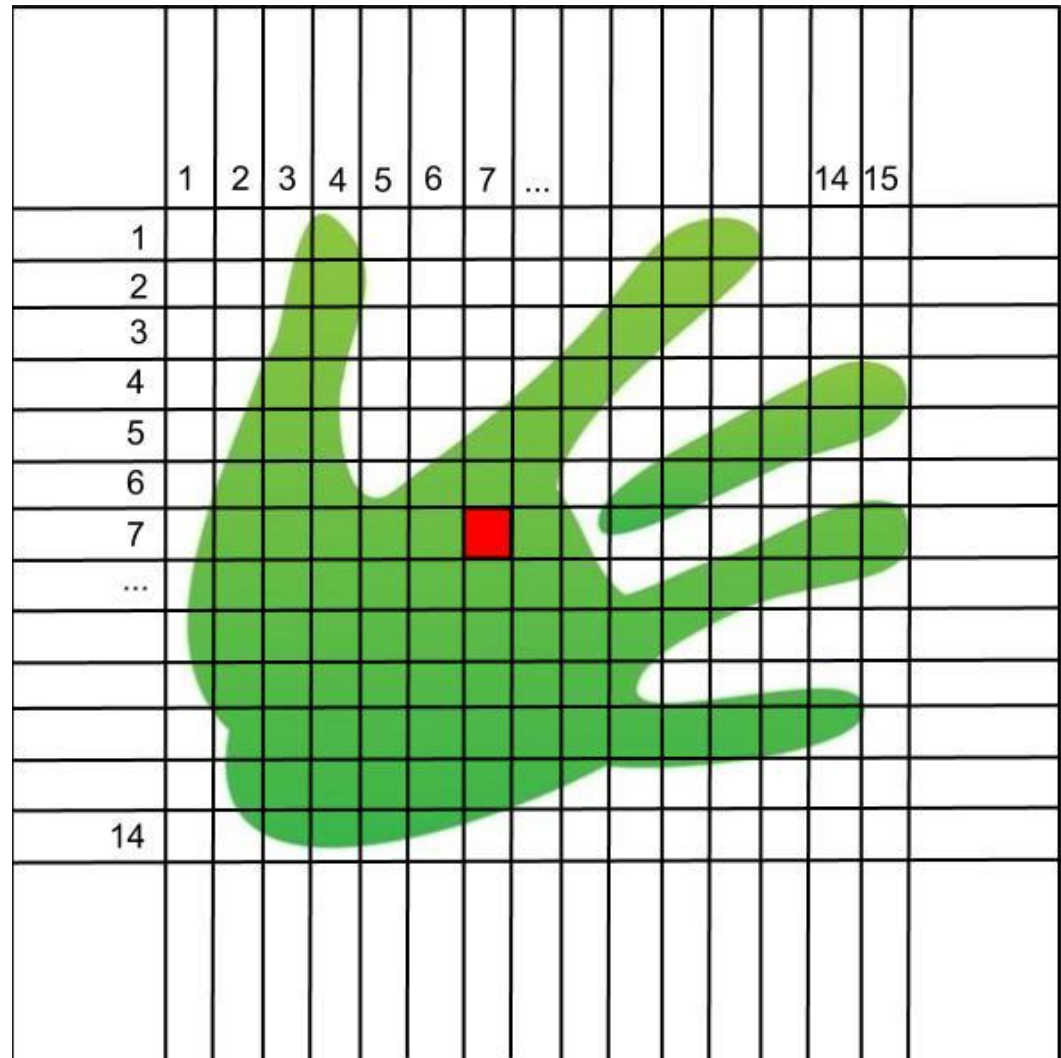


Video Processing - Center of mass

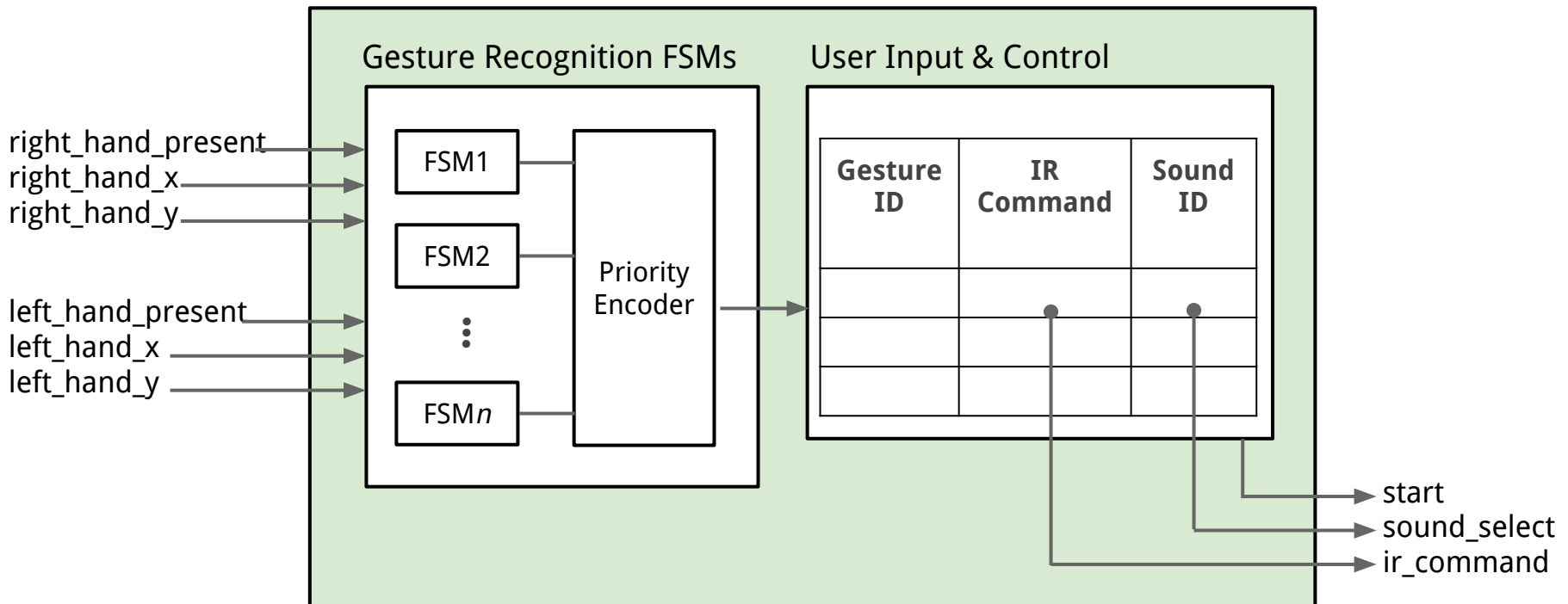
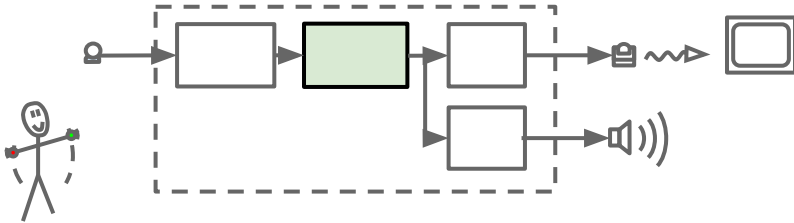
Formulas:

$$X_c = \frac{\sum x * threshold(x, y)}{\sum threshold(x, y)}$$

$$Y_c = \frac{\sum y * threshold(x, y)}{\sum threshold(x, y)}$$

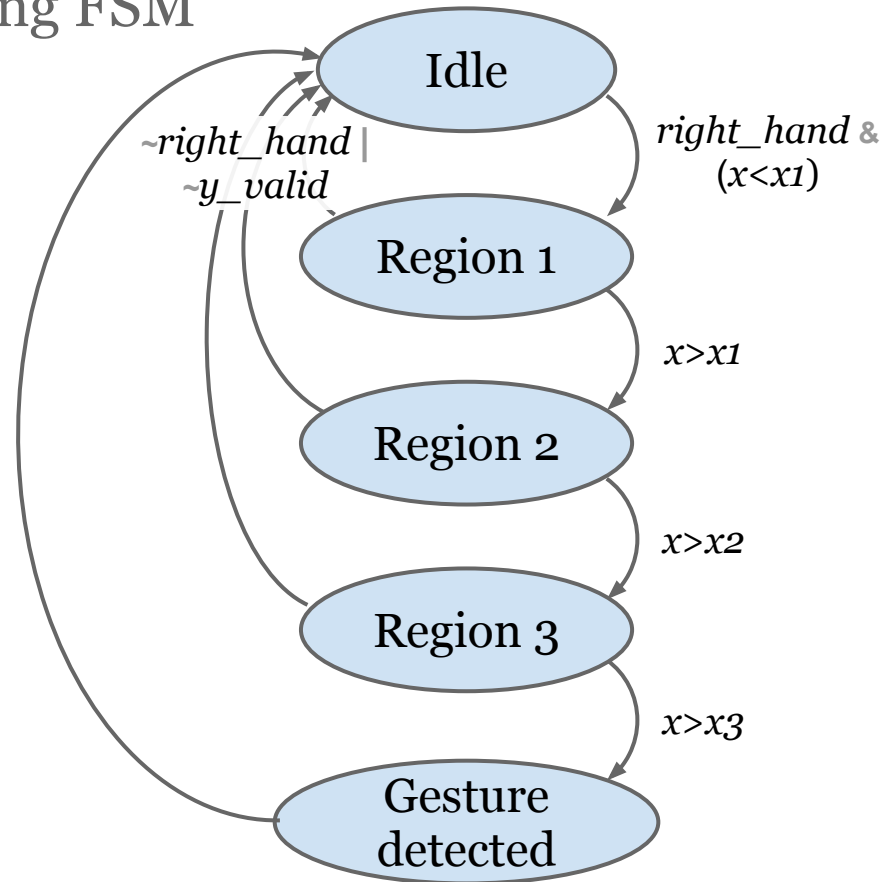
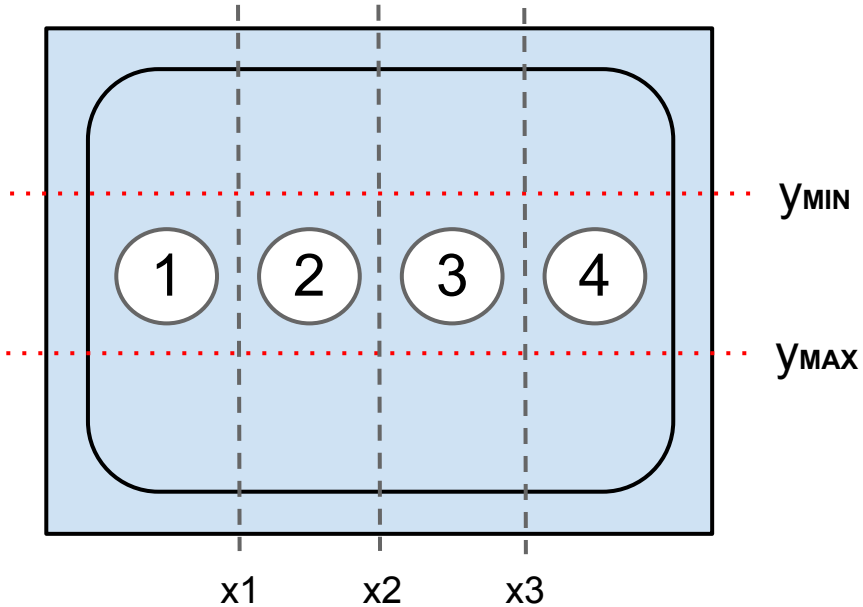


Gesture Recognition

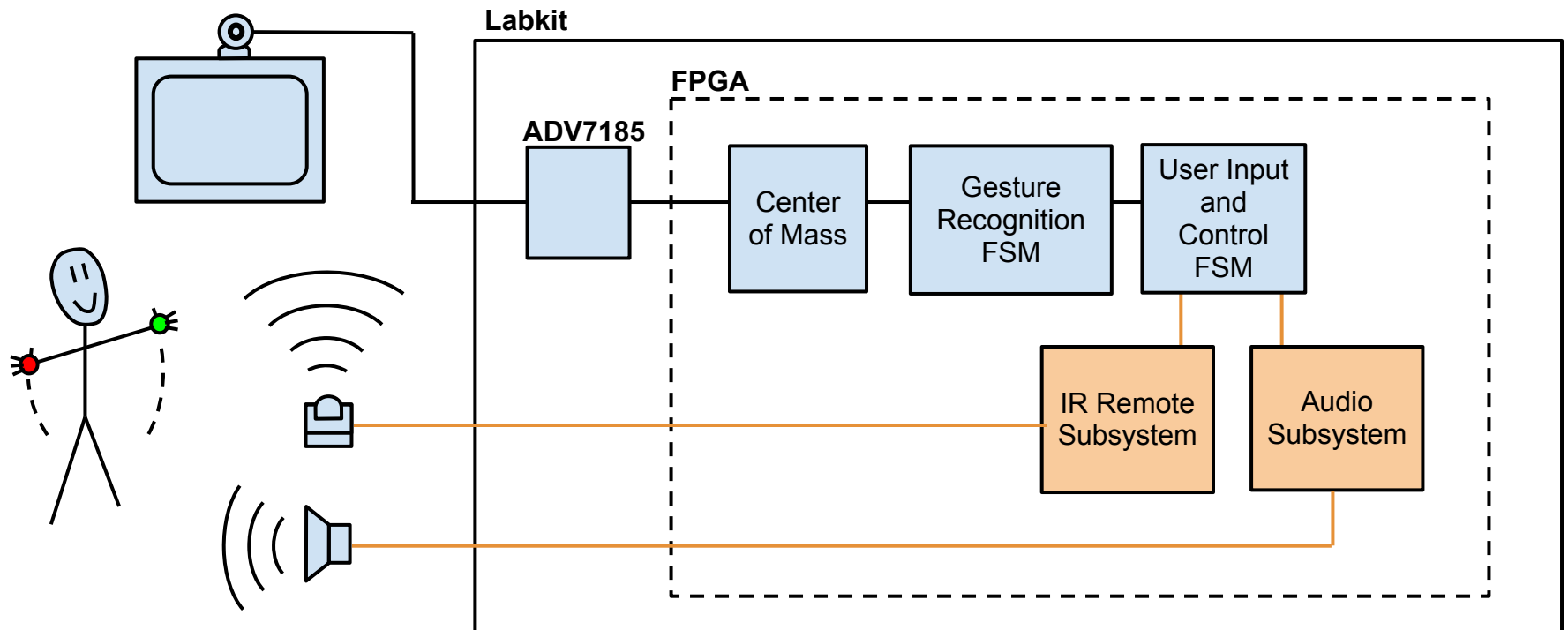


Gesture Recognition FSMs

- Each gesture has a corresponding FSM
- FSM Inputs:
 - $x, y, present$
 - $x, y, present$
- Example: Right Swipe FSM



Block Diagram - IR and Audio



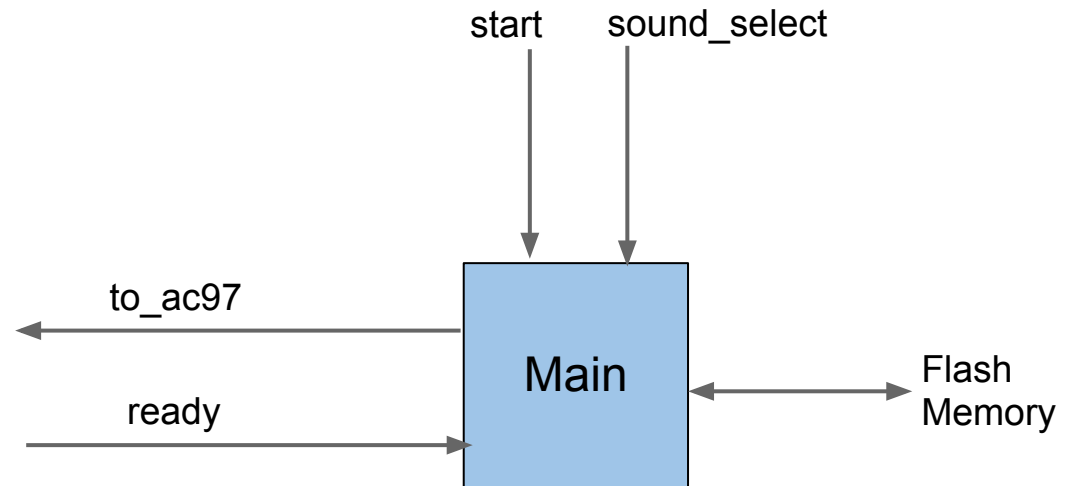
Writing to Flash Memory

- Different project entirely
- Sound files:
 - generated in wave (.wav) format
 - converted to AC97 format using a Matlab script
 - sent to labkit using a Python script
- Output files transmitted to the labkit via USB module
- Custom hardware reads USB data and writes to flash chip



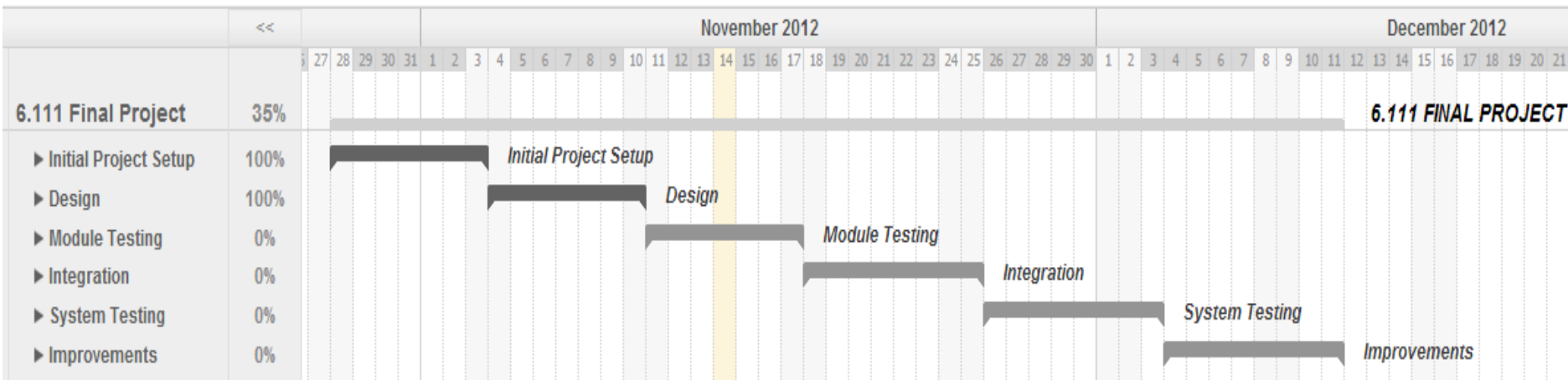
Audio Module

- Similar to lab 5a
- Inputs:
 - start
 - sound_select
 - ready
 - flash data
- Outputs:
 - flash_address
 - wave samples to AC97 for audio



- Similar to lab 5b
- Inputs:
 - ir_address (TV, DVD, VCR, etc.)
 - ir_command (Vol+/-, Channel +/-, Power, etc.)
 - start
- Output:
 - 40 kHz modulated IR signal
 - Connected to external IR led

Timeline



- Graph created with TeamGantt (teamgantt.com)
- Milestones:
 - Finish building modules (11/21)
 - Finish testing modules (11/26)
 - Finish system integration (11/30)
 - Finish testing & debugging (12/5)
 - Finish polishing and improvements (12/9)
 - Final Checkoff (12/10)

Conclusion

- Questions?



Etornauta (WordPress) - goo.gl/K0oSS



CollegeHumour.com - goo.gl/Syl1j