EECS 150 - Components and Design Techniques for Digital Systems

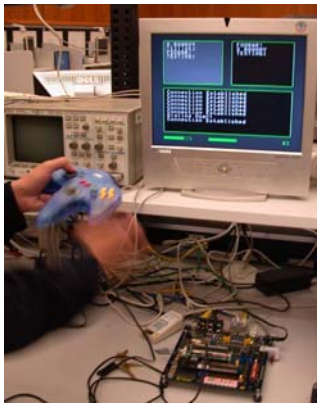
Lec 11 – Project Introduction

David Culler
Electrical Engineering and Computer Sciences
University of California, Berkeley

http://www.eecs.berkeley.edu/~culler
http://inst.eecs.berkeley.edu/~cs150

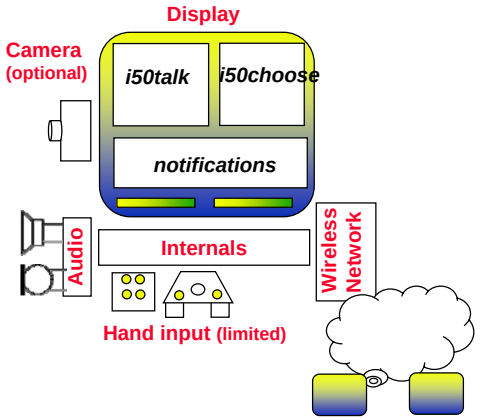
iPhone

i50Phone



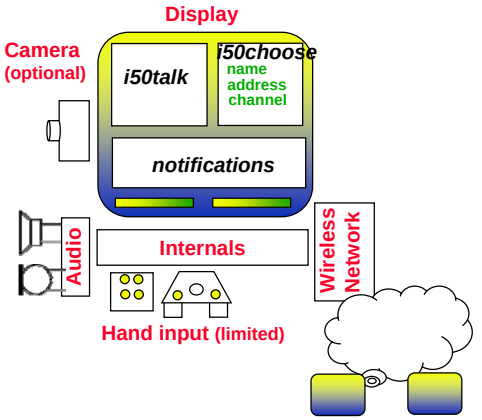
I50phone Concept

- ***i50choose***
  - Define and present local configuration
    - » Name, addr
    - » channel, …
  - Announce to chosen group
- ***i50talk***
  - Collective and individual info about current group
  - Constructed from announcements
  - Select 2-way sessions
- ***Notifications and status***
  - Useful information about what is going on
- **Extensions & Options**



i50choose

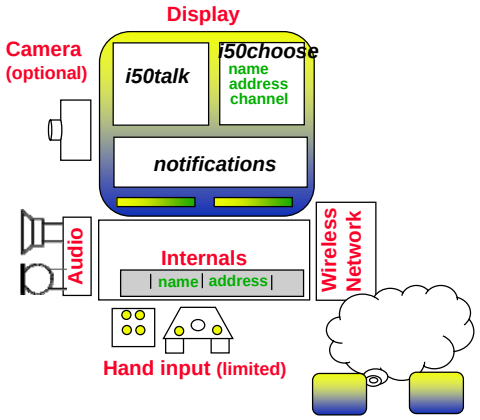
- ***Configure and display various aspects of your local device***
- ***Wireless channel***
  - Determines set of potential participants



i50 announcements

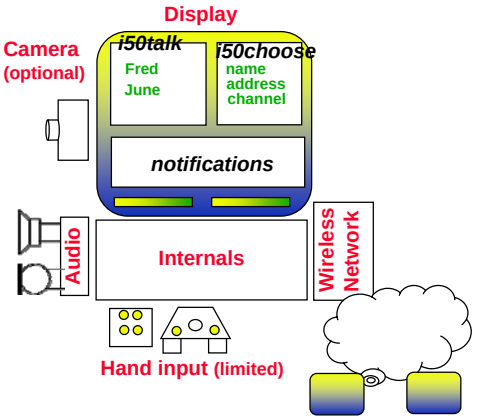
- Periodically, each devices announces itself on its selected channel.
- Wireless network is used for control
  - Like call setup, login, join
- And for audio communication
- And potentially for other things

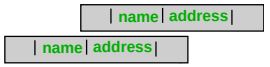
=> wireless protocol with extensible packet format



i50talk Registry

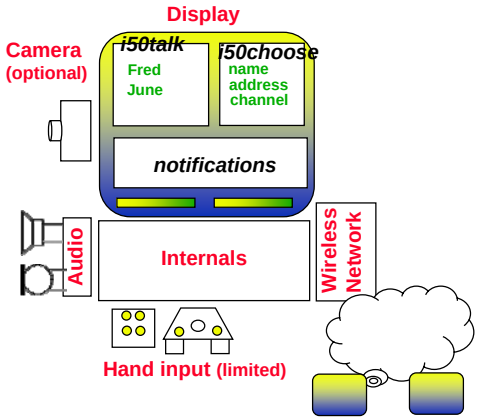
- Announcements received from other devices on the channel.
- Add entry to registry on arrival.
- Display current registry
- Allow user selection among entries
  - Open a audio session
- Age and delete entries
- Choose new channel => repopulate registry





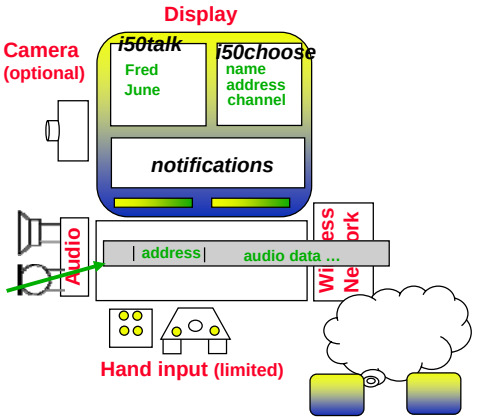
Who Talks to Whom?

- Select i50talk member to make a call
- Accept call
- 2-way conversation over digital wireless
  - Audio signal?
  - Audio compression?
- Party line?



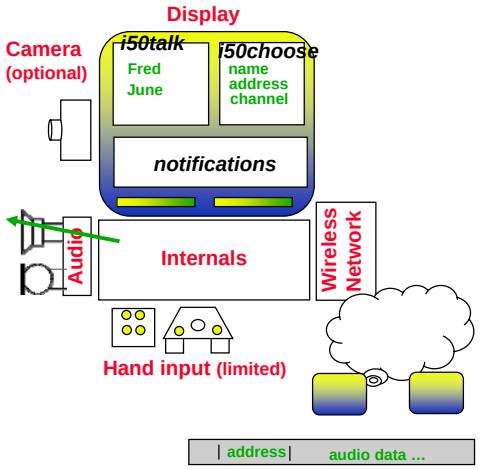
Audio Capture and Transmission

- Capture: convert the analog signal from the microphone into a series of digital values
  - 4 khz sampling
  - 16-bit samples (maybe 12)
- Package a chunk of samples into an audio message.
- Transmit the chuck onto the wireless channel.
  - Specific destination address
  - Broadcast
- 40 B @ 4 kHz => 5 ms voice
- 40 B + 16 B preamble and header @ 250 kbps => 1.8 ms "channel time" to xmit
  - 38% of the channel for 1 way
- Congestion => delay
  - Some jitter tolerable
  - Drop if gets too old



Audio Reception and Presentation

- Receive a chunk of audio samples in an audio message.
- Drive the speaker @ a constant rate
  - 4 kHz
  - DAC – digital to analog
- Buffer enough incoming audio data that can maintain smooth playback



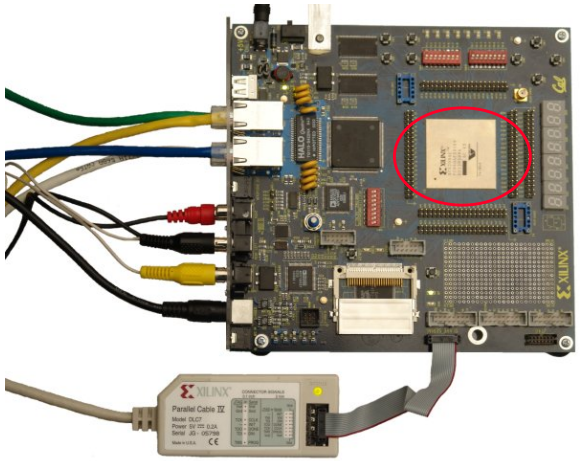
Extensions

- Session record and playback
- Teleconferencing
- Ring tones
- Audio effects
- Background
- Multisource mixing
- Registry images
- Video effects
- Game elements
- Text exchange

Functional Elements

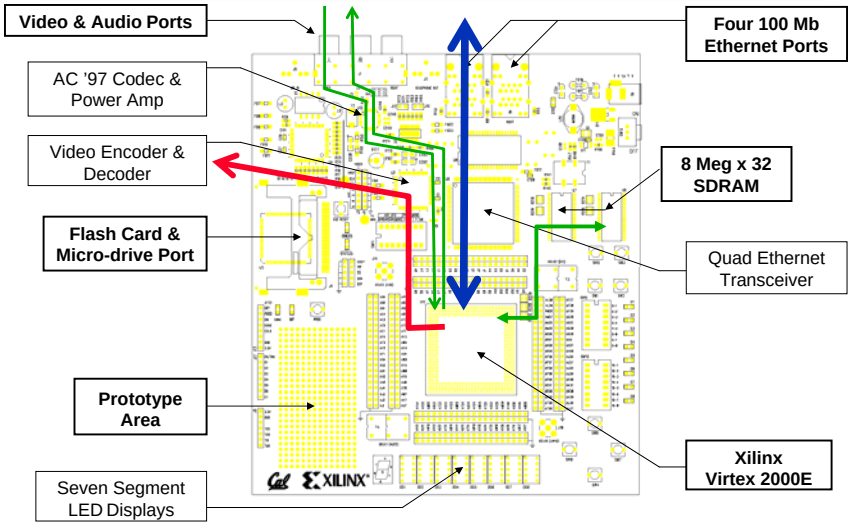
- Construct Local Configuration
- Render display elements
- Announce Self to Group as Configured
- Maintain Registry of announcements
- Capture, packetize, transmit RT Audio
- Receive and Play RT Audio packets
- Play digital audio files
- Capture button & Cursor actions
- Receive / Transmit digital audio files (???)
- Transmit and Receive button & cursor actions (?)

CaLinx2 – Your EECS150 …



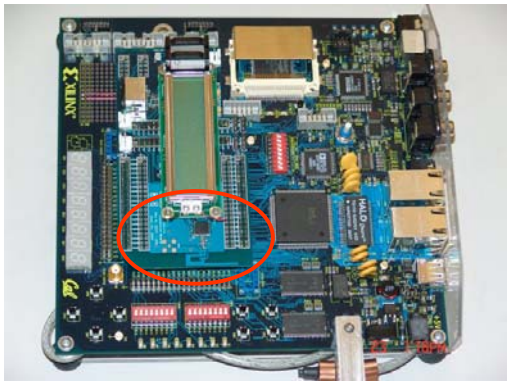
Focus so far has been on constructing the combinational logic, storage elements, and interconnect to form useful synchronous systems

Extending digital design



Over Wireless network

- IEEE 802.15.4 Personal Area Network
- ADC channels
- Simple display
- Serial interface



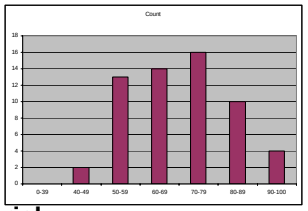
Getting from here to there

- Week 6 – Lab 5: Network Digital Audio
  - Spool winamp stream from ethernet to audio codec
  - Tools: Chipscope.
- Week 7 – CP 1: RT audio record and replay
  - Audio capture on button press from Mic to RAM.
    - » Light LED when speaking is active
  - Audio play on button press from RAM to speaker
- Week 8 – CP 2: Display
  - Render canned source to video using Block SRAM
  - Build basic display capability
- Week 9-10 CP 3: Wireless
  - Stream RT audio to and from 15.4 radio
- Week 11 CP4: Basic i50phone
  - Wireless audio 2-way line with GUI
- Week 12-13: i50phone+
  - Select option that you will implement
- Week 14: Final i50phone+ Project Checkoff
- Week 15: Writeup the Report

Announcements

- Reading for Today: K&B 10.4.1-3
- Mid Terms
  - Mean: 70, Median: 71, Mode: 80, Max: 97
  - Regrade policy: submit written request for grading correction by Friday 2pm. We will review and make final decision.
  - Special offer: Reclaim 20% of points lost by correcting your mid term and turning it in F@2pm.

Range	%	Count
0-39	0%	0
40-49	3%	2
50-59	22%	13
60-69	24%	14
70-79	27%	16
80-89	17%	10
90-100	7%	4



- HWs will provide include review material
- Discuss scheduling of Mid III
- No discussion sections this week
- Friday 9am will no longer be held

Underneath the Project

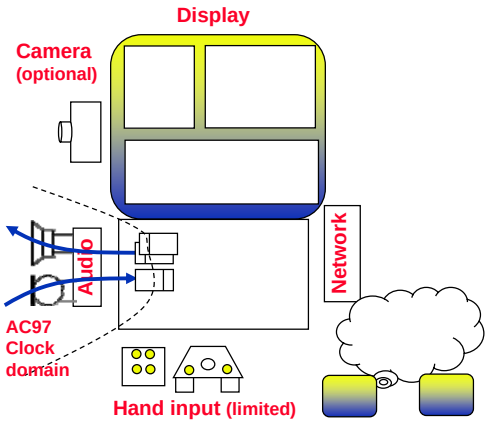
Local Configuration

- Form: logical registers
- Implementation: FPGA
- Entry
  - Hardcoded (bit file)
  - User entry (very limited)
  - User selection
  - Network provided
    - » Ethernet or Wireless

Registry

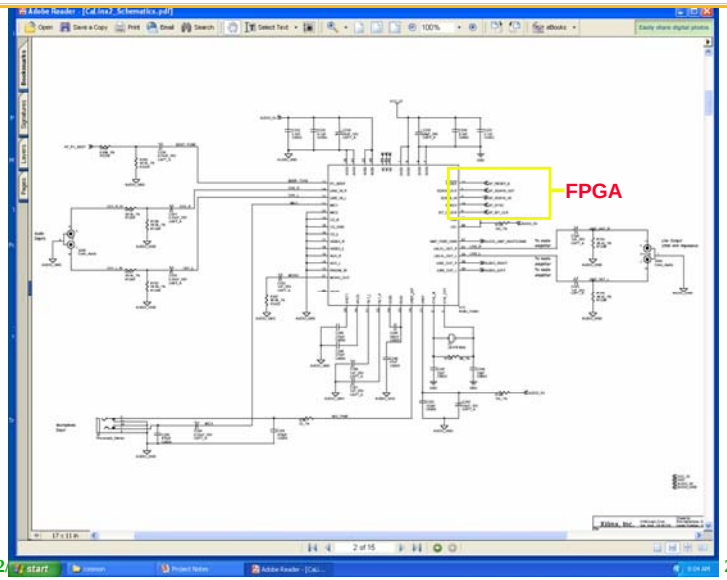
- Form
  - Array of Registers (RAM) - table
- Announcement
  - Packetize Configuration Registers
  - Packet format specification
  - Transmit according to current channel config
- Current Membership
  - Array of entries, timestamped
  - Entry insertion / update on rcv announcement
  - Entry reclamation
    - » Aging and leave
  - Channel change clear

Basics: Audio

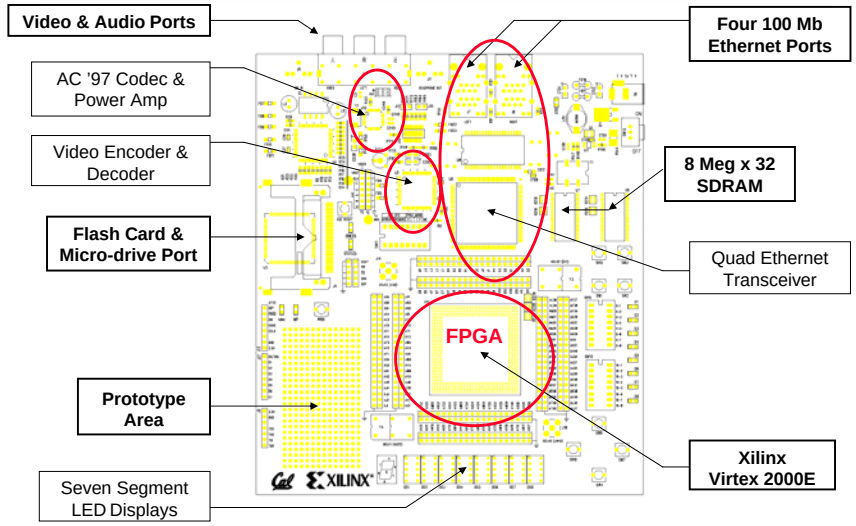


- Capture real time Microphone input in digital form
  - Simple processing and packetization
  - Optional coding / compression
  - Store or transmit
- Render packetized Mic stream to speaker
- Render digital coded audio
- Issues
  - Time slotted serial protocol to/from queue
    - » State machine
  - Clock domains
    - » AC97 vs Core
  - Analog <-> Digital for Audio
    - » Signal / Data
  - Audio coding
- Data sheets
  - Audio Codec LM4549A
- Standards / Protocols
  - AC97 Audio Codec

Calynx2: Audio



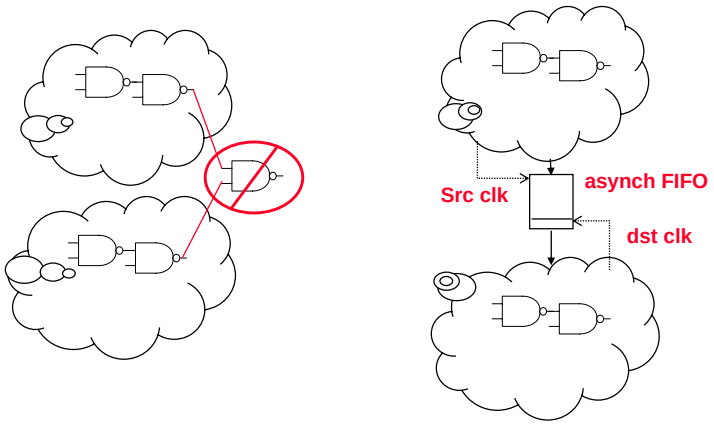
Clock Domains in EECS150



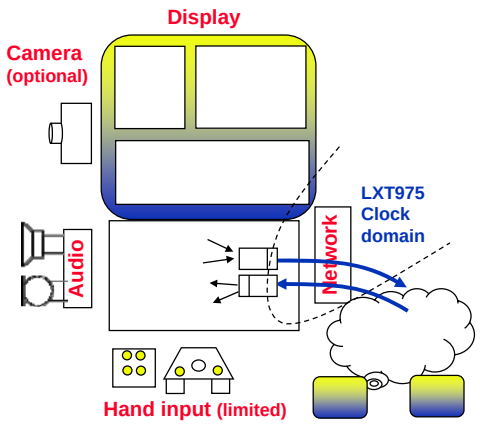
Clock Domain

- **Wikipedia: A clock domain crossing (CDC), or simply clock crossing, is when a signal crosses from one clock domain into another. If a signal does not assert long enough and is not registered, it may appear asynchronous on the incoming clock boundary.**
- **Clock domain is a collection of digital devices (gates, FFs, registers) operating on a common clock.**
- **Everything we've learned about synchronous systems is WITHIN a clock domain.**
- **The key is dealing with multiple clock domains is crossing the bopundaries**
  - **to be very explicit where and how**

Example

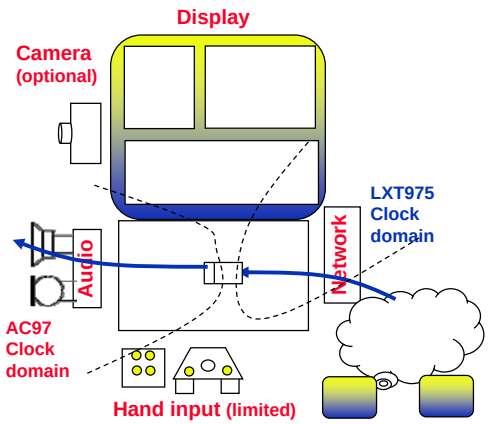


Basics: Network Ethenet



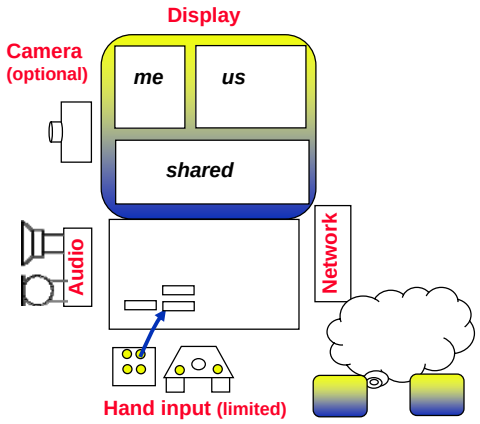
- Spool packets to 802.3
  - Mux sources
- Spool packets from 802.3
  - DeMux sinks
- Issues
  - Header formating
  - FIFOs
- Data sheets
  - LXT975
- Standards / Protocols
  - IEEE 802.3

Lab 5



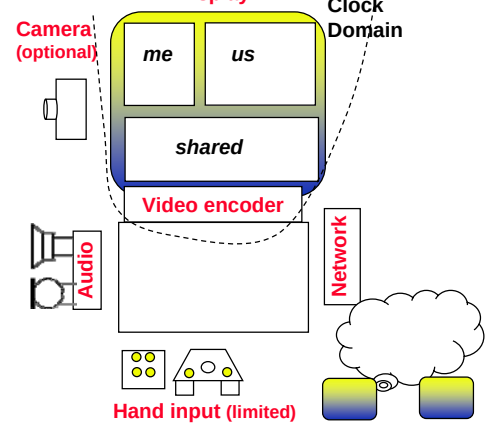
- Use the FPGA and its asynch FIFO primitive to cross two clock domains
- No FPGA storage elements
- Ethernet Header
- Packet formats
- MPEG encoded digital audio

Basics: Hand Input



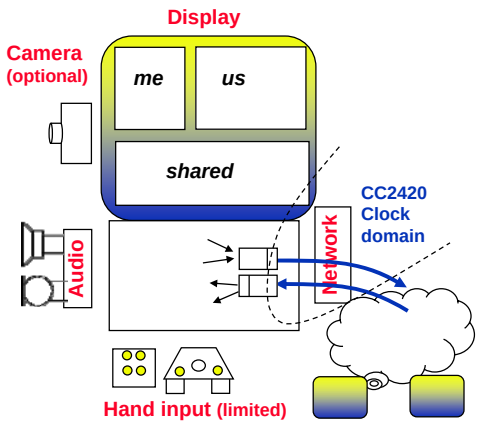
- Poll buttons and register state
- Capture NC64 Controller input
- Issues
  - Very simple analog capture
    - » 1 bit button
  - NC64 serial protocol
  - Aynchronous handshake
    - » Different from crossing clock domains
  - Integration
    - » Display elements and system logic
- Data sheets
  - Non-existent NC64
- Standards / Protocols
  - defacto NC64

Basics: Display



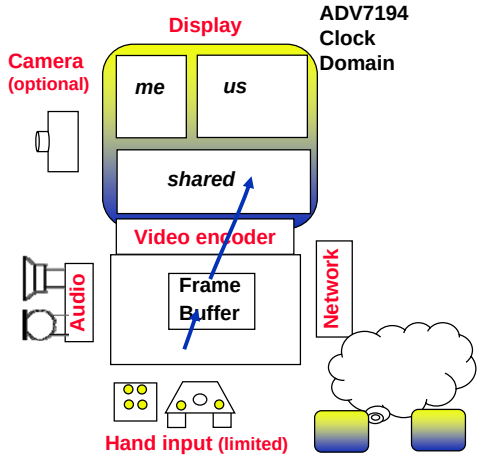
- Render pixels
- Render objects
  - Rectangles
  - Text
- Issues
  - Video coding (YCrCb, 4:2:2)
  - Screen Clock Domain
  - Pixel generation
    - » Frame buffer vs algorithmic
  - Interface abstraction
    - » Pixel queue vs pt
- Data sheets
  - ADV7194
- Standards / Protocols
  - NTSC
  - ITU
  - I2C

Basics: Network Wireless



- Spool packets to 15.4
  - Mux sources
- Spool packets from 15.4
  - DeMux sinks
- Issues
  - Frame Format
    - » Header, Payload
  - SPI protocol
  - CC2420 Configuration
  - MAC
    - » CCA, BackOff, TX
    - » RX / TX state machine
  - Encapsulation
    - » No Net / Transport
    - » Application Protocol
      - Header, Payload
- Data sheets
  - CC2420
- Standards / Protocols
  - IEEE 802.15.4

Basics: Frame Buffer



- Spool FB to Display
- Render pixels to FB
- Render objects to FB
- Issues
  - SDRAM protocol
    - » CMD Address / Data
    - » Burst
  - Dual Ports
    - » Arbitration
    - » Req / Grant
  - Simple Data structure
    - » Base address, Row
- Data sheets
  - SDRAM Chip MT48LC16M16A2TG -7E

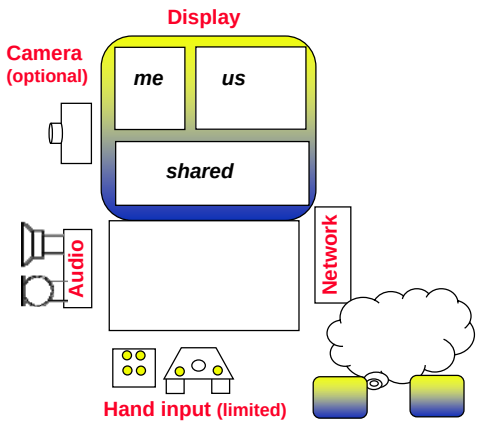
FA07 RAM

- We'll use simpler block RAM "object buffer" in Check Pt 2.
  - Screen positions point to character map
- We'll bring SDRAM in later as audio storage
  - Unencoded digital audio streams
  - MPEG coded – ring tones, etc.

Real Time Audio

- Capture and Packetize
- "Silence" suppression
  - Fixed time window per packet => curtail and send
- Bandwidth
  - 8 KHz sampling x 8 bits => 64 kbps , 8 kBps
    - » 20% of channel
  - One reasonable active voice per channel
  - 64 byte packet => 1/128 sec => 8 ms of voice
  - ~100 byte frame @ 40 kB/s => 2.5 ms of radio
- Contention Protocol
  - Limit generation rate (min interval)
  - Should have reasonable bidirectional voice
  - May want to suppress low gain origination during active reception
  - Favor reception over transmission
  - Transmit only if "louder" than recent receive window
- Audio compression
  - Provide coding field in packet

Basics: Camera



- Capture real time camera input in digital form
  - Still images
  - Video stream
- Render images to display
- Render mpeg
- Packetize and transmit
- Receive and render
- Collect and store
- Issues
- Data sheets
- Standards / Protocols

Options

- Share tones, songs
- Audio effects
- Integrate game
- Images
- Video
- Synthesizer capabilities
- SMS text

Getting from here to there (1/3)

- Week 6 – Lab 5: Network Digital Audio (Udam)
  - Spool winamp stream from ethernet to audio codec
  - Given ethernet black box and AC97 black box, build connections to Asynch FIFO
  - Key Learnings:
    - » mediating two peripheral clock domains and associated protocols through a synchronous intermediate
    - » Timing
  - Tools: Chipscope.
- Week 7 – CP 1: RT audio record and replay (Udam)
  - Audio capture on button press from Mic to RAM.
    - » Light LED when speaking is active
  - Audio play on button press from RAM to speaker
  - Key learnings:
    - » Understanding a synchronous serial protocol
      - Happens to be a weird slotted protocol
    - » Digitization of an analog signal
      - Can we look at the data?
    - » Simple digital signal processing
      - Detect start of speaking. Detect silence. Track signal energy.
      - Packetization

Getting from here to there (2/3)

- Week 8 – CP 2: Display (Allen)
  - Render canned source to video using Block SRAM
    - » Simpler than SDRAM frame buffer
  - Build basic display capability
  - Key learnings
    - » NTSC, ITU, Display representation
    - » Synching with external source
- Week 9-10 CP 3: Wireless (Shah, Ofer)
  - Stream RT audio to and from 15.4 radio
  - Key learnings
    - » SPI protocol, wireless MAC, nasties of wireless, interplay of CBR (audio) and asynchrony (network), bandwidth management

Getting from here to there (3/3)

- **Week 11 CP4: Basic i50phone (Shauki)**
  - Wireless audio 2-way line
  - Integration of many subsystems through registers, queues, memory data structures, and state machines
  - Basic display and functionality
  - Starts after Midterm II and checked off before Tday
- **Week 12-13: i50phone+ (Sarah)**
  - Select option that you will implement
  - SDRAM Storage (Allen)
    - » Stored audio streams
    - » SDRAM controller and memory arbiter
  - Key learnings
    - Bus protocols, memory, implementing complex sequencing in FSMs
- **Week 14: Final i50phone+ Project Checkoff**
- **Week 15: Writeup the Report**