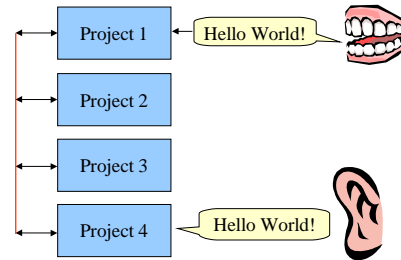


CS150 Fall 2000 Project Info

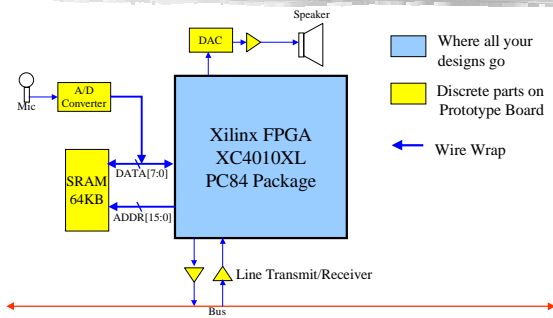
By: Nan (Norman) Zhou

Voice Over a Shared Network



CS 150 - Fall 2000 - Introduction - 2

Board Level Block Diagram



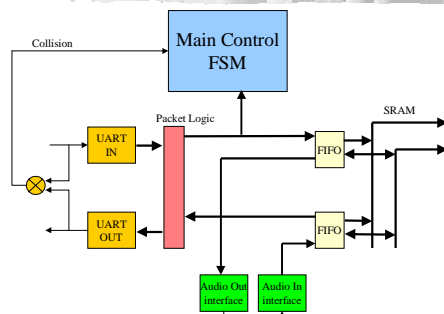
CS 150 - Fall 2000 - Introduction - 3

High Level Ideal

- Start With Digitized Voice Data from A/D converter
- Transmit Data in Packets Over the Shared Bus
 - ┆ Listen for traffic
 - ┆ Send only if there is no traffic on the bus
 - ┆ Listen for collisions and retransmit if necessary
- Receive Data Packets Over the Shared Bus
- Send Received Data to D/A converter to Recreate Sound
- Buffer Incoming and Outgoing Data in SRAM

CS 150 - Fall 2000 - Introduction - 4

Inside the FPGA



CS 150 - Fall 2000 - Introduction - 5

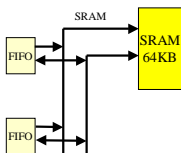
Project Break Down

- Complete project in pieces/checkpoints
- Check Point 1
 - ┆ SRAM Interface
 - ┆ FIFO
- Check Point 2
 - ┆ UART Transmitter Receiver
- Check Point 3
 - ┆ Audio Interfaces
- Check Point 4
 - ┆ Preliminary Global Data-path and Control
- Finished Project

CS 150 - Fall 2000 - Introduction - 6

Check Point 1

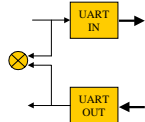
- **SRAM interface.**
 - ┆ Generate Control signals (OE CE WE) for Asynchronous SRAM
- **FIFO**
 - ┆ Generate Pointer Addresses For Head/Tail of ring Buffer
 - ┆ Keep track of Buffer status (full, empty, half full)



CS 150 - Fall 2000 - Introduction - 7

Check Point 2

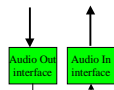
- **UART Transmitter**
 - ┆ Takes Parallel Data Input
 - ┆ Converts to Serial
 - ┆ Outputs Serial Data to Bus
 - ┆ Look for Collision and Abort Transmission When Appropriate
- **UART Receiver**
 - ┆ Reads Serial Data from Bus
 - ┆ Converts from Serial to Parallel
 - ┆ Sends Data to Rest of the Design
 - ┆ * Must Super Sample Data From Asynchronous Bus



CS 150 - Fall 2000 - Introduction - 8

Check Point 3

- **Audio In from A/D**
 - ┆ Expand Data Path to Take Data From A/D
 - ┆ Write Data in FIFO Buffer in SRAM
 - ┆ Shares Data line with SRAM
- **Audio Out to D/A**
 - ┆ Take Data from FIFO Buffer
 - ┆ Convert Data to Serial Form
 - ┆ Output to D/A chip



CS 150 - Fall 2000 - Introduction - 9

Check Point 3, comments

- Please read the Specs for the A/D, D/A chips
- Use the Pulse generator to generate input for test.
 - ┆ MAKE SURE you first measure the output Peak to Peak voltage before you hook it up to the A/D. it should be 2 V
 - ┆ Also try to use a audible frequency much lower than the sampling rate. 400Hz or so should be good.
- Keep you chips close together! Save space!
 - ┆ There are more chips on the way
- Electrolytic Capacitors have a specific pin for ground and a different one for Vcc

CS 150 - Fall 2000 - Introduction - 10

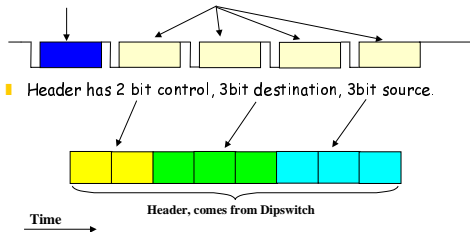
Check Point 4

- **Packet Logic**
 - ┆ Receive Logic
 - ┆ Take Data From UART
 - ┆ Reconstruct Packet
 - ┆ Interpret Packet and Take Appropriate Action
 - ┆ Send Logic
 - ┆ Take Data from SRAM
 - ┆ Construct Packet
 - ┆ Send to UART
- You are expected to show successful transmission of a packet for this checkpoint.

CS 150 - Fall 2000 - Introduction - 11

What is a Packet?

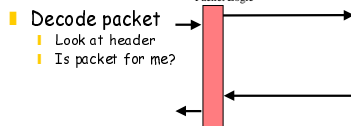
- 5 bytes of UART transmission
- 1 header, and 4 data samples



CS 150 - Fall 2000 - Introduction - 12

Packet Logic

- Construct Packet from byte input, coming from either UART in or FIFO (SRAM)
- Send Packet byte at a time to UART out or FIFO (SRAM)



CS 150 - Fall 2000 - Introduction - 13

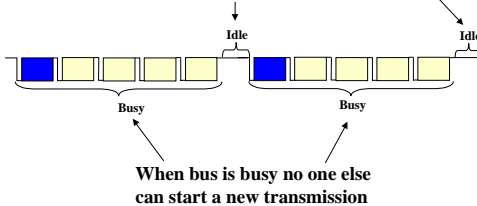
Network Protocol

- Always listen on the shared bus. (everyone picks up everything on the line)
- Filter out packets that does not have you as a source in the packet logic.
- Transmission policy (greatly reduces collision frequency)
 - Before you transmit first check Packet Logic and UART to see if bus is being used.
 - If bus is not used then wait a random amount of time before you start transmission
 - If 2 stations happens to decide to start at the same time, they collide, the station with the first 1 in the transmission loses and backs off.
 - To tell collision check to see if output is same as input on the shared bus.

CS 150 - Fall 2000 - Introduction - 14

Network Protocol

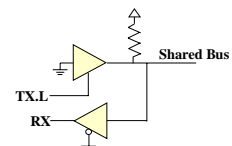
Collision can only occur when more than one station wants to transmit when the bus is idle. Even then it only happens when they both decide to start at the same time



CS 150 - Fall 2000 - Introduction - 15

What actually happens in a Collision?

- When TX is low the line is Driven low.
- However when TX is high the Line is left floating, and the resistor pulls the line high
- This means that during a collision whoever is sending a zero always wins.
- The winner continues to send as if nothing ever happened
- The loser detects that the 1 it's trying to send is not being sent on the wire. So it stops immediately and tries later.



CS 150 - Fall 2000 - Introduction - 16

Trace of data flow in Final project

- Analog input voltage is converted to 8bit unsigned #
- This 8bit DATA is written directly into the SRAM through the shared data line of A/D and SRAM
 - this is done by asserting the WE of SRAM during the time when RD is asserted for the A/D chip
 - The Address for the write is provided by the FIFO for the out going data.
- Data is read out of the SRAM 4 bytes at a time and put into the packet logic. A header byte is also added by taking input from the DIP SWITCH

CS 150 - Fall 2000 - Introduction - 17

Trace of data flow in Final project

- The packet is sent to the UART out one byte at a time. If the bus is idle and there is no collision
- DATA travels bit serially to another station and gets picked up by UART in.
- The UART in then sends the data to the Packet Logic one byte at a time
- Packet Logic decodes the packet
 - If the destination matches the station then send the 4 data bytes to the SRAM one at a time.
 - The OUTPUT FIFO generates the address

CS 150 - Fall 2000 - Introduction - 18

Trace of data flow in Final project

- **Read DATA from SRAM**
 - Convert from 8bit unsigned to 16bit 2's compliment (note I did a lousy job of that from the schematic I gave you guys for checkpoint 3. I expect better from you for the final project)
- **Send the converted data to the D/A**
 - (and hopefully voice comes out)

CS 150 - Fall 2000 - Introduction - 19

Miscellaneous info

- **Max bus bit rate, 250Kbps**
- **Super sampled clock rate 2MHz**
- **Voice sampling, 8bit, 2KHz**
 - $8 \times 2K = 16Kbps$ pure data rate
- **Network Bandwidth taken per voice stream**
 - Pure data rate plus...
 - UART overhead 2bits per 8 data bits
 - Packet 1 header byte per 4 data bytes
 - Idle time overhead for random wait time after bus idle
- **Many shared resources!** Think hard about how to share them.

CS 150 - Fall 2000 - Introduction - 20

Happy Halloween!

CS 150 - Fall 2000 - Introduction - 21