Design and Verification of a Complete Application Specific IC

Members: Julie Swift, Zachary Nelson*,
Dhruvit Naik, Kevin Cao and Whitley Forman
Advisors: Dr. Larry Pearlstein and
Dr. Orlando Hernandez

* Team Leader

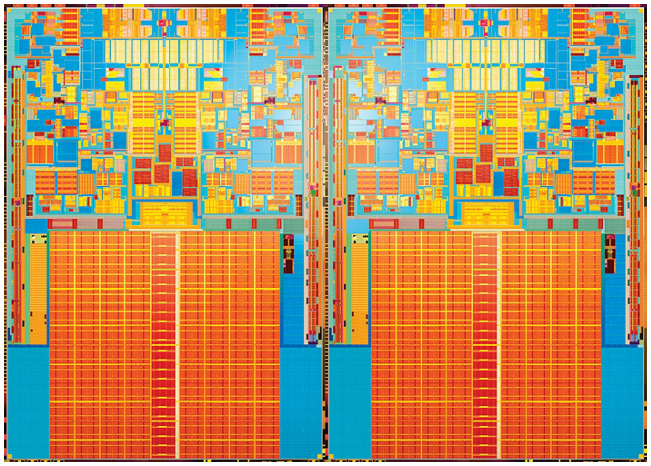
TCNJ THE COLLEGE OF NEW JERSEY

Agenda

- Project Goals
- Open Action Items
- Detailed Specifications
- Project Status:
  - Schedule
  - Work Plan/Tasks
  - Task Details
- Budget
- Summary

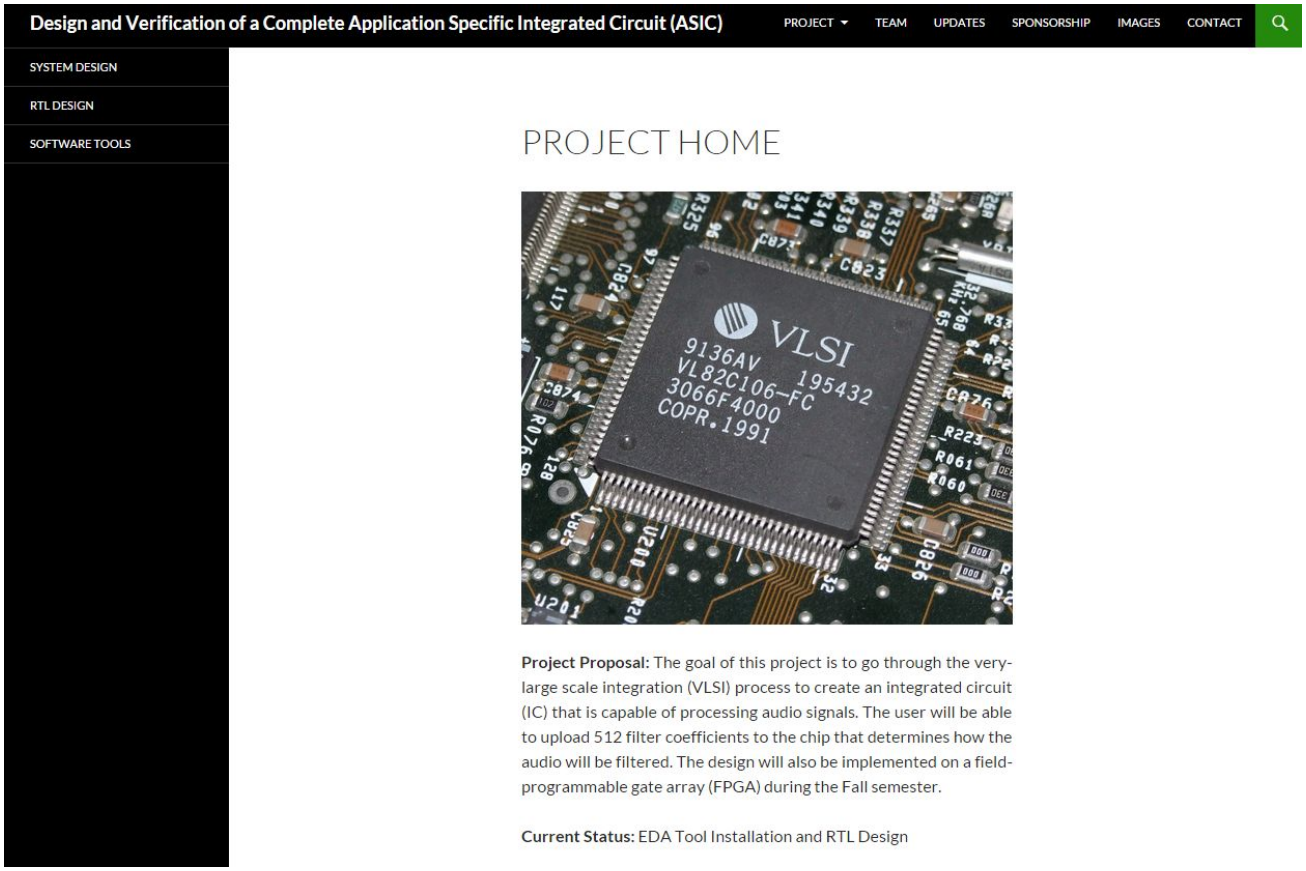
Project Goals

- VLSI Design of an ASIC used for audio processing
- Why are we doing this project?
  - Not common at undergraduate level because of high cost
  - MOSIS Educational Program (MEP)
- ASIC versus FPGA
- Why audio processing as our application?
  - Complex and interesting
  - Simple enough so the project can be completed



Website

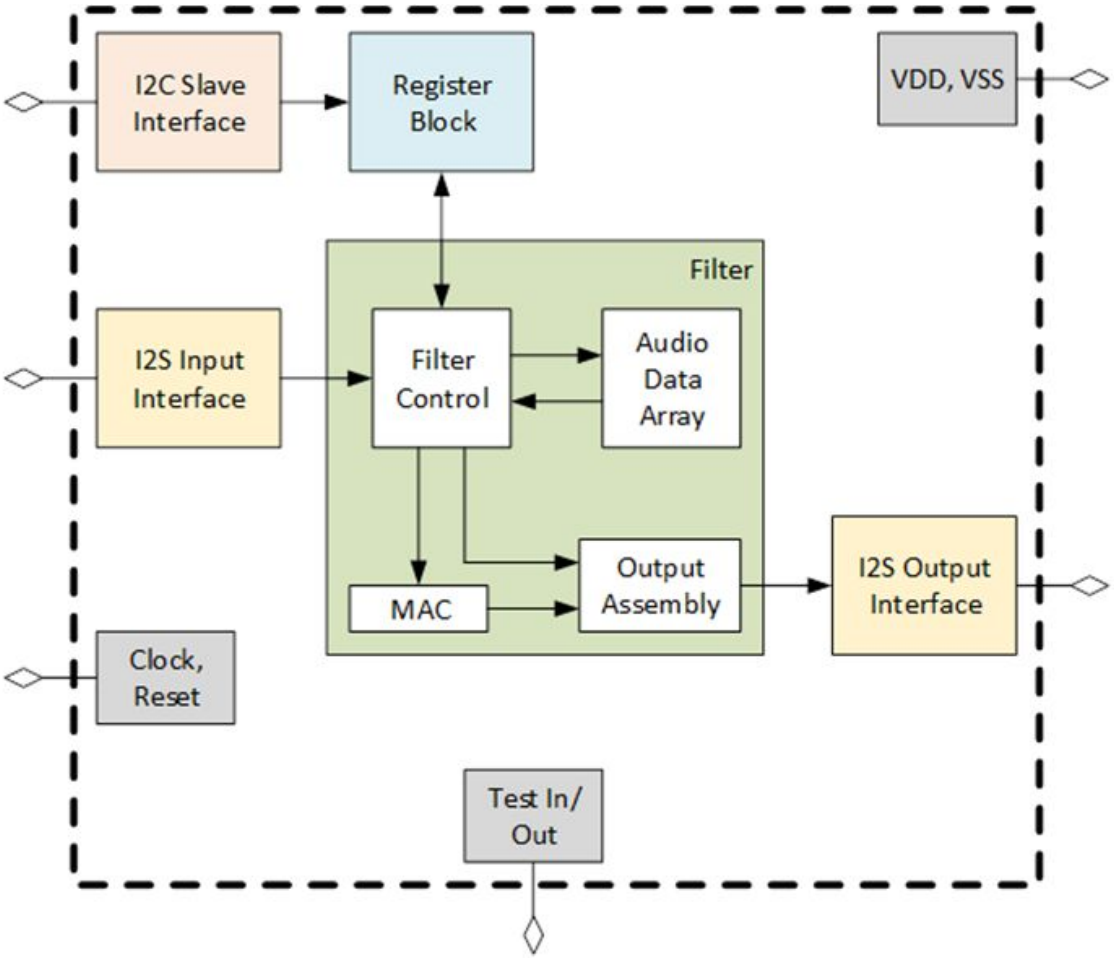
- http://tcnjchip.pages.tcnj.edu/



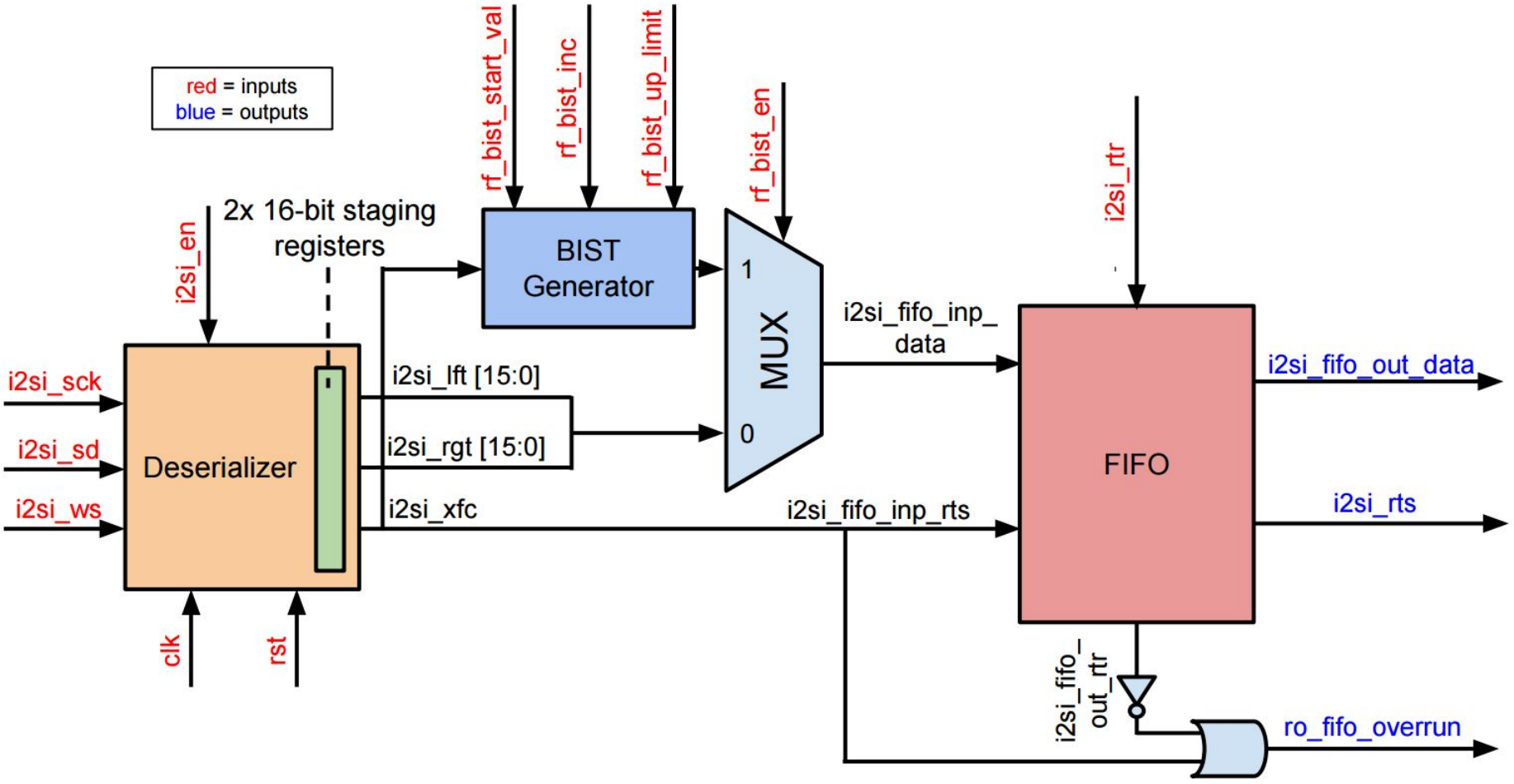
Open Action Items

- Downloading the software for Mentor Graphics
  - Assigned to Dhruvit
  - Install Pyxis Place and Route Tool
  - We will evaluate if Mentor Graphics will be sufficient
  - We will consider Cadence and Synopsys if not

Chip Overview

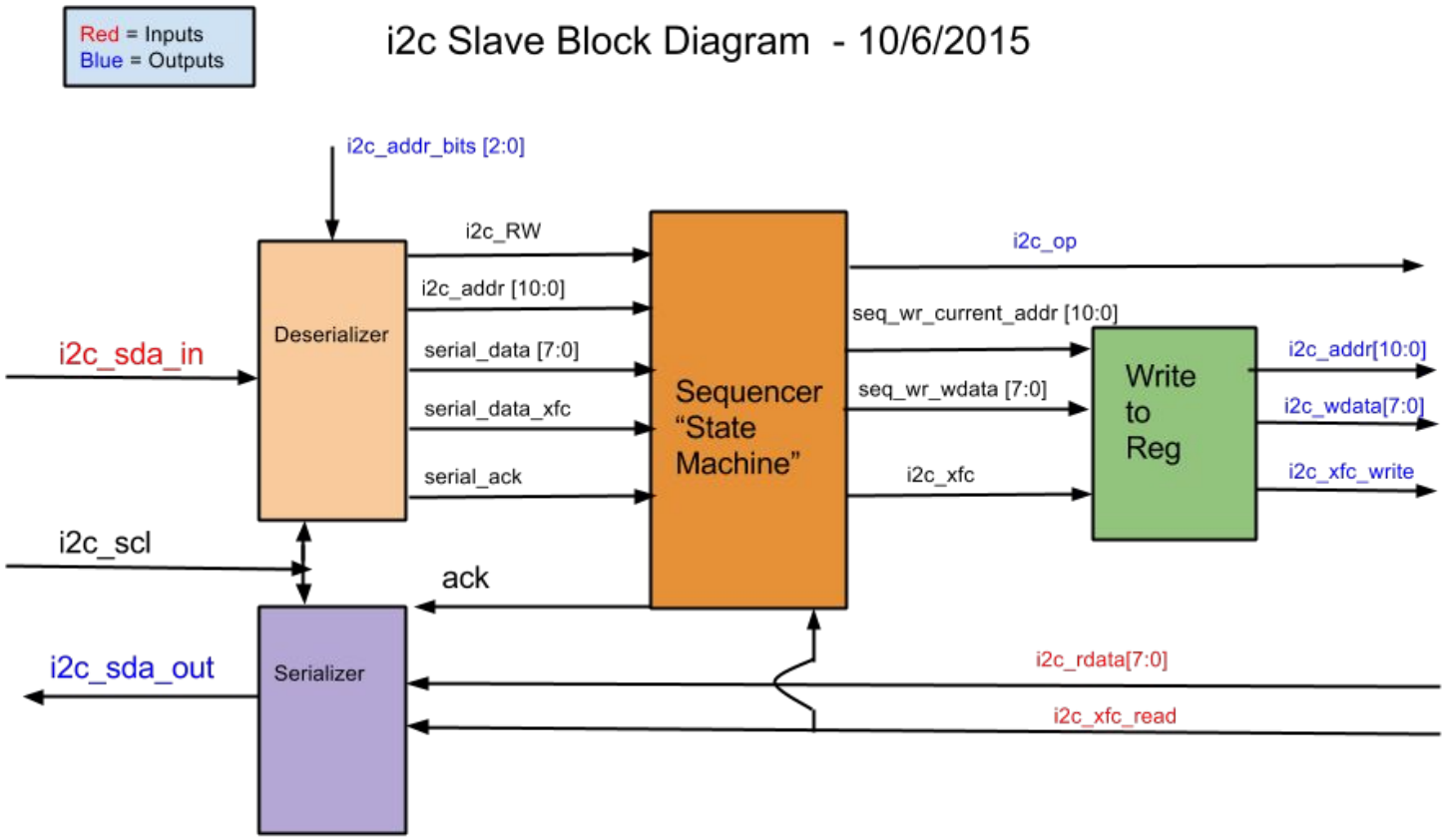


Detail Specifications: I2S



TCNJ THE COLLEGE OF NEW JERSEY

Detail Specifications: I2C_SLAVE



Detail Specifications: REGISTER

Address	Register Name	Field Name	Bits	Description	RO/WO/RW	Default Value
0x000	CHIP_INFO					
		ro_chip_id	7:0	Fixed Chip ID	RO	0x1234
		ro_revision_id	15:8	Fixed Revision ID	RO	0
0x004	CONTROL					
		rf_soft_reset	0:0	0- normal operation. 1- assert soft reset	RW	0
		rf_i2si_bist_en	1:1	0- audio sorce is i2si. 1- audio source is BIST	RW	0x1
		rf_filter_shift	5:2	number of bit postions to shift after filter accumulator	RW	0xF
		rf_filter_clip_en	6:6	1- performs clipping 0- no cliping	RW	0x1
	optional?	rf_i2si_dec_factor	10:7	sample and hold audio values	RW	0
	optional?	rf_i2so_dec_factor	14:11	sample and hold audio values	RW	0
0x008	I2S_CLOCK_CONTROL					
		rf_i2so_clk2sck_div	16:0	half of the clock frequency divided by this #	RW	0x40
0x00C	STATUS					
		trig_fifo_overrun	0:0	fifo overrun clear	WO	NA
		ro_fifo_overrun	1:1	input audio fifo overrun	RO	0
		trig_fifo_underrun	2:2	fifo underrun clear	WO	NA
		ro_fifo_underrun	3:3	output audio fifo underrun	RO	0
0x010	BIST					
		rf_i2si_bist_start_val	12:0	start value of sawtooth wave	RW	0x800
		rf_i2si_bist_incr	20:13	increment of sawtooth wave	RW	0x010
		rf_i2si_bist_upper_limit	32:21	upper limit of the sawtooth wave	RW	0x7FF
0x014	I2C_REG_INDIR_ADDR					
		rf_i2c_reg_indir_addr	11:0	address register used for indirect addressing via i2c	RW	0
0x018	I2C_REG_INDIR_DATA					
		ro_i2c_reg_indir_data	8:0	data access register	RO	NA
0x040	FILT_COEFFS_0_1					
		rf_filter_coeef0	15:0	Filter Coefficient 0	RW	0x0
		rf_filter_coeef1	31:16	Filter Coefficient 1	RW	0x0
0x044	FILT_COEFFS_2_3					
		rf_filter_coeef2	15:0	Filter Coefficient 2	RW	0x0
		rf_filter_coeef3	31:16	Filter Coefficient 3	RW	0x0
0x43C	FILT_COEFFS_510_511					
		rf_filter_coeef510	15:0	Filter Coefficient 510	RW	0x0
		rf_filter_coeef511	31:16	Filter Coefficient 511	RW	0x0

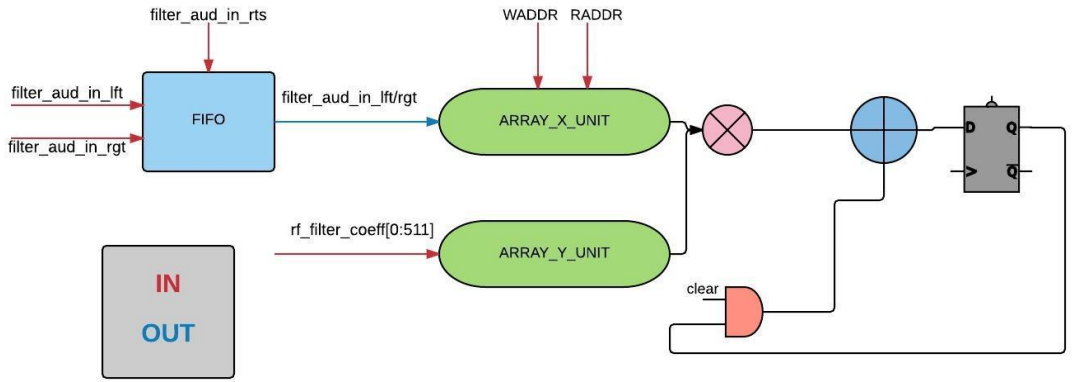
Detail Specifications: FILTER

Stage 1: Transfer

When: Ready to Send and Ready to Receive are both asserted

What: Data goes into the FIFO and into a Storage Module

Transition: Once there is data in the Storage Module it will transition



Detail Specifications: FILTER

Stage 2: Convolution

When: There is data in the Storage Module

What:

First: Data from Storage Module(X-Unit) is Multiplied by Values from the Register(H-Unit)

Second: Result is Accumulated

Transition: When all the data is processed

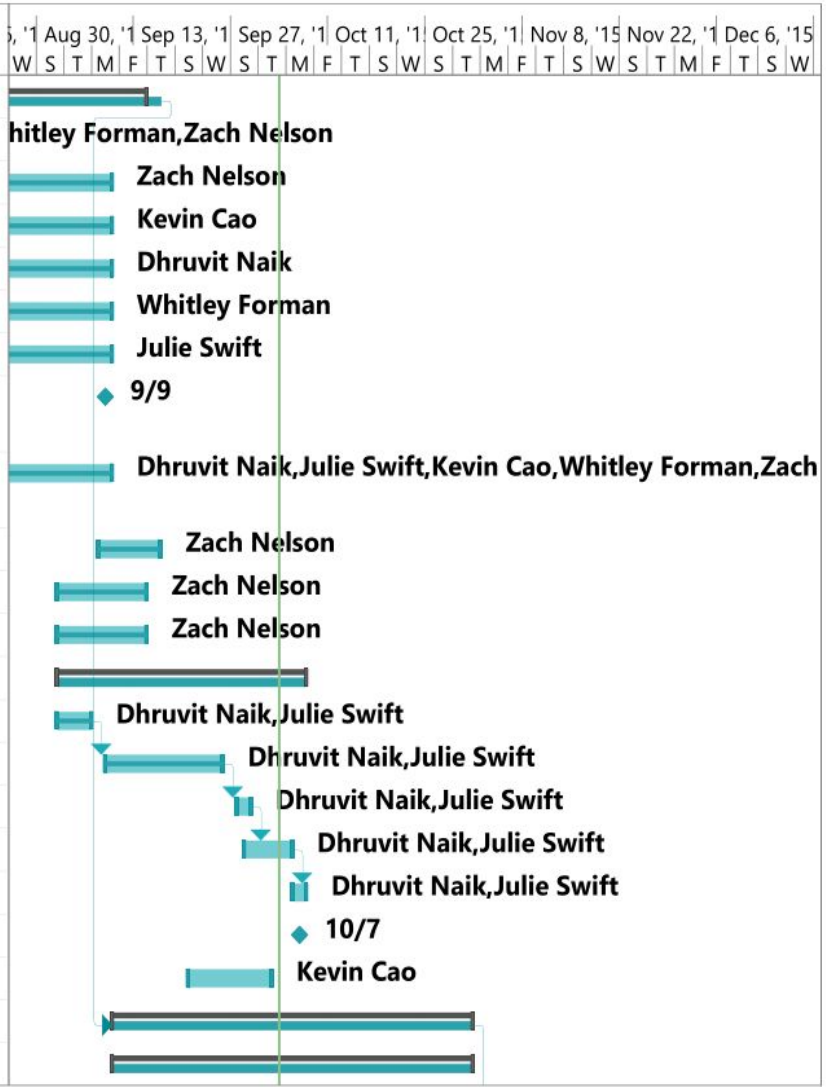
$$\sum_{k=-\infty}^{+\infty} x[k]h[n-k]$$

Software Tools

- GitHub (Version Control)
- DropBox (Confidential Files)
- CORE 9 (System Design)
- Microsoft Project (Project Management)
- Xilinx ISE Design Suite
- Mentor Graphics*
- Linux RedHat Operating System

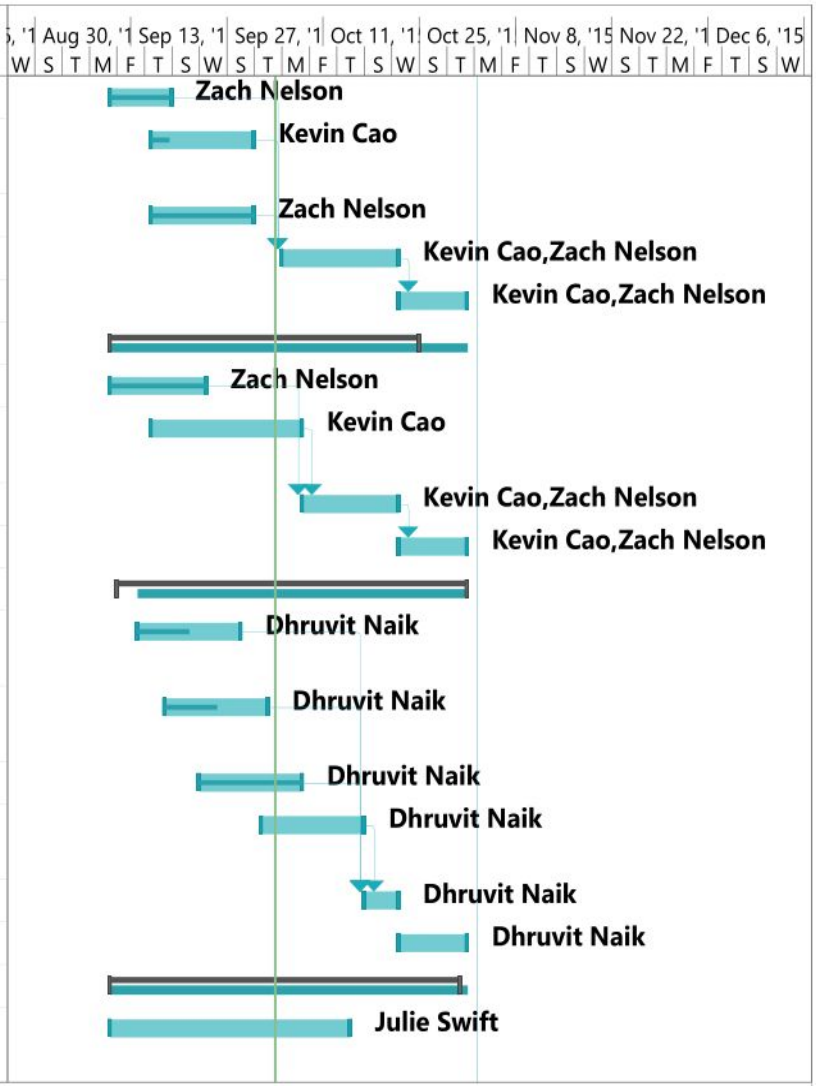
Project Status: Schedule

ID	Task Mode	Task Name	Duration	Start
1		**System Design**	**88 days**	**Thu 5/14/15**
2		General Planning	26 days	Thu 5/14/15
3		Complete I2S In Document	60 days	Thu 6/18/15
4		Complete I2S Out Document	44 days	Sat 7/11/15
5		Complete Filter Document	60 days	Thu 6/18/15
6		Complete I2C Document	60 days	Thu 6/18/15
7		Complete Register Document	60 days	Thu 6/18/15
8		Finish System Design	0 days	Wed 9/9/15
9		Document System in CORE	60 days	Thu 6/18/15
10		RTS/RTR and XFC Protocols	7 days	Tue 9/8/15
11		**Create a Detailed Schedule**	**9 days**	**Wed 9/2/15**
12		**Install Microsoft Project**	**9 days**	**Wed 9/2/15**
13		**Install EDA Tools**	**26 days**	**Wed 9/2/15**
14		Determine EDA Requriments	4 days	Wed 9/2/15
15		Install OS	13 days	Wed 9/9/15
16		Install EDA Tool	2 days	Mon 9/28/15
17		Create a Simple Test Case	5 days	Tue 9/29/15
18		Place and Route Test	2 days	Tue 10/6/15
19		Finish EDA Installation	0 days	Wed 10/7/15
20		**Create Web Page**	**10 days**	**Mon 9/21/15**
21		**RTL Design and Testing**	**38 days**	**Thu 9/10/15**
22		**Create i2si.v**	**38 days**	**Thu 9/10/15**



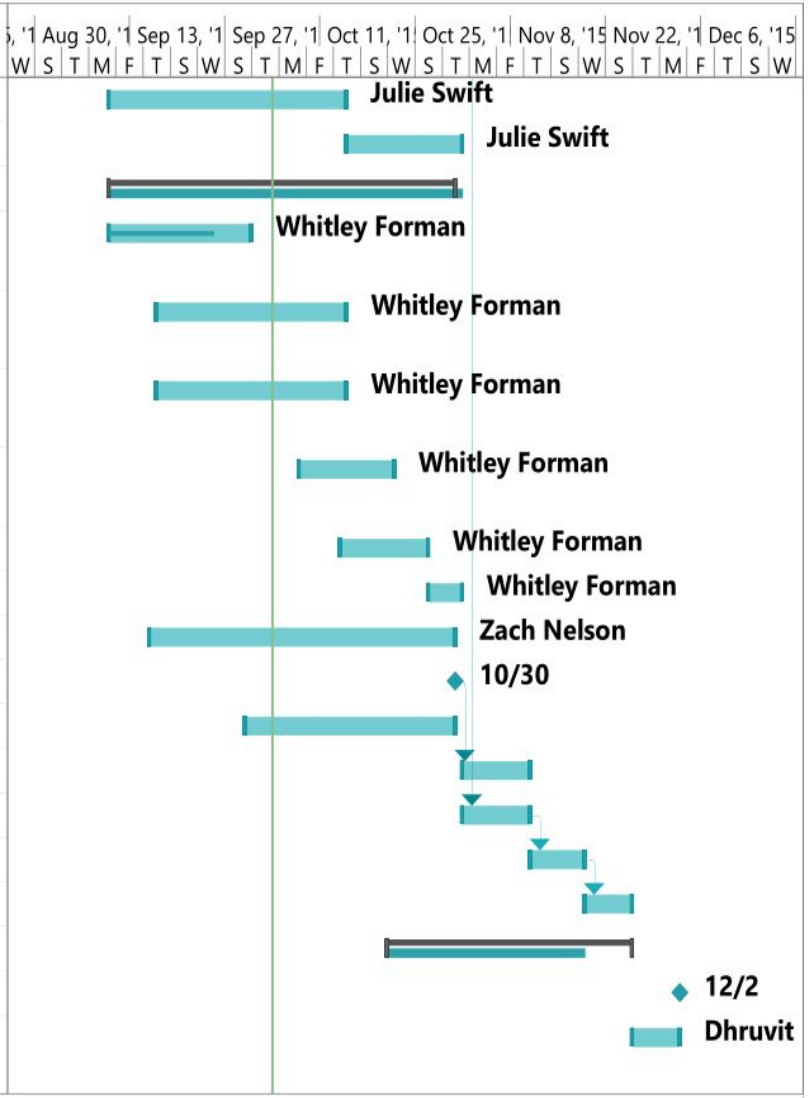
Project Status: Schedule

ID	Task Mode	Task Name	Duration	Start
23		Create and Test i2si_fifo.v	7 days	Thu 9/10/15
24		Create and Test i2si_deserializer.v	11 days	Wed 9/16/15
25		Create and Test i2si_bist.v	11 days	Wed 9/16/15
26		Create i2si.v Top-Level	13 days	Mon 10/5/15
27		Test i2si.v	8 days	Thu 10/22/15
28		**Create i2so.v**	**33 days**	**Thu 9/10/15**
29		Create and Test i2so_fifo.v	10 days	Thu 9/10/15
30		Create and Test i2so_serializer.v	16 days	Wed 9/16/15
31		Create i2so.v Top-Level	10 days	Thu 10/8/15
32		Test i2so.v	8 days	Thu 10/22/15
33		**Create filter.v**	**37 days**	**Fri 9/11/15**
34		Create and Test filter_convolution.v	11 days	Mon 9/14/15
35		Create and Test filter_accumulator.v	11 days	Fri 9/18/15
36		Create and Test filter_fifo.v	11 days	Wed 9/23/15
37		Create and Test filter_round_shift_clip.v	11 days	Fri 10/2/15
38		Create filter.v Top-Level	4 days	Sat 10/17/15
39		Test filter.v	8 days	Thu 10/22/15
40		**Create register.v**	**37 days**	**Thu 9/10/15**
41		Create and Test register_trig_gen.v	25 days	Thu 9/10/15

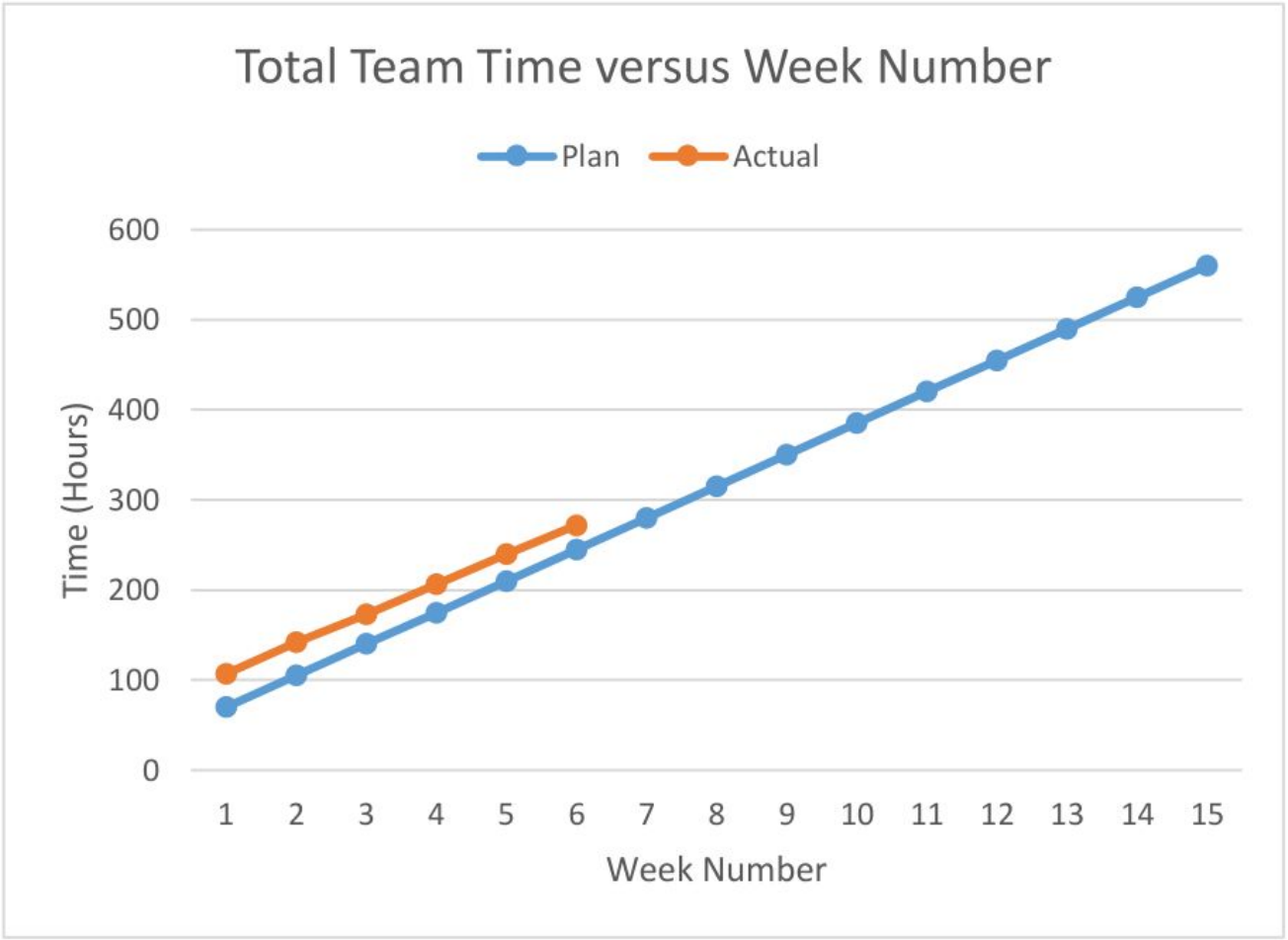


Project Status: Schedule

ID	Task Mode	Task Name	Duration	Start
42		Create register.v Top-Level	25 days	Thu 9/10/15
43		Test register.v	13 days	Thu 10/15/15
44		**Create i2c_slave.v**	**37 days**	**Thu 9/10/15**
45		Create and Test i2c_slave_deserializer.v	15 days	Thu 9/10/15
46		Create and Test i2c_slave_sequencer.v	20 days	Thu 9/17/15
47		Create and Test i2c_slave_writetoreg.v	20 days	Thu 9/17/15
48		Create and Test i2c_slave_serializer.v	10 days	Thu 10/8/15
49		Create i2c_slave.v Top Level	9 days	Wed 10/14/15
50		Test i2c_slave.v	5 days	Tue 10/27/15
51		**Create chip.v**	**33 days**	**Wed 9/16/15**
52		**Finish RTL Design**	**0 days**	**Fri 10/30/15**
53		**Create Text Fixtures**	**23 days**	**Wed 9/30/15**
54		**Integration Testing**	**8 days**	**Sun 11/1/15**
55		**Logic Synthesis**	**8 days**	**Sun 11/1/15**
56		**Floor Planning**	**6 days**	**Wed 11/11/15**
57		**Place and Route Gates**	**5 days**	**Thu 11/19/15**
58		**Board Design**	**26 days**	**Wed 10/21/15**
62		**Have a Functional FPGA**	**0 days**	**Wed 12/2/15**
63		**Write Final Report and Presentation**	**5 days**	**Thu 11/26/15**



Time Schedule



Time Schedule

Name	Summer	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
Julie	10	5	10	6	5	4	5	7	7	7	7	7	7	7	7	7	108
Zach	64	12	10	10	6	6	6	7	7	7	7	7	7	7	7	7	177
Dhruvit	1	5	5	4	8	8	7	7	7	7	7	7	7	7	7	7	101
Kevin	2	2	5	5	7	8	6	7	7	7	7	7	7	7	7	7	98
Whitley	3	3	5	6	7	8	8	7	7	7	7	7	7	7	7	7	103
Total	80	27	35	31	33	34	32	35	35	35	35	35	35	35	35	35	587

Work Plan:

Task 1: System Design - (100% Complete)
Task 2: RTL Design - (20% Complete)
Task 3: Install EDA Tools - (30% Complete)
Task 4: Logic Synthesis - (0% Complete)
Task 5: DFT (Design for Testing) - (0% Complete)
Task 6: Gate Level Simulation - (0% Complete)
Task 7: Place and Route Gates - (0% Complete)
Task 8: Verification - (0% Complete)
Task 9: Implementation - (0% Complete)
Task 10: Documentation

Task 1: System Design (100% Completed)

- CORE - 100%
- Microsoft Project - 100%
- Block Design - 100%
  - Interfaces
  - Functional Requirements
  - Micro-Architecture
  - Design
  - Verification

Task 2: RTL Design (20% Done)

Julie:
- Trig_Generator - 20%

Zach:
- FIFO created and tested - 100%
- Bist_Generator created and tested - 100%

Kevin:
- Deserializer - 85%

Whitley:
- I2C Deserializer - 80%

Dhruvit:
- Filter Convolution Accumulator - 40%

Task 3: Install EDA Tools (30% Done)

- Installed Red Hat Linux
  - Armstrong Room 144B
- Mentor Graphics
  - Install Pyxis - Place & Route Tool
- Create a Test Circuit
  - Perform Place & Route

Tasks 4-8: Logic Synthesis, DFT, Gate Level Simulation, Place and Route, Verification (0% Complete)

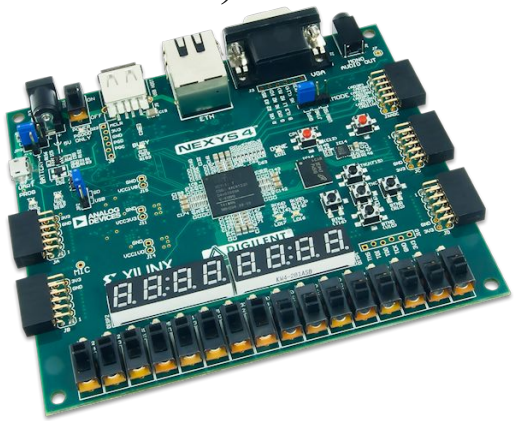
- Start after completion of RTL Design (end of October)
- Logic Synthesis can begin once EDA tools are installed
- DFT: Using BIST Generator and performing test insertion
- Gate Level Simulation: Search for startup and reset errors, glitches associated with combinational logic, clock cross domain issues, and X propagation

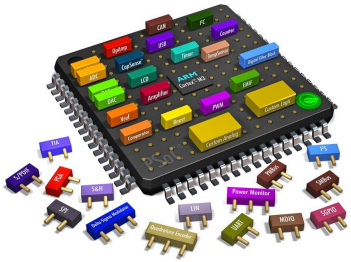
Tasks 4-8: Logic Synthesis, DFT, Gate Level Simulation, Place and Route, Verification (0% Complete)

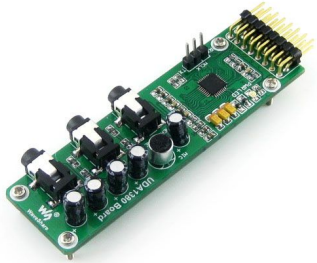
- Place and Route: Place electronic components, circuitry, and logic elements in limited space. Decide design of all wires needed to connect the placed components
- Verification
  - Develop a Test Plan (Word/CORE 9)
  - Develop RTL Test Environment (Verilog, C++)
  - Test Software (C++)
  - Test Vectors
  - Run and Debug RTL Tests
  - Gate Level Simulation (GLS)

Task 9: Implementation (0% Done)

- Planning to purchase a Nexys 4 Artix-7 FPGA Board
  - Meets clock speed specification @450Mhz (100Mhz needed)
  - Easy implementation of i2c DIP switches
  - Easy implementation of reset button
  - Meets memory size specification @ 16Mbits (~34 Kbits needed)
  - Under budget @ $180 ($500 Max)
- Planning to use a stock PSoC5LP from TCNJ
  - Will do i2c control of chip
  - Meets requirement of 100khz i2c bus speed
- Planning to purchase a UDA 1380 Chip
  - ADC for analog audio to i2s conversion
  - Meets specifications for i2s conversion
  - *Not directly matable to Nxys 4







Task 10: Documentation

- GitHub being used as repository for all documents
  - Has been shown to be useful for for easy one place sharing of materials
  - Some changes to design have required in person meetings for relevant parties to understand what the change is
- Testbench
  - Corner testing testbench files stored on Github
  - Corner Test results also stored on Github
- Dropbox
  - Storing all IBM confidential information
- Core 9
  - Tracking all requirements of the system

Budget

Item	Cost
Electronic Design Automation Tools	$0.00
ISE Design Suite 14.7	$0.00
CORE 9 University	$0.00
MOSIS Fabrication	$0.00
Crystal Oscillators	$20.00
Field Programmable Gate Array (FPGA)	$180.00
UDA 1380 Board	$20.00
CY8CKIT-050 PSoC 5LP Development Kit	$0.00
Total	$220.00

Summary/Conclusion

- Design Specifications and requirements are finished
- RTL Synthesis is still ongoing
  - This is the Bulk of our work this semester
- Software tools are being installed
  - Mentor Graphics is to be evaluated
- Hardware has been selected
  - Has yet to be purchased
- We are on target with the schedule besides small delays and trouble with RTL modules
- We are well under budget
- Documentation continues