EE175ABC Final Report

High-speed interconnection Simulation

EE 175AB Final Report

Department of Electrical Engineering, UC Riverside

Project Team Member(s)	Meng Lee Yuan Yao
Date Submitted	06/07/2015
Section Professor	Sheldon Tan Ping Liang
Revision	Revision v5.0
URL of Project	https://drive.google.com/folderview?id=0B15mlkxHMyqLflA5YTJLckdHRnh0LTlDaFZHTjE3T1dWXzdLNy1ZaGpuRG5DcWxLZHRtTTQ&usp=sharing

Summary

This report presents a detailed and technical overview on the senior design project: High-speed interconnection Simulation.

Revisions

Version	Description of Version	Author(s)	Date Completed	Approval
1.0	First draft	Meng Lee Yuan Yao	05/04/2015	
2.0	Second edition	Meng Lee Yuan Yao	05/21/2015	
3.0	Third edition - completed test results, conclusion, high-level, low-level, and test plan.	Meng Lee	05/24/2015	
4.0	Final draft - Completed Essay questions	Meng Lee Yuan Yao	05/25/2015	
5.0	Final edition – fixed typos	Meng Lee Yuan Yao	06/07/2015	

Table of Contents

1 * Executive Summary

High-speed signal processing in the industry relies heavily on channel simulation software tool to generate and design for robust high-speed communication channels. The most common tool used in this field is SerDes channel simulation with IBIS AMI model from Advanced Design System (ADS). However, while these are available options for industry usage, they are in need of improvement on performance accuracy. Therefore, our goal of this project is to develop a basic high-speed channel simulation tool with more accurate channel simulation.

For this purpose, Matlab is chosen to be the coding environment for its conveniently built-in function of matrix algorithms. The program contains three key features:

- Scattering Matrix Concatenation
- Obtaining Transfer Function
- Data Simulating using Input/output Buffer Information Specification (IBIS) AMI Models

The program is tested using the following software programs:

- Matlab RF tool
- Advance Design System

The most significant achievement of this project is to be able to completely developing the software package for the channel simulation tool in Matlab and obtained some numerical results. Throughout this project, we encountered countless amount of difficulties, which were resolved in the end by our hard works.

2 * Introduction

2.1 * Design Objectives and System Overview

The concept of our program is to perform a complete channel simulation with arbitrary input data (bits) generated within Matlab (by designers). Intended usage of this program is for high-speed circuit and system designers, who need to perform the simulation and analysis of the high-speed circuits before the production of the designs.

The program three major components:
-Concatenation for multiple channel models
-Derivations for channel impulse response
-Data bits simulation with IBIS AMI model

The system platform is Matlab. User must have a 32-bit Matlab installed on a computer. External systems include the usage of AMI model, which is in a direct link library package.

There is some system issue while using AMI_funciton. This issue is known to crash Matlab software after a second run on this program. This issue stems from the channel model problem. We are unable to solve this problem because it is the AMI files' issue.

Technical Design principle is to have at least a 14-digit accuracy comparing to testing software.
In our program:
-Concatenation must be compared to Matlab RF tool result
-Transfer function must be compared to ADS generated result
-Channel simulation should give the final eye-diagrams, which agrees with the ADS simulation and measured results (but this part is beyond the scope of this senior design).

Concatenation is developed by team member Meng Lee
Transfer function is developed by team member Yuan Yao
Data simulation is developed by both team members.

2.2 * Backgrounds and Prior Art

- Serializer/Deserializer (SerDes)
  SPICE-Based Analog Simulation Program: TINA-TI

  TINA-TI is a powerful circuit design and simulation tool. TINA-TI is ideal for designing, testing, and troubleshooting a broad variety of basic and advanced circuits, including complex architectures, without any node or number of device limitations. (The references are listed in section 9)

- Advanced Design System (ADS)
  Advanced Design System is the world's leading electronic design automation software for RF, microwave, and high speed digital applications. In a powerful and easy-to-use interface, ADS pioneers the most innovative and commercially successful technologies, such as X-parameters* and 3D EM simulators, used by leading companies in the wireless communication & networking and aerospace & defense industries. For WiMAX™, LTE, multi-gigabit per second data links, radar, & satellite applications, ADS provides full, standards-based design and verification with Wireless Libraries and circuit-system-EM co-simulation in an integrated platform. (The references are listed in section 9)

The advantages of these products:

1. They all have a nice user interface;
2. They can let user design their own channels;
3. They have much more functions rather than just simulate the channel;
4. They can fit in many situations.

The advantages of our project:

1. We take in account of reference and terminal impedances;
2. Our users only need to install MATLAB 32bit to run our program;
3. We have a clear and concise goal.

2.3 * Development Environment and Tools

- Design environment:
  MATLAB R2012a 32bit
- Testing equipment:
  Advanced Design System (ADS)
  Matlab RF tool

2.4 * Related Documents and Supporting Materials

Supporting Materials:

IBIS (I/O Buffer Information Specification) Version 5.1

The development environment for our program is a computer with 32 bit Matlab installed. Computer does not have to be 32 bit windows.

Coding environment and software tools include:

- 32 bit Matlab r2012a for coding
- Matlab RF tool for verification and testing
- Advanced Design System for assist on testing

2.5 * Definitions and Acronyms

IBIS AMI - Input /Output Buffer Information Specification Arithmetic Modeling Interface

Serdes model - Serializer/deserializer model

3 * Design Considerations

3.1 * Assumptions

- The program is to be used on a computer with products later than Windows 2003 R2 Service Pack 2 with 32 bit Matlab installs.
- Users of the program are assumed to be channel design engineers who have sources for channel model files.

3.2 * Realistic Constraints

3.2.1 Matlab Memory Size issue

A major part of our program is to generate data bits for channel simulation. In our program, we must generate at least ten thousand bits. For each bit, we take 64 samples. The resulting sampling wave form is very large in terms of size. When we try to generate data bits more than a hundred thousand, Matlab program is unable to complete the task. The program results in error in convolution function because the data size is too large. Therefore, the program runs out of memory.

3.2.2 Matlab and AMI Compatibility Issue

Moreover, there is a compatibility issue between Matlab and the AMI model functions. After running our code once, the program does not terminate our code correct because the AMI model files do not contain the function AMI_close(). Result is a crash on the second run of the program. However, this issue may not be critical because this technology is fairly new. And we can only get the existing model files that appear to be defective. In the near future, this technology will become much more common. Therefore, there should be no error in using users new model files.

3.3 * Industry Standards

Matlab maximum matrix array size - Maximum possible array size 11862MB

We are not able to simulate more than 100 thousand bits using convolution function because of this standard.

3.4 * Knowledge and Skills

Describe the knowledge and skills required by this project. Complete the following for each team member:

List all the Electrical Engineering and other technical courses (e.g., Computer science or upper division physics) you took or are taking that are related to your project.

List any new knowledge and skills that you had to learn while working on the project in order to complete it.

Member:

Meng Lee:

The knowledge and skills require by this project are extremely broad. Some areas are what we have learned in courses in UCR. Other areas are what we learn hands on.

For this project, the following courses have help accomplish it:

- EE150 Digital Communication

  - Introduces concepts on transmitting digital data. For example, the data bits we generate in Matlab are NRZ convention.

- EE141 Digital Signal Processing

  - Introduces Discrete Fourier Transform. Used to obtain channel impulse response

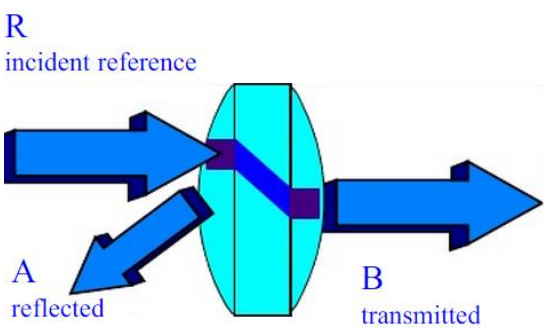
- Matlab Computing courses: EE110, EE115
- Software courses: CS10, CS13, CS61, EE/CS 120, EE128

New knowledge learned through the course of this project:

- Transmission line theory
- Matrix, for specifically, scattering matrix and through matrix

  -knowing the fundamentals of network matrix is extremely important. Complex derivations cannot be derived if I cannot understand the fundamentals.

  The figure below is the simple explanation on how scattering matrix is derived



Yuan Yao:

After this project, I have earned a lot of knowledge that isn't covered in class. For example, during the coding in MATLAB, I searched a lot on Internet to find the proper way to meet the requirement of out project. The most valuable thing I learned from this project is that I have earned some industrial experience. Our project is a newly industrial demand so that we are in the start of this field. Therefore, we have encountered many unexpected problems during the project. The channel model we use is a 32-bit dll library that contains all of the data and functions we need in the simulation. However, the model is a newly coming out channel model and the AMI_INIT function in the model deals with the pointer improperly so that we must do some trick on it in order to make our project work fine. These researches help to build up my own studying system and do great contribution to my future study in both Computer Science and Electrical Engineering.

The courses I took from UCR that are related to our project:

- EE115 Intro to Communication Systems

- Introduces signals and systems that we are used to obtain transfer function and channel impulse response.

- EE150 Digital Communication

  - Introduces digital circuits and digital encoding. We are using NRZ convention to generate our input data bits.

- EE141 Digital Signal Processing

  - Introduces Discrete Fourier Transform that we are used to obtain channel impulse response.

- Computer courses: CS010, CS012, CS100

3.5 * Budget and Cost Analysis

Present your budget and/or cost analysis.

The budget of this project is entirely funded by the University of California, Riverside providing downloadable access to 32 bit Matlab. Although this project does not physical cost anything, the developmental time and effort on this project is extremely large. My team spends at least 2 hours a day understanding and developing for this project.

3.6 * Safety

Because our simulation system is software based, there is no immediate thread or harm against human bodies. However, there are risks of overloading a computer when too much data are requested to be simulated.

3.7 * Documentation

The processes for generating and maintaining technical and user documentation is very systematic. Design notes and engineering change notices have to be documented in weekly report and discussed prior to version update. Once it is approved by team members and advisers, engineer can edit the user document. If the engineering change notice covers more area than 3 sections, new update has to be released while keeping the old version file for reference.

3.8 * Understanding of Professional and Ethical Responsibility

Ethical responsibility is a very import area that we need to learn before we can be called engineers. With every designed product, it comes with professional and ethical responsibility. There are many aspects in the perspectives of professional and ethical responsibilities that we may encounter if our design becomes a service. These aspects include:

- Integrity
- Competence

Integrity - Integrity comes from honesty, we must not deceive others especially our teammates or clients. Our program will most likely become a service program for engineers who wish to design and simulate their channels. And our program is to be a very convenient service for them. We must address any question that these engineers have with honesty. For example, in the future, if our product has a critical error that is caused by our team's design, we must address this issue directly instead of trying to cover up such mistake.

Throughout this project, we learned to be honest to team members. It is most important to let team members know the project at the same level as ourselves. Therefore, we cannot lie about any of our own mistakes when debugging. We learned that it is best to pause and solve the problem as a team, rather than go off individually when one member tries to solve a problem by himself, while the other member just mindlessly continue on with the project with potential error.

Competence - Competence is to maintain the knowledge and skills to provide professional service. As mentioned in integrity, our program will most likely become a service for engineers. Therefore, to be useful and able to provide service to our client engineers, we must be competent in our own field of study. In this case, we must know every little bit of detail on the simulation program to make sure we are able to provide efficient assistance to our clients. Moreover, if our product becomes a commercial product, many improvement and updates are definitely going to happen to improve our design. We must keep up with time and continue learning new knowledge to provide service.

Through this project, we are able to learn the aspect of competence through our adviser James Zhou. Before we step in any part of the project, we make sure that we know what we are dealing with first. However, we still made many mistakes because the subject is foreign to both of us. Therefore, when we encounter any mistake, we learn thoroughly and completely so that we understand where exactly the mistakes come from and solve the mistake.

There are several more aspects of professional and ethical responsibility that we did not talk about such as confidentiality, fairness, and professionalism. Throughout this project, we are able to work in an environment that we must behave and follow these guidelines. Having been immersed in this type of environment, we are now more able and suitable to enter the real world.

3.9 * Global, Economic, Environmental and Societal Impact

Our goal of this product is to improve network channel simulation. Current technology such as television cables, internet connections, to connections on PCB's may benefit from our product. With our product, engineers may more accurately simulate channel output while designing their products. Data loss is a common error in data transmitting information technology. With our product, it simulates a more accuracy summary which includes an eye diagram and bit error rate. Channel designers can then use these results to improve the design of network channels. Thereby, it can increase the performance of every new generation transmitting information technology. With the potential benefits that can improve on any information technology design, our product can have an impact on resource allocation.

- Resource allocation - In the case of our project, the resource allocation refers to digital/virtual resource. Because our simulation may improve the accuracy of the channel designs, we may predict that channel design engineers can build more efficient data transmitting devices. Therefore, data transferring speed may drastically increase. Even though the current information technology is already very advanced, our product can improve this aspect even more. We already have over 100mbps Internet connection for home internet. And with even more improved channel design, we maybe can download large files at an instant. With this advantage of the channel design, entertainment industry can also be improved.

One of many problems that entertainment users face is the limited broadband in internet connection. Video streaming, online interactions, and file down/uploads are all limited by the speed of the internet connection. With better-designed channels, our data can be transferred almost instantly with lower bit errors and better accuracy. Entertainment industry can create more advanced programs that could not even be thought of before due to lower data transferring speed. This aspect of internet connection is the most direct way our program can impact our lives. There are many more aspects that our program can improve but has a lesser impact such as more stable TV cable.

With fast information technology, the issue of copyright violation is definitely going to be more visible. Decades ago when computer started earning its importance in our lives, copy right issues were not much of a concern because there were limited ways to share our files. Moreover, it would take forever to share, for example, a video game file through Internet. However, now that with faster data transferring speed, copy right issues are much more noticeable. Many entertainment productions cannot make enough profit for what they produced because users illegally share and download their products off the internet. But just imagine if we can almost instantly download anything shared from the Internet, it would be extremely hard to reinforced the copyright law and solve this problem.

3.10 * Contemporary Engineering Issues

Nowadays, new or updated forms of technology become available every month. Most of these technology get replaced very fast because they do not solve a critical enough problem or they do not follow the current trend of technology. Our project design is a software based product for engineers to simulate their designed channel with. It does not have a direct relationship with technology trends. However, indirectly, our program may encounter several contemporary engineering issues.
Contemporary engineering issues that are related to our project are the following:

- Cloud computing
- Population becoming old and needing more care
- Internet security threats
- Reliability of channels

As discussed earlier in this report, our program can improve the information technology drastically by reducing errors and increasing accuracy. Therefore, internet speed may become much fast.

- Cloud computing is now a new trend that the current technology is currently building toward. With Cloud operations, a faster connection is more than desired to have. Our program can provide the tools for engineers to improve our internet connection speed and therefore suits the trend of cloud computing.

- Population becoming old and needing more care - average age of population has been increasing for some time. With more elderly people that require aid and care, technology forms a trend that goes toward assisting disabled or elderly people. For example, life alert system is a device that may detect any potential danger to its users such as trip and fall. The system can provide safety when emergency happens to disabled or elderly people.

With our program, we can improve information technology in terms of speed and accuracy. Therefore, a wider range of applications of technology can be made available to home and life security due to the faster data-transferring rate.

The above shows the benefits of our program that fits the trends in the present. However, our program is not only of benefits. Our program increases data transferring speed. Therefore, it is related to the current issue of internet security threats.
Nowadays, many individuals in the society practice to become hackers who steal information from people for personal gains. With faster data transferring, tracking and controlling for internet security would become more difficult as more data will be exchanged. In order to deal with this type of issues, internet security engineers must form a better and more efficient way to protect personal information.

The most important part of our program is increasing the reliability of newly designed channels. As discussed earlier in the report, although there are several products that can do simulation of the channel, we take in account of reference and terminal impedance of the channel. In this way, we can improve the accuracy of the simulation and let engineers design more reliable channels.

3.11 * Recognition of the need for and an ability to engage in lifelong learning

This design project helped us realize the need for lifelong learning process. Throughout this design project, we have learned to do many things that we have to been taught to. And these knowledge and skill that we have now on the project were all in the area of unknown a year ago when we first started doing the project. The project has taught me to efficiently deal with errors, testing data, and professional documentation skills. Moreover, every error we encountered in this project gave a different lesson that we have not learned from academic courses.

Moreover, the term “lifelong learning” now has a much more important meaning to us. In the academic world, students are expected to learn what the courses “toss” at them. Similar to a resistor in a circuit, learning in school is a passive mode of learning because students passively receive information. This project has taught us active mode of learning. Like a transistor, there is an external source required for it to work. It has taught us the process of self-motivate to learn. Thus, making it more difficult but interesting at the same time. Therefore, we may continue onto our career with a mind set for learning new things.

3.12 * Importance of Team Work

This project made me realize many things that I could not experience so deeply in the previous courses. First, about teammate, I cannot stress the importance of having a teammate.
A teammate provides:

- Potentially a second pair of eyes when debugging our codes
- Emotional support when project goes wrong
- Sharing work load
- Having a person on your side in presentation

There are times when my partner found a typo and fixes the whole program result. And there are times either of us understand a topic more and can explain to the other teammate. There are much more benefits on having a teammate.

We are both EE majors with similar knowledge. One of us knows more about Matlab coding. The other one knows more about GitHub structure. We were able to complete our software package and release it onto Prof. Tan's GitHub account despite of the difficulties we encountered.

Another thing that shows teamwork is important is that we can point the other member’s mistake in time when we are working together. This can really save a lot of time because we don’t need to spend useless time on one small and hard to find bug.

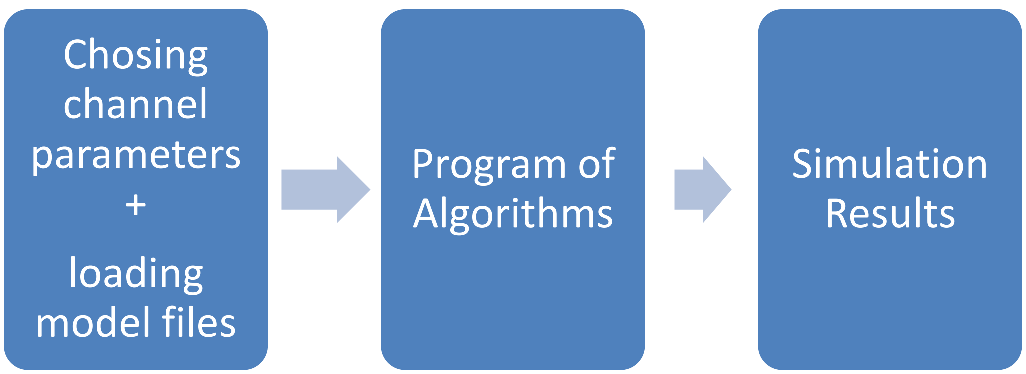
4 * High Level Design

4.1 Conceptual View

The system contains many steps including using the three major components discussed in previous sections. Conceptual view of the design includes: setting simulation and channel network parameters, selecting model file, Program algorithms, Result.

- Simulation and channel parameters include number of data bits to be simulated, channel frequency span, reference and terminal impedance settings.
- Selecting model file - Manual select AMI channel model and touchstone s parameter data for simulation
- Program algorithms - contains the three key features discussed in previous section 1.
- Results - result include the simulated output data and eye diagram

The diagram below overview on the conceptual view of our system

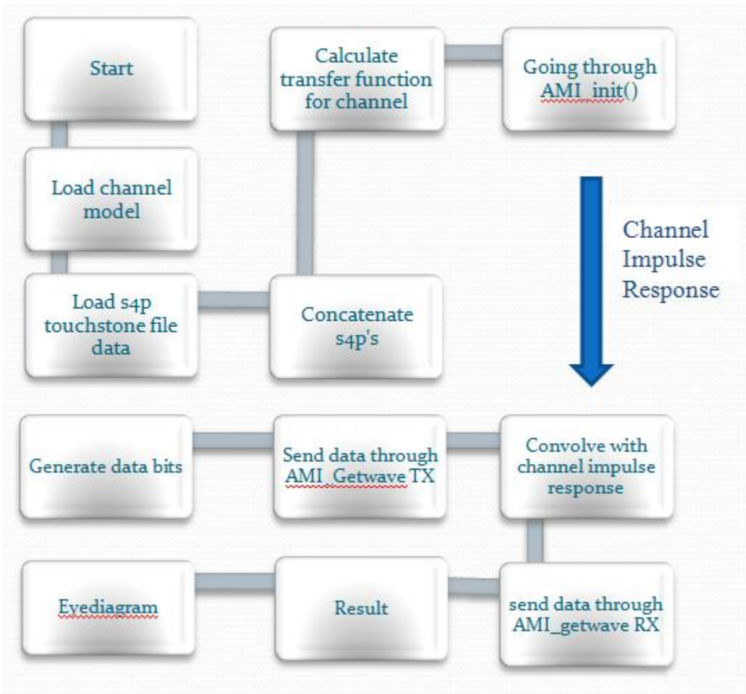


The conceptual high-level diagram of our project is shown on the diagram above.

The concept is to plug in designed, known parameters, and channel model files into our program of algorithms, then the output of the two is the result. Therefore, our program acts as a transfer function or the simulation.

4.2 Software

To understand our program, we have to use a more detailed software high-level design of our system because our program is entirely built on and for software uses.



The diagram above shows the software perspective of our entire project.

Each step presents the necessary steps for our simulation to run smoothly.

Step 1: Start - start of program

Step 2: Load channel model using tsnimporti function to load in channel .ami files

Step 3: Load channel S parameters in touchstone files

Step 4: Concatenate - Combine the loaded S parameters using concatenate function. *details in Section 5.

Step 5: Calculate transfer function for channel - using s2tf function to calculate the impulse response from the combined S parameters.

Step 6: AMI_init() - to initialize AMI functions for later use

Step 7: Generate data bits: Generate numbers of random bits, in our simulation, we generated 10000 bits of 1 and 0's. Each bit is to be sampled 64 times of magnitude of 0.5. The result is a sampled bit graph with NRZ convention.

Step 8: Using AMI_Getwave() function with TX channel model file and send in generated data. The output is a new simulated wave form from Transmitter Port.

Step 9: Convolve the new waveform in step 8 with our channel impulse response.

Step 10: Using AMI_Getwave() function with RX channel model file and send in the convolved waveform in step 9. The output is the output simulated wave form from Receiver Port.

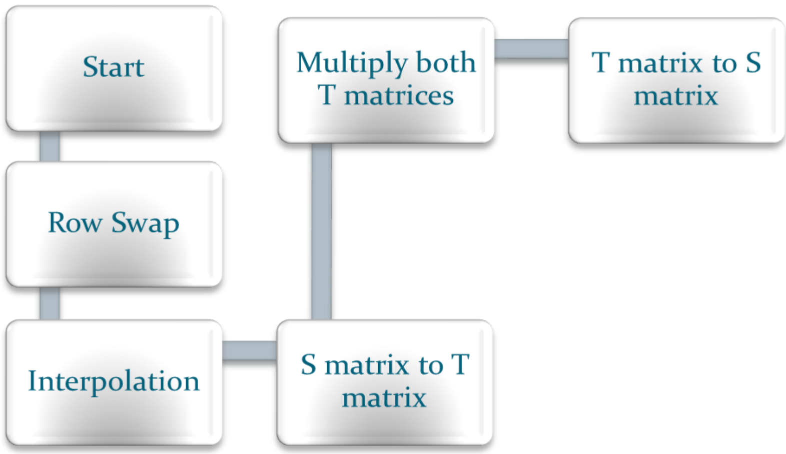
Step 11: The output from step 10 is our simulated result. We may now plot eye diagram.

5 * Low Level Design

5.1 Concatenate function module

The purpose is to combine channel s parameters into one in order to calculate channel impulse response. Note that a channel typically contains more than 1 component. Using the algorithms built into the concatenate function, we are able to combine these components' s parameters into one.

Functionality: Take in two touchstone type data structure in Matlab and concatenate them.

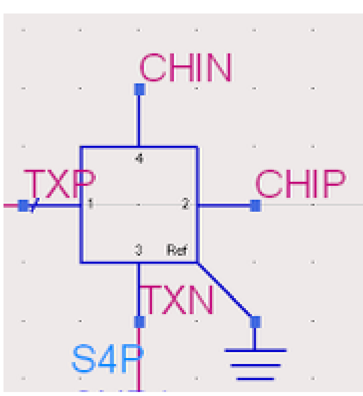


5.1.1 Processing narrative for the Concatenate function module

Concatenate function module is a function that has to be called on in the simulation program. The simulation program must contain:

- two touchstone structure files read into the program to be used with concatenate function
- two port indexes with respect to the two touchstone files

Port index is presented in the ADS schematics as shown below:



- This port index is needed in order to use the algorithm in jzhou_concatenate_s4p_derivation.pdf

5.1.2 Concatenate Module interface description.

A detailed description of the input and output interfaces of the module with other modules in the system, with other software or systems or module is presented.

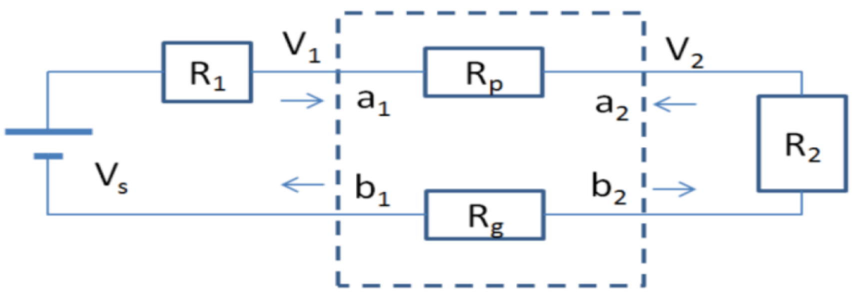
The input of concatenate function contains 4 components, SNP stands for S parameter structure type in Matlab:

1. SNP1 and SNP2
2. Pindex1 and Pindex 2
3. Desired frequency list fqi

The concatenate function outputs the concatenated SNPO, pindexo, error code, and error message.

5.1.3 Concatenate Module processing details

A detailed description for each module is presented, including hardware, algorithm, local data structures, design constraints, limitations, performance issues, etc.



Before concatenation, we need to insert DC points to S-parameter.

This is a two-port network. Rp and Rg are power and ground resistances at DC. We can obtain the values of Rp and Rg from DC analysis. Due to the lack of shunt resistances between power and ground, the Z-parameter of the network doesn't exist. For example, when port 1 is driven by a current source and when port 2 is open, the voltage at port 2 would be infinite. Due to this reason, we derive the S-parameter of this network directly from incident and reflected waves at DC. The reference impedances (resistances at DC) at port 1 and 2 are R1 and R2, respectively. The S-parameter at DC point can be derived through the following equations.

$$\sum R = R_1 + R_p + R_g + R_2$$

$$S_{11} = 1 - \frac{2R_1}{\sum R}$$

$$S_{21} = \frac{2\sqrt{R_1 R_2}}{\sum R}$$

$$S_{22} = 1 - \frac{2R_2}{\sum R}$$

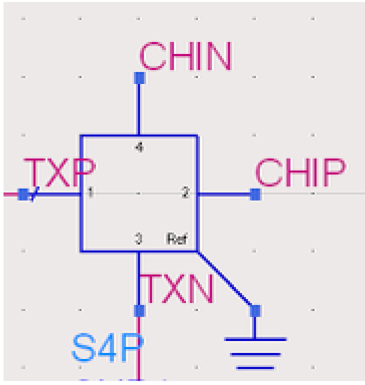
$$S_{12} = S_{21} = \frac{2\sqrt{R_1 R_2}}{\sum R}$$

After insertion of DC point, the concatenate function starts. There are couples steps in detail:

1. Interpolate the two S parameters into fqi. Note that this interpolation step does not extrapolate. Therefore, function user must choose this frequency list carefully.
2. Convert both S parameters to T parameters, this is done separately with function s2t_0415 written by Meng Lee, then do matrix multiplication
3. Then Convert the result T parameter back into S parameters, this is done separately with function t2s_0415 written by Meng Lee.

* Algorithms used in function s2t_0415 and t2s_0415 are inside jzhou_concatenate_s4p_derivation.pdf

* Reminder for using Port index



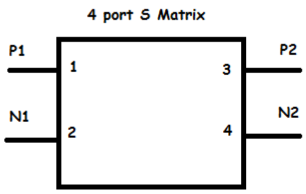
In order to using S-T and T-S conversion, we must define port index.

In this S4p (on the left), the port index is configured to be

$$\begin{bmatrix} TXP & CHIP \\ TXN & CHIN \end{bmatrix} -> \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

In order to use the derivation for conversion,

We must turn this into:



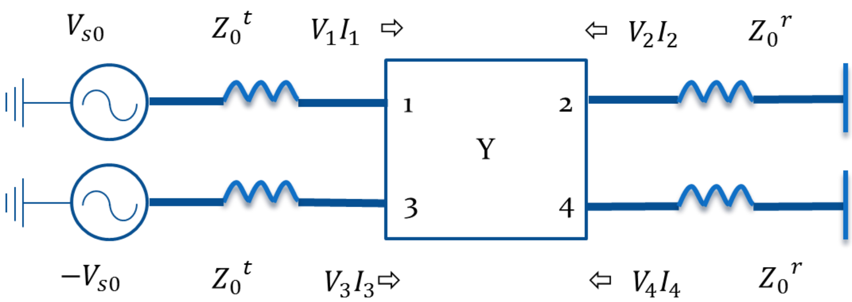
This is the reason for using swap function.

In this example case, we rowswap 2 and 3 and column swap 2 and 3 to obtain

$$\begin{bmatrix} TXP & CHIP \\ TXN & CHIN \end{bmatrix} -> \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$$

5.2 Channel impulse response module

5.2.1 Processing narrative for the channel impulse response module



This is the four-port network model we are using. R1 and R2 are $z_0{}^t$ and $z_0{}^r$. $z_0{}^t$ is TX reference impedance, $z_0{}^r$ is RX terminal impedance.

Because Z-parameter doesn't exist in DC point, we need to derive transfer function through Y-parameter after concatenate our touchstone files.

5.2.2 Channel impulse response module interface description

This module needs an s4p structure, $z_0{}^t$, $z_0{}^r$ and the case number as input and calculate the corresponding channel impulse response.

S4p structure is the concatenated touchstone files.

Case number is based on the channel's reference and terminal impedance. Below are 4 different scenarios that we have:

	$0< z_0{}^r <$inf	$z_0{}^r =$ inf
$z_0{}^t = 0$	Case 1	Case 3
$0 < z_0{}^t <$ inf	Case 2	Case 4

5.2.3 Derivation

Below is the derivation of case 2.

1. Convert S-parameter to Z-parameter using s2z function, and then convert Z-parameter to Y-parameter by inversing Z-parameter.

2. Then we have these, system equations: $\mathrm{I} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix}$ $\quad \mathrm{V} = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix}$ $\quad \mathrm{I} = \mathrm{YV}$

Terminal equations: V_2 = -${Z_0}^r I_2$ V_4 = -${Z_0}^r I_4$

$V_{s0} - V_1 = {Z_0}^t I_1$ -$V_{s0} - V_3 = {Z_0}^t I_3$

Mixed mode voltages and currents: $V_{d1} = V_1 - V_3$ $V_{c1} = \frac{V_1+V_3}{2}$

$V_{d2} = V_2 - V_4$ $V_{c2} = \frac{V_2+V_4}{2}$

$I_{d1} = \frac{I_1-I_3}{2}$ $I_{c1} = I_1 + I_3$

$I_{d2} = \frac{I_2-I_4}{2}$ $I_{c2} = I_2 + I_4$

3. Then derive single-end to mixed mode mapping:

$$V_m = \begin{bmatrix} V_{d1} \\ V_{d2} \\ V_{c1} \\ V_{c2} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0.5 & 0 & 0.5 & 0 \\ 0 & 0.5 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = M_v V$$

$$I_m = \begin{bmatrix} I_{d1} \\ I_{d2} \\ I_{c1} \\ I_{c2} \end{bmatrix} = \begin{bmatrix} 0.5 & 0 & -0.5 & 0 \\ 0 & 0.5 & 0 & -0.5 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} = M_I I$$

4. Then we have: $I_m = M_I Y M_v^{-1} V_m$

$$Y_m = M_I Y {M_v}^{-1} = \begin{bmatrix} Y_{dd} & Y_{dc} \\ Y_{cd} & Y_{cc} \end{bmatrix}$$

5. From previous terminal equations, we have:

V_{d1} = $2V_{s0}$ - $2{Z_0}^t I_{d1}$

$V_{c1} = -\frac{{Z_0}^t}{2} I_{c1}$

V_{d2} = -$2{Z_0}^r I_{d2}$

$V_{c2} = -\frac{{Z_0}^r}{2} I_{c2}$

$$\begin{bmatrix} V_{d1} \\ V_{d2} \\ V_{c1} \\ V_{c2} \end{bmatrix} = \begin{bmatrix} -2{Z_0}^t & 0 & 0 & 0 \\ 0 & -2{Z_0}^r & 0 & 0 \\ 0 & 0 & -\frac{{Z_0}^t}{2} & 0 \\ 0 & 0 & 0 & -\frac{{Z_0}^r}{2} \end{bmatrix} \begin{bmatrix} I_{d1} \\ I_{d2} \\ I_{c1} \\ I_{c2} \end{bmatrix} + \begin{bmatrix} 2V_{s0} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$Z^{tm} = \begin{bmatrix} -2{Z_0}^t & 0 & 0 & 0 \\ 0 & -2{Z_0}^r & 0 & 0 \\ 0 & 0 & -\frac{{Z_0}^t}{2} & 0 \\ 0 & 0 & 0 & -\frac{{Z_0}^r}{2} \end{bmatrix} \quad V_s = \begin{bmatrix} 2V_{s0} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

6. Then, we can derive transfer function through:

$V_m = Z^{tm} I_m + V_s$

$V_m = Z^{tm} M_I Y M_v^{-1} V_m + V_s$

$V_m = \left(u - Z^{tm} M_I Y M_v^{-1}\right)^{-1} V_s$

7. Get transfer function:

   According to the derivation, $(u - Z^{tm} M_I Y M_v^{-1})^{-1}$ is the transfer function we get.

   Because V_{d2} and V_{c2} are the output voltages and $V_s = \begin{bmatrix} 2V_{s0} \\ 0 \\ 0 \\ 0 \end{bmatrix}$, we only need to extract the (2, 1) and (4, 1) elements in the $(u - Z^{tm} M_I Y M_v^{-1})^{-1}$ 4×4 matrix. We add them together and that would be the final transfer function.

8. After we got the transfer function, we do inverse Fourier transform to it and get its impulse response by using MATLAB's ifft function.

   We need to make our transfer function a periodic function by mirror its whole part to the right so that after ifft our channel impulse response will not have any complex part.

9. To find the corresponding time list to frequency list of the transfer function, we derive a series of time and frequency domain relations.

   fstop = fq(nfp);   % in Hz

   fstep = fstop / (nfp - 1); % in Hz

   fspan = nfp * fstep;

   tstep = 1/fspan;   % in seconds

   tspan = 1/fstep;   % in seconds

   tlist = 0:tstep:(2*nfp-2)*tstep;

10. At last, we need to interpolate impulse response to have the same step with the input signal which should be the sample step. After that, we can get the final channel impulse response.

The other cases are with pretty same derivations, so I will put the result of each case in the following.

Case 1 (${z_0}^t$ = 0 and 0< ${z_0}^r$ <inf): $V_m = \left(u - Z^{tm} Y_m\right)^{-1} V_s$

Case 3 (${z_0}^t$ = 0 and ${z_0}^r$ = inf): $\begin{bmatrix} V_{d2} \\ V_{c2} \end{bmatrix} = -(Y_{22}^m)^{-1} Y_{21}^m V_s$

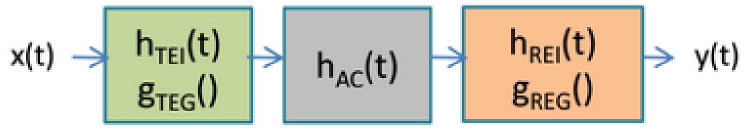
Case 4 (0 < ${z_0}^t$ < inf and ${z_0}^r$ = inf): $\begin{bmatrix} V_{d2} \\ V_{c2} \end{bmatrix} = -(Y_{22}^m)^{-1} Y_{21}^m \left(u - \begin{bmatrix} -2{Z_0}^t & 0 \\ 0 & -\frac{{Z_0}^t}{2} \end{bmatrix} Y_{11}^{m\prime}\right)^{-1} V_s$

$$Y_{11}^{m\prime} = Y_{11}^m - Y_{12}^m (Y_{22}^m)^{-1} Y_{21}^m$$

The derivations of channel impulse response from the transfer function are all the same.

5.3 AMI functions module

5.3.1 Processing narrative for the AMI functions module



We use the algorithm of IBIS 5.1 to recover the input signal after it passing through the channel.

The green block is the TX and the orange block is the RX. The hac(t) block is the channel impulse response.

We can get the channel model from a form of .ami and .dll files. The executable model file of a Serializer-Deserializer (SERDES) transmitter or receiver contains up to three functions:

- AMI_Init
- AMI_GetWave
- AMI_Close

The interfaces to these functions are designed to support three different phases of the simulation process:

- Initialization
- Simulation of a segment of time
- Termination of the simulation.

5.3.2 AMI functions module interface description

AMI_Init:

This function requires inputs of the ami file name, dll file name, channel impulse response, sample step and bit time.

The outputs of this function are optimized impulse response, a memory handle that is needed by AMI_GetWave and the contents of the ami file.

AMI_GetWave:

This function requires inputs of the dll file name, memory handle from AMI_Init, input signal and clock times.

The outputs are the restored signal, clock time after restored and a memory handle that needed by another AMI_GetWave.

The AMI_GetWave is a black box using to transmit and recover our data bits.

AMI_Close:

This function is automatically called after the RX AMI_GetWave to terminate our simulation.

5.3.3 Process of calling AMI functions

After get the channel impulse response, we need to pass it through TX and RX AMI_Init to initialize the ami functions. Actually, AMI_Init takes in our channel impulse response and convolve with some algorithms it has within its model file to give back an optimized impulse response.

When input signal x(t) comes in, it first pass through TX AMI_GetWave. In our case, x(t) is a sequence of 10k random bits for which we take 64 samples each bit so that we have 640k data points using NRZ conversion. Then the data is ready to convolve with the channel impulse response hac_t to simulate the result of the signal x(t) that is passed through the channel. That result is as well the input of RX AMI_GetWave. At last, we send out data into RX AMI_GetWave in a similar fashion as TX to restore x(t) and plotted into eye diagram.

6 * EXPERIMENT DESIGN AND TEST PLAN

6.1 * Design of Tests

Design tests to verify whether your design meets the design objectives. Design experiments to verify the functions of your modules and the prototype you built. The set of tests, which constitute your test plan, are to be specified, including as much detail as is possible at this stage.

Tests and expected responses

Test Case 1:

1. The objective of test case one is to verify the result calculated with concatenation feature. The technical design objective is for our programs result to be at least 16 digits accurate compared to the Matlab RF tool. Touchstone file contains thousands of 4-by-4 S parameters for corresponding frequency points. To test this function, we compare to Matlab RF tool. Test result will be the max difference between any number(2000*4x4 = 32000 numbers) between RF tool and our program
2. Test setup

   Step 1: Inside test and verification folder in the software package, concatenate_test_0501.m is used to compare result between program result and Matlab RF Tool Concatenation results. Locate the file and open.

   Step 2: Run the concatenation_test .m. The result 'max_error' is the max value of difference between program result and Matlab RF tool result.

   Record the value
3. Expected results - Max error should be less than 10^(-15)

   Reason to have this error standard - In computer architecture when performing calculation, results are not entirely accurate. 16-digit accuracy is sufficient to state that the difference is due to computer calculation error.

Team member Meng Lee is responsible for Test Case 1.

Test Case 2:

1. The objective of test case two is to verify the result calculated with channel impulse response function: s2tf_0506.

   The technical design object is to check and verify two perspectives:

   a. Frequency list to Time list conversion

   b. Compare our program to Matlab RF tool.
2. Test setup:

   Step 1: Inside test and verification folder in the software package, s2tf_0506_test .m is used to compare result between program result and Matlab RF Tool s2tf function results. Locate the file and open.

   Step 2: Run the s2tf_0506_test.m.

   'Max_TF_error' is the max value of difference between program result and Matlab RF tool result.

'Time_conversion_error' is the max value of difference between original time list and time list after converting.

Record the values

3. Expected results - Max errors should be less than 10^(-15)

   Reason to have this error standard - In computer architecture when performing calculation, results are not entirely accurate. 16-digit accuracy is sufficient to state that the difference is due to computer calculation error.

Test Case 3:

1. The objective of test case three is to test the eye pattern result.

   The technical design object is to record and measure the eye pattern of our result

2. Test setup:

   Step 1: Run main.m program

   Step 2: Visually assess eye diagram first.

   Measure and record the values of the following measurements:=

   a. Eye width

   b. Eye Amplitude

   c. Zero crossing time

   d. Best sampling time

6.2 * Bug Tracking

A database will be used to track defects found while performing the test cases. All defects will be logged as they are discovered. Defects will be assigned to Person A to fix, or to Person B to investigate.

Test 1 Bug Tracking:

Detailed information on how to perform the concatenate_test_0501.m is commented inside the code.

Defects will be assigned to Meng Lee to investigate and fix.

Test 2 Bug Tracking:

Detailed information on how to perform s2tf_0506_test.m is commented inside the code.

Defects will be assigned to Meng Lee to investigate and fix

6.3 * Quality Control

The completed test cases will be reviewed to ensure that all cases were run; that all were completed successfully; and that any deviations from the test cases were noted accordingly. Each step should be marked as Passed or Failed. Failed cases should be marked with the date and time of the failure, and the associated test track number. When a failed case is fixed, the date and time of the retest should be noted

6.4 * Identification of critical components

The results from test case 1 and 2 are equally important to be tested. If one of the two fails to stay in the range of error, the end result will be completely off and not worth using.

The result from test case 3 is only a measurement. There is no comparison/expected result to compare to.

6.5 * Items Not Tested

We have not tested the AMI model functions. The reason is that we do not have to test these function because the software package that IBIS AMI provides is in a black box. Therefore, we assume that this part works correctly.

7 * Test Report

Carry out the tests designed in the section above to test your modules and prototype and present the results. Present the results of the tests and provide an analysis of the test data.

State clearly who is responsible for which test case

7.1 * Test Iteration 1

Each experiment may need to be run multiple times. When you find deviation from the expected results, you must take action to debug your design, then test it again.

7.1.1 Test 1-001

1. Meng Lee performing Test# 1-001:
   Track #: Conc-Test-T1-001
   Date : 5_2_2015
   Procedure:
   - Run Test program concatenate_test_0501.m in test and verification folder
   - Screenshot Result in Matlab Command Window

2. Comparison with expected results:

```
Max_error =

      0.5570
```

3. Analysis of test results
   The result presented a critical error in the max error between expected result and real result.
   A 0.5570 Magnitude of difference definitely a misunderstanding in algorithm of some reason.
   Failed

4. Corrective actions taken:
   - Reason for errors: at the time of this function's starting development, Meng Lee used element multiplication rather than matrix multiplication.
   - Corrective actions: Changed all .* into * in order to have matrix multiplication

7.1.2 Test 1-002

1. Meng Lee performing Test# 1-002
   Track #: Conc-Test-T1-002
   Date : 5_06_2015
   Procedure:
   - Run Test program concatenate_test_0501.m in test and verification folder
   - Screenshot Result in Matlab Command Window

2. Comparison with expected results:

```
Max_error =

    0.5570
```

3. Analysis of test results
   As shown in the Error Matrix in the test program below, The upper right hand 2-by-2 matrix generated the most errors.

```
Error <4x4x2001 complex double>

  -0.0002 - 0.0002i  -0.0008 - 0.0003i  -0.0231 - 0.0561i  -0.9045 - 0.4073i
   0.0006 + 0.0006i   0.0001 + 0.0002i   0.6385 + 0.7120i   0.0140 + 0.0306i
        0 + 0.0000i        0             0.0000 - 0.0000i        0 + 0.0000i
   0.0000 - 0.0000i  -0.0000 + 0.0000i  -0.0000 - 0.0000i  -0.0000 + 0.0000i


val(:,:,4) =

   1.0e-06 *

  -0.0003 - 0.0001i  -0.0010 + 0.0001i  -0.0458 + 0.0655i  -0.9240 - 0.1223i
   0.0007 + 0.0005i   0.0001 + 0.0002i   0.5655 + 0.5720i   0.0439 - 0.0[illegible]0i
        0 + 0.0000i  -0.0000 - 0.0000i  [illegible] + 0.0000i   0.0[illegible]
   0.0000 - 0.0000i   0.0000 + 0.0000i  -0.0000 + 0.0000i   0.0000 + 0.0000i


val(:,:,5) =

   1.0e-05 *

  -0.0000 - 0.0000i  -0.0001 + 0.0001i  -0.0011 + 0.0062i  -0.1006 + 0.0288i
   0.0001 + 0.0000i   0.0000 + 0.0000i   0.0780 + 0.0289i   0.0004 - 0.0061i
```

   The error is most likely from t2s_0415 because of the following reasoning:
   - If algorithms in s2t_0415 caused this error, then most likely all four matrices would have a similar magnitude of error.
   - Algorithms in t2s_0415 are more likely to result in this localized error.
   Failed

4. Corrective actions taken
   - Action: Investigate the ending step of t2s_0415 that generated our top right 2-by-2 matrix result.
   - Errors found: the arrangement of 3 matrix multiplication is wrong resulting in this error.

7.1.3 Test 1-003

1. Meng Lee performing Test# 1-003
   Track #: Conc-Test-T1-003
   Date : 5_07_2015
   Procedure:
   - Run Test program concatenate_test_0501.m in test and verification folder
   - Screenshot Result in Matlab Command Window

2. Comparison with expected results:

```
Max_error =

   7.8554e-16
```

3. Analysis of test results
   This result shows a less than 10e-14 error. It means that our result is at least 15 digits accurate comparing to the Matlab RF tool result. Therefore, the concatenate function is correct. No need for corrective action.
   Passed

7.2 * Test Iteration 2

7.2.1 Test 2-001

1. Meng Lee performing Test# 2-001
   Track #: S2TF-Test-T2-001
   Date : 5_09_2015
   Procedure:
   - Run Test program s2tf_test_0506.m in test and verification folder
   - Screenshot Result in Matlab Command Window
2. Comparison with expected results

```
Max_TF_error =

    0.0090


Transfer_function_test_pass =

     0


Time_conversion_error =

     0


Time_conversion_test_pass =

     1
```

3. Analysis of test results
   The result shows that compared result between RF tool and our result has a 10e-4 magnitude of error. This is a distance from our technical goal which has to be smaller than 10e-14. Possible error may come from the algorithms of obtaining transfer function.
   Data results Failed
   The time conversion result shows an error of 0. This is within the range of 0 to10e-14.
   Time conversion results Passed

4. Corrective actions taken
   - Investigate the algorithm used for transfer function.
   - Found cause: in mix mode, transmitter generally ignore common mode result, therefore, by having only differential mode, the result should be correct.

7.2.2 Test 2-002

1. Meng Lee performing Test# 2-002
   Track #: S2TF-Test-T2-002
   Date : 5_17_2015
   Procedure:
   - Run Test program concatenate_test_0506.m in test and verification folder
   - Screenshot Result in Matlab Command Window
2. Comparison with expected results

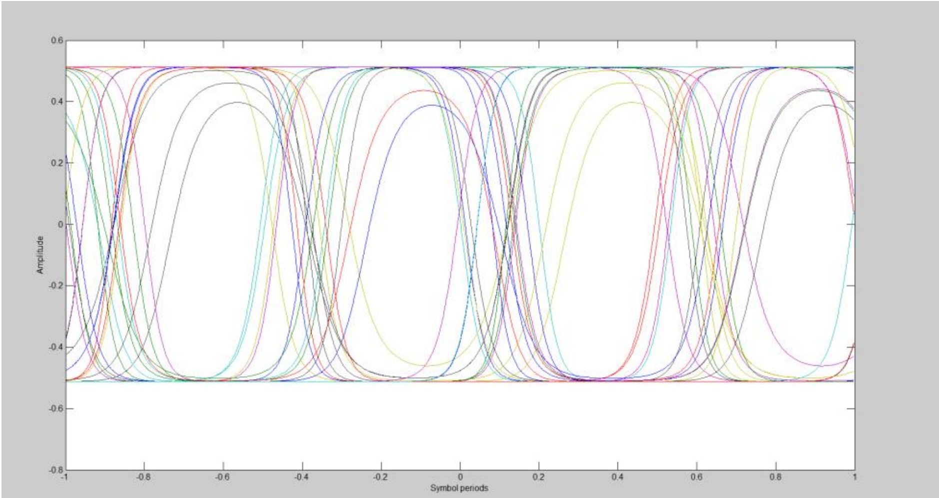
```
Max_TF_error =

   1.6665e-15


Transfer_function_test_pass =

     1


Time_conversion_error =

     0


Time_conversion_test_pass =

     1
```

3. Analysis of test results
   The result shows that compared result between RF tool and our result has a 10e-4 magnitude of error. This is a distance from our technical goal which has to be smaller than 10e-14. Possible error may come from the algorithms of obtaining transfer function.
   The result shows that expected result and our calculation is only 10e-15 off. This is in the range of 0 to 10e-14 of our technical objective.
   Data conversion results Passed
   No need for further testing

7.3 * Test Iteration 3

7.3.1 Test 3-001

1. Meng Lee performing Test# 3-001
   Track #: Eye-Measure-T3-001
   Date : 5_24_2015
2. Results



3. Analysis of test results
   Measure and record the values of the following measurements:=

   a. Eye width - 0.21

   b. Eye Amplitude - 1

   c. Zero crossing time - 0 - 0.22

   d. Best sampling time - 0.06

   After taking these measurements, it shows that our timing is a little bit off. We can see from the eye pattern that the zero crossing time is too wide. Even though there are four eyes showing in our result, we cannot be sure that the result is 100% correct.

4. Corrective actions taken
   - Investigate the usage for AMI functions.

8 * Conclusion and Future Work

8.1 * Conclusion

Present the conclusion of your project, state clearly whether the finished work meet the overall project goals and the quantitative technical design objectives. If not, just state what and how it failed and the TECHNICAL reasons why it failed and what you have learned.
State clearly who is responsible for achieving or missing which goals/objectives.

Meng Lee:

After tremendous time and effort for understanding the concept, implementing the concept, and debugging our program, our finished program meets the overall project goal. The program meets the quantitative technical design objective. However, there is still work to be done on different cases of simulation.

There are countless aspects I learned from this project.
These following aspects are some of the most important learning experience

1. The project is very different from normal academic curriculum. I have to self-motivate to complete parts of the project so that we can meet the deadline of the project. This is very similar to what a work field should be. Before everything step of software development, we did a clear overview on topics to be covered and the date to complete it by. There are many times when I simply cannot meet the deadline because unexpected results came up due to wrong approaches or coding errors.
2. People are not perfect. We are all prone to make mistakes no what how well we know any subjects. During the software development, I have encountered at least hundreds of errors. Most of them are fixed quickly, but some of them are hidden crucial errors that take days to investigate. Therefore, I learned very effective ways to approach and detect errors:
   - Understand accepted range of error
   - Form hypotheses and speculations
   - Form test plans
     - For example, S matrix to T matrix conversion is irreversible such that changing back from T matrix to S matrix should result in not more than computing error.
   - Pin point errors and possible affecting range

Yuan Yao:

After months of work, our efforts finally paid off. We finished the project in time and achieved the expected goal. However, the whole project still needs to de more perfect.

I have learned much from this project.
We have been doing this project for a very long time and I find it is really difficult to totally understand every part of the project. I knew little about channel simulation when I first met the project and it took me three to four weeks to read about its corresponding documents. After understood what this project actually doing, I spent weeks of time learning how to use MATLAB and ADS that are needed in the project.
When I started writing my own MATLAB program, there appeared numbers of difficulties related to how to use the algorithm of transfer function. Fortunately, our advisor James and Sheldon are very nice people and helped us solve problems every week. Thanks to their patient explanation, I can figure out most of the difficulties I met.

The most difficult part in the project was debugging. There were large numbers of bugs after we wrote our program. Most of the bugs were easily to find while some of them were hidden in the depths of our program. For those hidden bugs, we didn't realize those bugs even if they caused our result incorrect and would think the algorithms we use were incorrect instead. We spent so much unnecessary time on debugging our algorithms and turned out nothing. When we finally found the bug, it was just that we mistook a small detail hidden in the very depth of one of our functions. It turned out that we had already finished our project a few weeks ago. Unfortunately, because of that small bug, we had to extend our project for weeks.
The most important thing I learned from the project is that try to first best understand one thing before you are going to do it. In this way, we can save a lot of time spending on unnecessary debugging.

8.2 Future Work

Expansion and Improvement; discuss the impact of this work and its possible expansion into perhaps a more promising design than what you had started. This is particularly important in order to address the marketability of your design. Or why this project merits another look by perhaps next year's students.

8.3 * Acknowledgement

Professor Shelton Tan, for giving us the opportunity to be part of the team to develop this new technology

James Zhou, Sr. Principal Engineer, Qlogic Corporation, for walking us through the entire project and teach us the professional aspect of Electrical Engineering

Professor Ping Liang, for helping us with issues and question regarding this project and school requirements.

9 * References

List the references used in the design, including books, data sheets, technical documents, industry standard documents. References can be printed documents or online.

Channel_analysis_derivation_2015_02_24_0005	James Zhou
Channel_analysis_incident_voltage_method_2015_02_25	James Zhou
IBIS (I/O Buffer Information Specification) Version 5.1	EDA Industry Working Groups
Jzhou_concatenate_s4p_derivation	James Zhou
Jzhou_four_port_S_parameter_at_DC_2015-02-16	James Zhou
Network-analysis-05	James Zhou, Sheldon X.D. Tan
Sedres_new	James Zhou
Serdes_1211	James Zhou
Serdes Channel Simulation Algorithms and Workflows using IBIS AMI Models - IBIS SUMMIT at DesignCon JZHOU 2013-01-28	James Zhou
Time-frequency transform of band-limited signals 02 - jzhou_20150303	James Zhou

10 Appendices

All source codes and references can be found in the software package:

https://drive.google.com/drive/folders/0B15mlkxHMyqLflA5YTJLckdHRnh0LTlDaFZHTjE3T1dWXzdLNy1ZaGpuRG5DcWxLZHRtTTQ