Evaluation of Space-Time Block Codes Under Controlled Fading Conditions Using Hardware Simulation

A Dissertation Proposal Submitted to the Temple University Graduate Board

In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Engineering

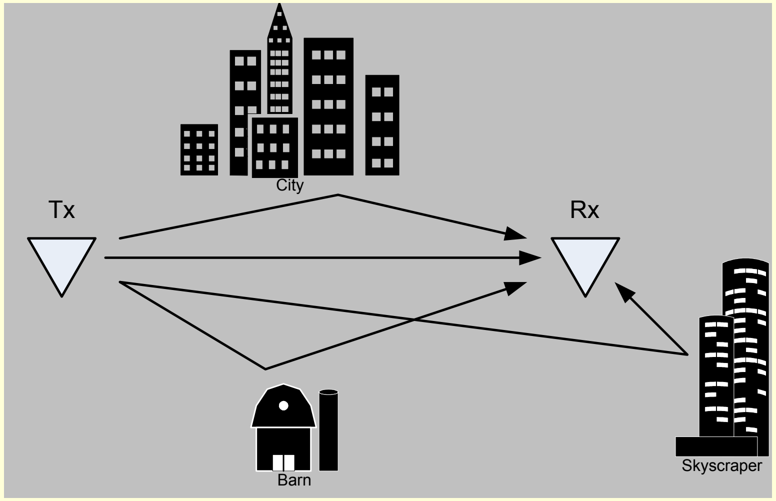
By Leonard Colavito

College of Engineering
TEMPLE UNIVERSITY

Introduction

Multiple-Input Multiple-Output (MIMO)

- MIMO digital wireless communication systems achieve significant BER performance increase over other systems by utilizing multipath as an advantage.
- Multipath results when signals take two or more paths from a transmitter to a receiver introducing signal fading
- MIMO systems take advantage of multipath by using two or more antennas at the transmitter or receiver or both



Space-Time Block Codes (STBC)

- Class of MIMO codes that encode data symbols in blocks (Calderbank, et. al., 1997)
- Code block defines a sequence of symbol sets
- Symbol set is transmitted simultaneously during one symbol time
- STBC allows for linear decoding (Alamouti, 1998)

Modeling MIMO Systems

- MIMO systems modeled to test performance
- Models used
  - Software simulations (Alamouti, 1998, Tarokh, et al 1999)
    use random data and channel characteristics to compute BER performance
  - Real-world experiments (Goud, et al, 2003)
    measurements made in office environment, data post-processed
  - Hardware simulations (Murphy, et al, 2007)
    use DSPs and PGAs to support rapid-prototyping of MIMO system designs
- Models evaluate bit error rate (BER) verses signal to noise (SNR) performance

Controlled Channel Conditions

- Extend the definition of channel path characteristic to include a path specific fixed attenuation A_{ik} and additional phase delay Φ_{ik}.
- Allows for configuration of specific conditions
  - Extra loss in one or more paths
  - Additional delay of a path
- Progressive variation allows for evaluation of performance over changing channel conditions

$$h_{ik} = \mathrm{A}_{ik}\alpha_{ik}\exp(j[\theta_{ik} + \Phi_{ik}])$$

Hardware Acceleration

- Reduces time required to obtain MIMO system performance results.
- Allows for greater BER precision through processing of more bits.
- Allows for greater BER versus SNR curve resolution through processing of more SNR points.
- Allows evaluation of BER versus SNR curves under more conditions.

Research Objectives

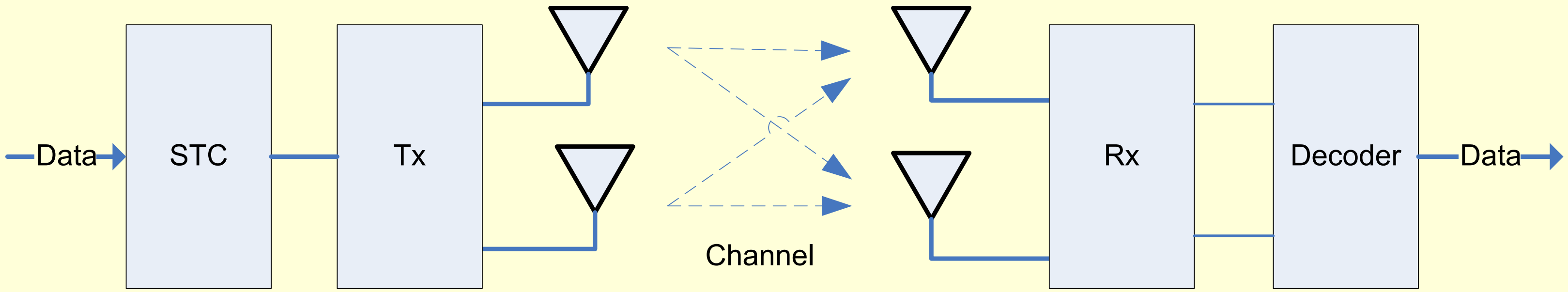
- Create a software STBC base MIMO system model employing controlled channel conditions
- Create a hardware accelerated version of the model.
- Demonstrate the use of the models in evaluation of a STBC under degrading signal path conditions.
- Compare the performance of the software model to the hardware accelerated model.

Model Design

College of Engineering
TEMPLE UNIVERSITY

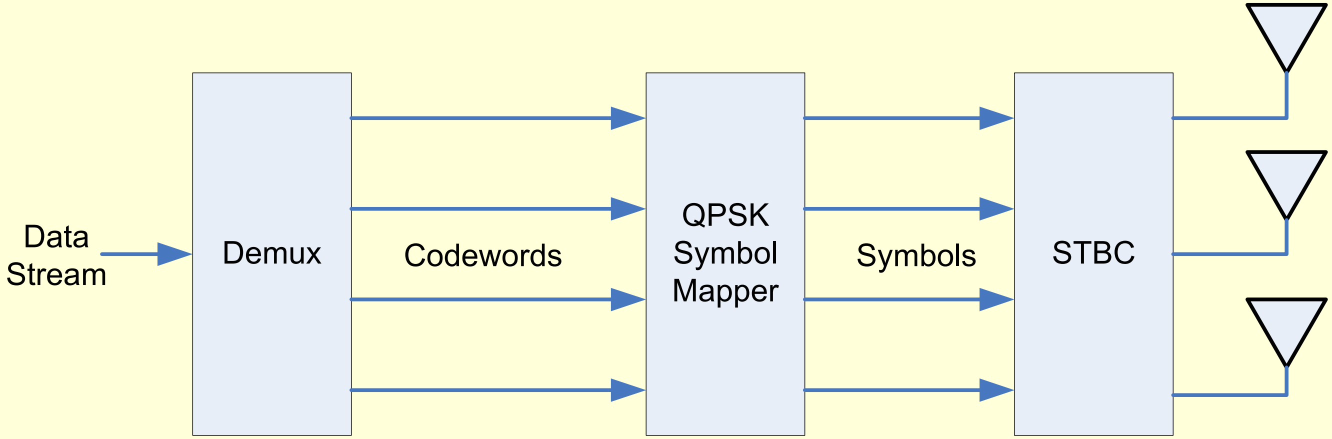
MIMO System

- Space-Time Code
- Transmitter and Receiver
- Multipath Channel
- MIMO Decoder



Transmitter

- Data stream divided into codewords
- Codewords are mapped to base modulation (QPSK) symbols
- Symbols encoded and transmitted according to the STBC



College of Engineering
TEMPLE UNIVERSITY

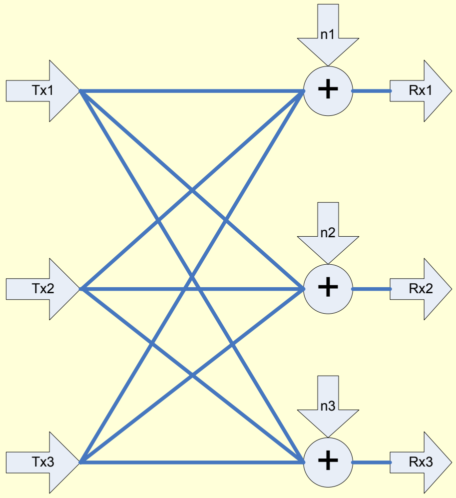
Space-Time Block Code

- 3-Antenna Rate ½
  (Tarokh et. al., 1999)

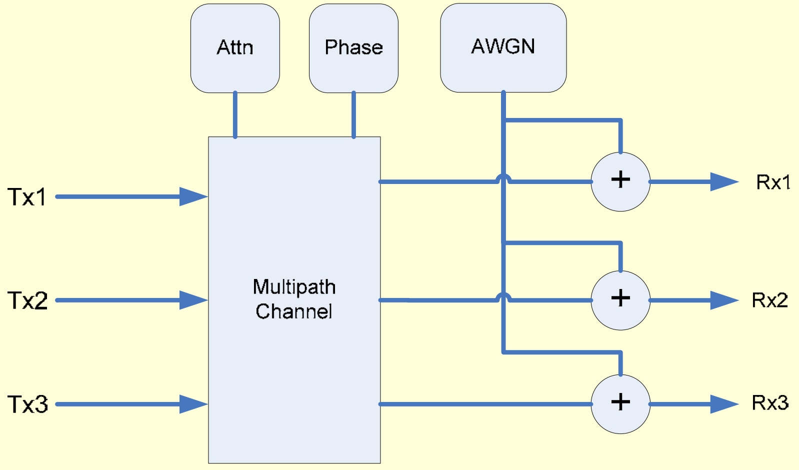
$$\mathbf{G} = \begin{pmatrix} x_0 & x_1 & x_2 \\ -x_1 & x_0 & -x_3 \\ -x_2 & x_3 & x_0 \\ -x_3 & -x_2 & x_1 \\ x_0^* & x_1^* & x_2^* \\ -x_1^* & x_0^* & -x_3^* \\ -x_2^* & x_3^* & x_0^* \\ -x_3^* & -x_2^* & x_1^* \end{pmatrix}$$

Channel

- Fading path between each transmit and receive antenna
- Additive white Gaussian noise (AWGN)
- Assumed quasi-static, channel characteristics do not change over block time



Path View



Process View

College of Engineering
TEMPLE UNIVERSITY

Channel Characteristic

- Channel matrix H describes the fading characteristics of the signal paths

$$\mathbf{H} = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{bmatrix}$$

- Path characteristic
  - Rayleigh distributed random attenuation α_{ik}
  - Linearly distributed random phase θ_{ik}
  - Constant attenuation A_{ik}
  - Constant additional phase shift Φ_{ik}

$$h_{ik} = A_{ik}\alpha_{ik}\exp(j[\theta_{ik} + \Phi_{ik}])$$

College of Engineering
TEMPLE UNIVERSITY

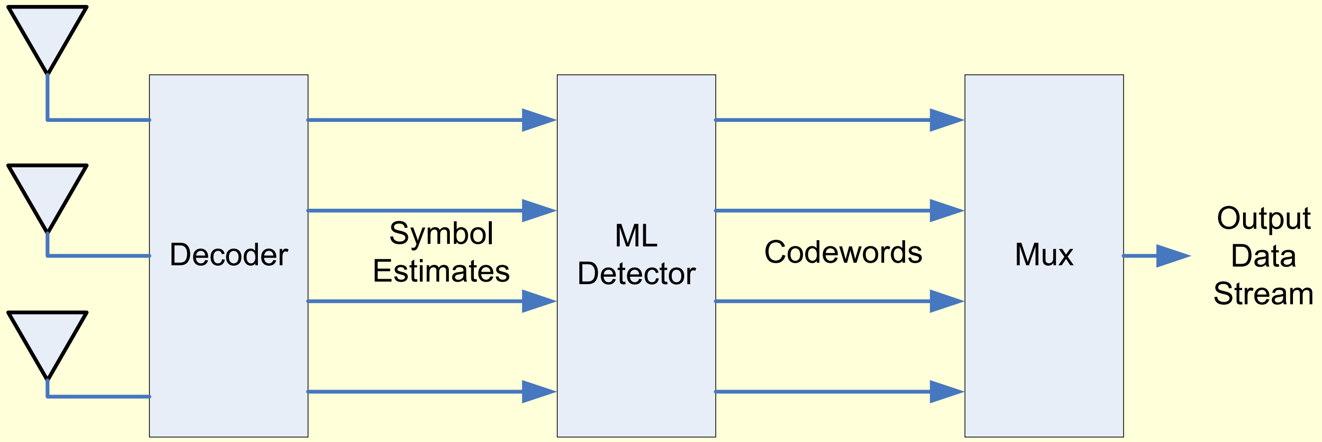
Received Signals

- Receive signal r_k is the sum of signals and noise
  - s_i : Signal from transmit antenna
  - h_{ik} : Path characteristic
  - n_k : AWGN at receive antenna

$$r_k = \sum_{i=1}^{m} h_{ik} s_i + n_k$$

Receiver

- Decoder estimates the transmitted symbols by linear computation
- Detector maps symbols to codewords by maximum likelihood
- Codewords multiplexed into output data stream



Decoder

- Assumes complete channel state information (CSI)
- Computes estimates of transmitted symbols by linear combination of received signals

$$\tilde{x}_0 = \sum_{k=0}^{2}\left[r_k^{(0)}h_{0k}^* + r_k^{(1)}h_{1k}^* + r_k^{(2)}h_{2k}^* + \left(r_k^{(4)}\right)^* h_{0k} + \left(r_k^{(5)}\right)^* h_{1k} + \left(r_k^{(6)}\right)^* h_{2k}\right]$$

$$\tilde{x}_1 = \sum_{k=0}^{2}\left[r_k^{(0)}h_{1k}^* - r_k^{(1)}h_{0k}^* + r_k^{(3)}h_{2k}^* + \left(r_k^{(4)}\right)^* h_{1k} - \left(r_k^{(5)}\right)^* h_{0k} + \left(r_k^{(7)}\right)^* h_{2k}\right]$$

$$\tilde{x}_2 = \sum_{k=0}^{2}\left[r_k^{(0)}h_{2k}^* - r_k^{(2)}h_{0k}^* - r_k^{(3)}h_{1k}^* + \left(r_k^{(4)}\right)^* h_{2k} - \left(r_k^{(6)}\right)^* h_{0k} - \left(r_k^{(7)}\right)^* h_{1k}\right]$$

$$\tilde{x}_4 = \sum_{k=0}^{2}\left[- r_k^{(1)}h_{2k}^* + r_k^{(2)}h_{1k}^* - r_k^{(3)}h_{0k}^* - \left(r_k^{(5)}\right)^* h_{2k} + \left(r_k^{(6)}\right)^* h_{1k} - \left(r_k^{(7)}\right)^* h_{0k}\right]$$

Detector

- Uses maximum likelihood criteria to select transmitted symbol based on decoder estimate
- *Minimum distance* selection criteria

$$d(\tilde{x}_n) = \min_{i=0,1,2,3}\left(\left|\tilde{x}_n - s_i\right|^2 \right)$$

BER Evaluation

- Bit-error-rate used to quantify system performance
- Bit errors detected by comparison of output with input
- Counters for total bit errors and total bits transferred

$$BER = \frac{(Total\ Bit\ Errors)}{(Total\ Bits\ Transfered)}$$

Software Model

Purpose of Software Model

- Proof of design
  - Easier to debug, fast design changes
  - Uses standard libraries
  - Uses floating point numbers
- Confirm hardware model results
- Performance reference for hardware model

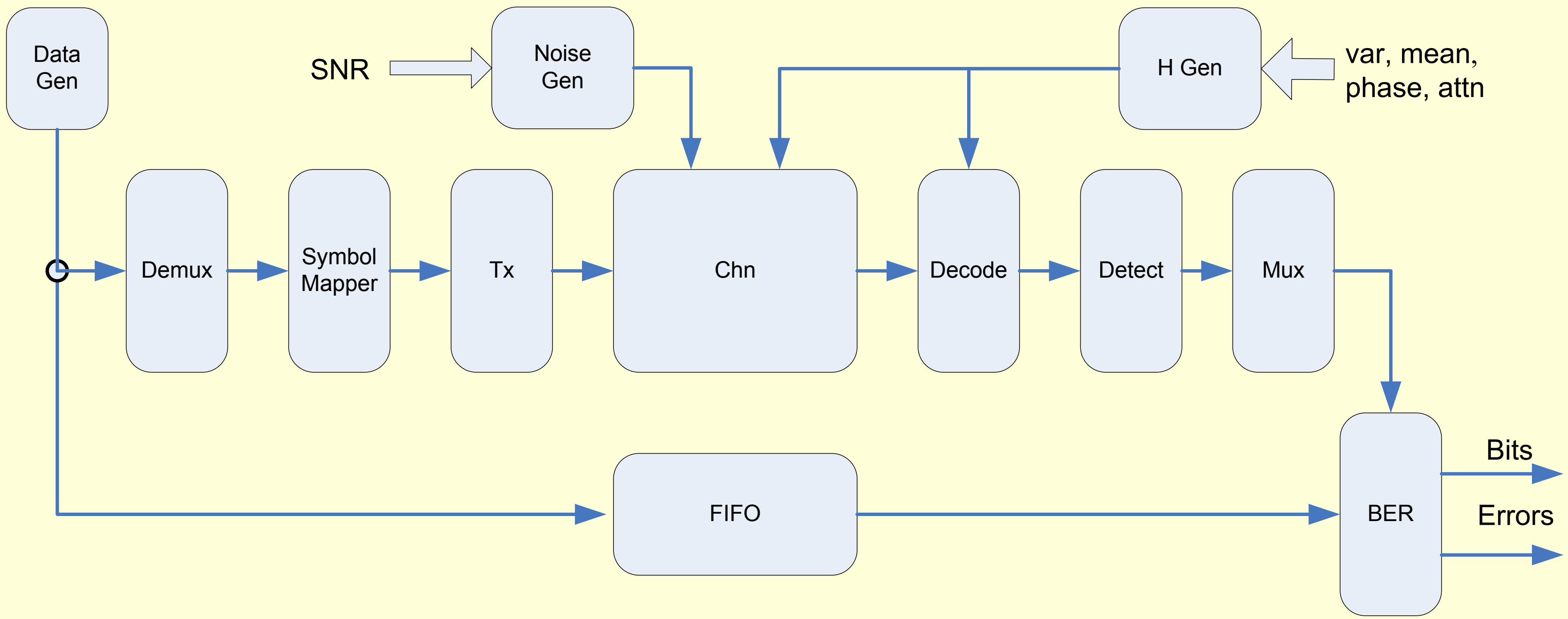
College of Engineering
TEMPLE UNIVERSITY

Development

- Tools
  - Microsoft Visual Studio 2008
  - Microsoft .NET 3.5 Framework
  - Hydesoft Computing Dplot (only post plotting)
- Characteristics
  - Visual C#
  - IEEE 754 64-bit floating point number representation
  - Only basic optimization
  - Uses inverse transform for Gaussian random variables
  - Uses incremental decoding

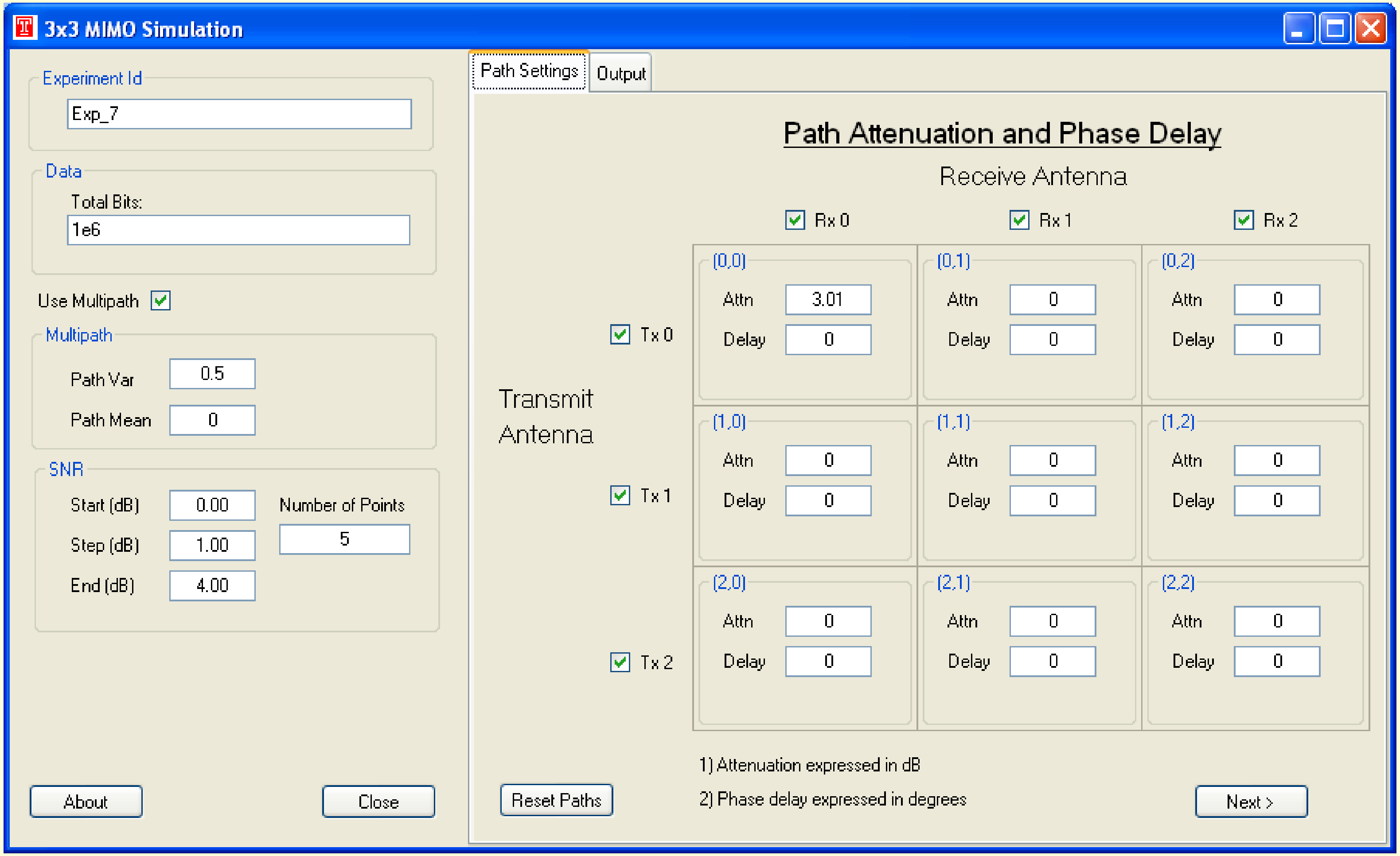
College of Engineering
TEMPLE UNIVERSITY

Model Architecture

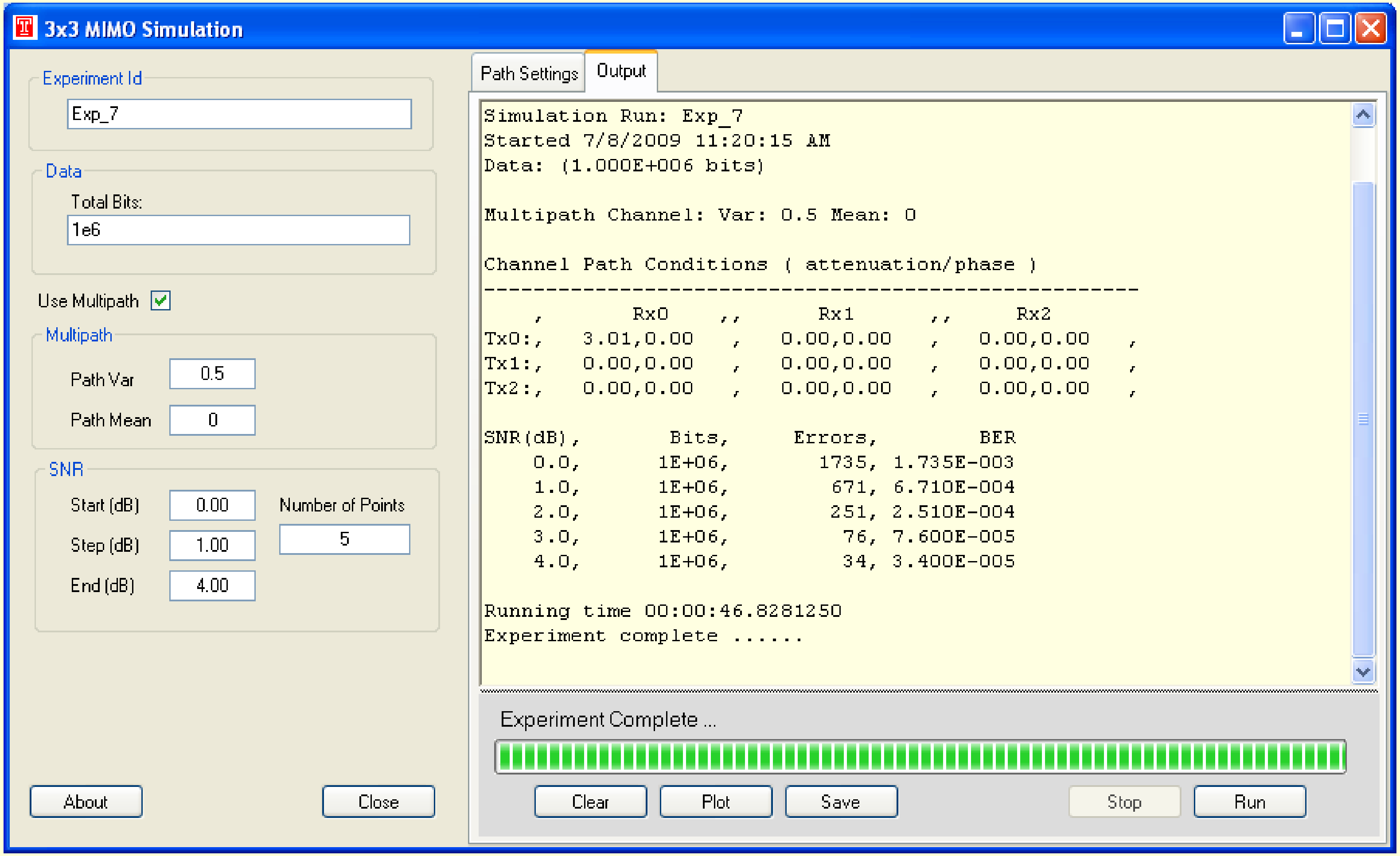


College of Engineering
TEMPLE UNIVERSITY

Path Configuration



Results Display



Hardware Model

College of Engineering
TEMPLE UNIVERSITY

Purpose of Hardware Model

- Performance (Processing Rate)
  - Better BER precision
  - Greater BER versus SNR curve resolution
  - Evaluation under multiple conditions
- Proof of practicality
  - Methods are practical if the predicted MIMO system performance can be achieved using available technology in real-time

Development

- MATLAB/Simulink
  - Development environment
  - Provides user interface and scripting
- Xilinx System Generator for DSP
  - Generates PGA programming from model
  - Provides means to load PGA programming
  - Provides means to set simulation parameters
  - Provides means to retrieve simulation results
- Xilinx ML402 (Vertex-4 FGPA)

Hardware Platform

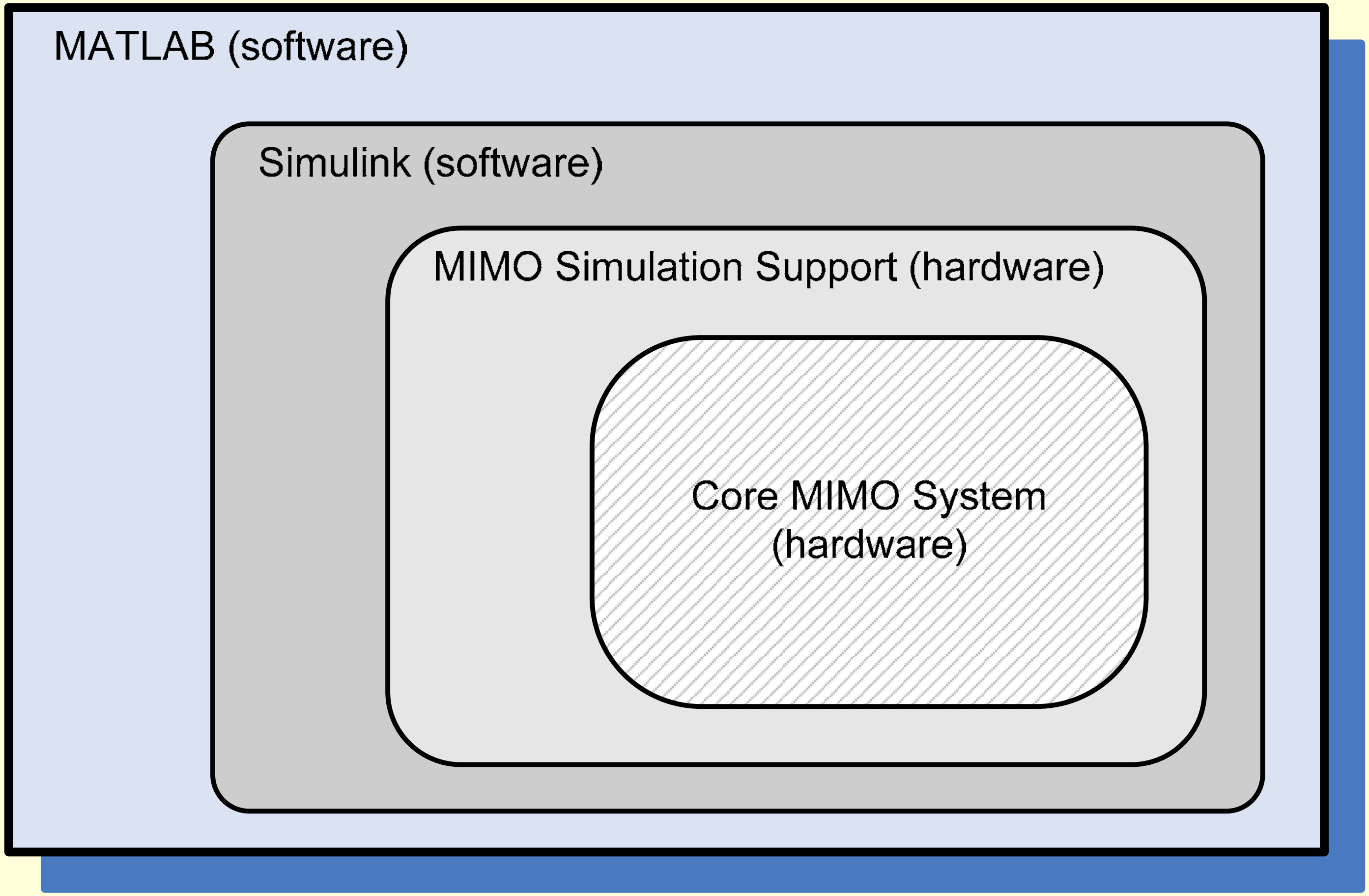
- Target PGA device is Xilinx Vertex 4 SX
- Vertex 4 features
  - 18x18 bit multiply accumulate blocks
  - Block RAM
  - Configurable logic blocks (CLB)
- Xilinx ML402 evaluation platform
  - Vertex 4 XC4VSX35
  - Ethernet interfaces

Model Characteristics

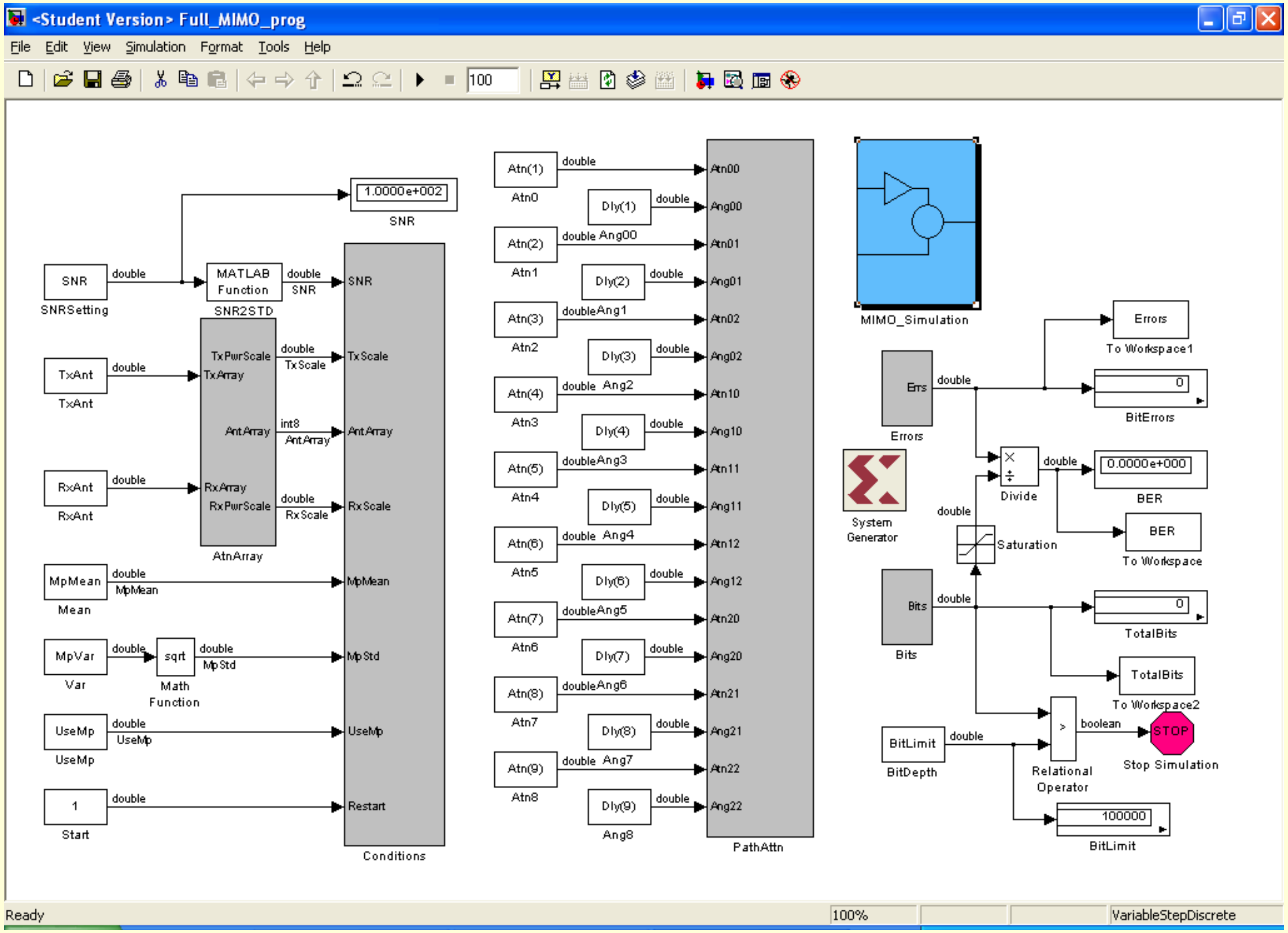
- Fixed point number representation
- Skip-ahead LFSR for uniform random values
- Composite lookup table inverse transform for Gaussian random values
- Incremental linear decoding

College of Engineering TEMPLE UNIVERSITY

Nested Architecture

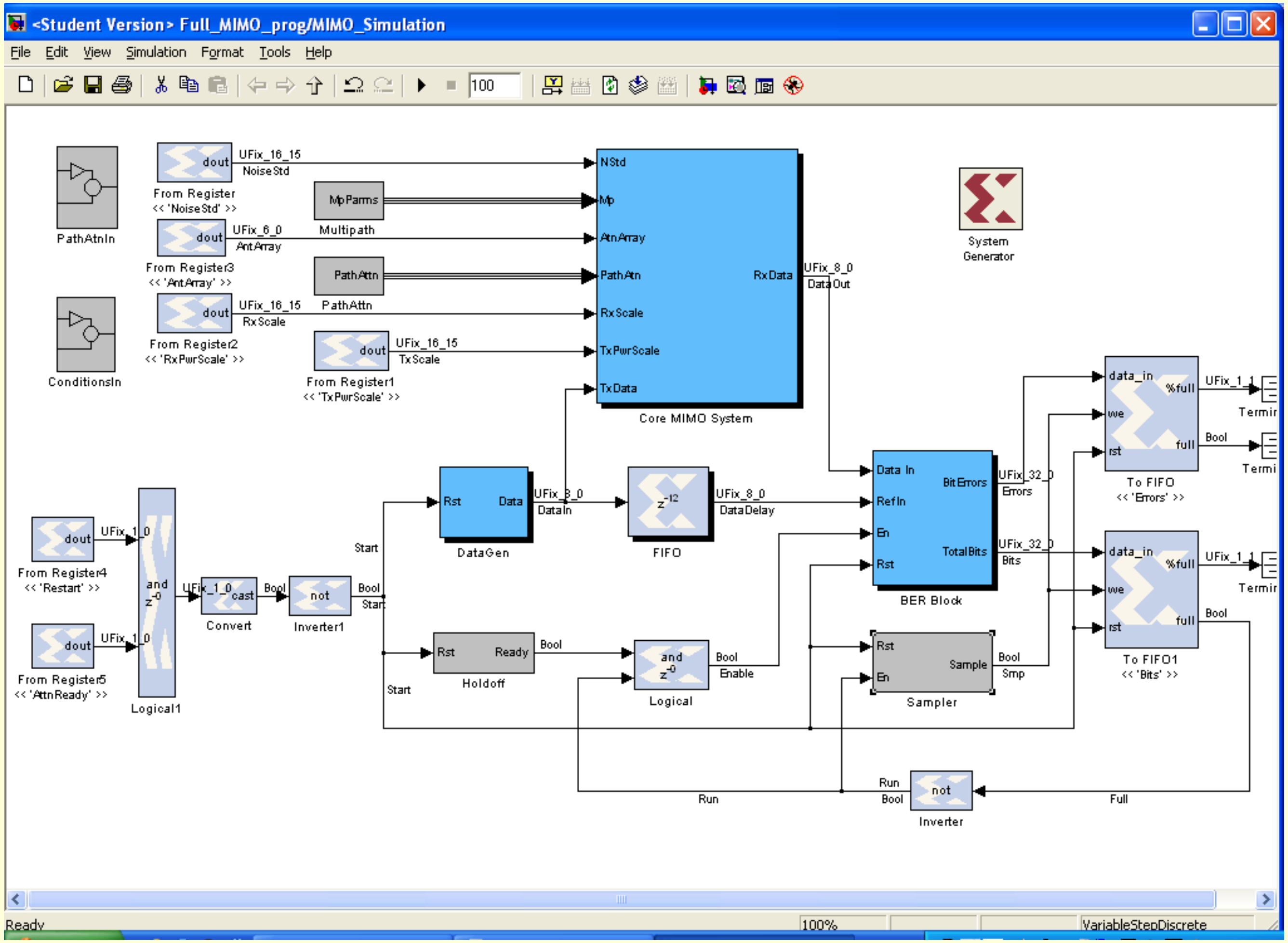


Simulink Layer



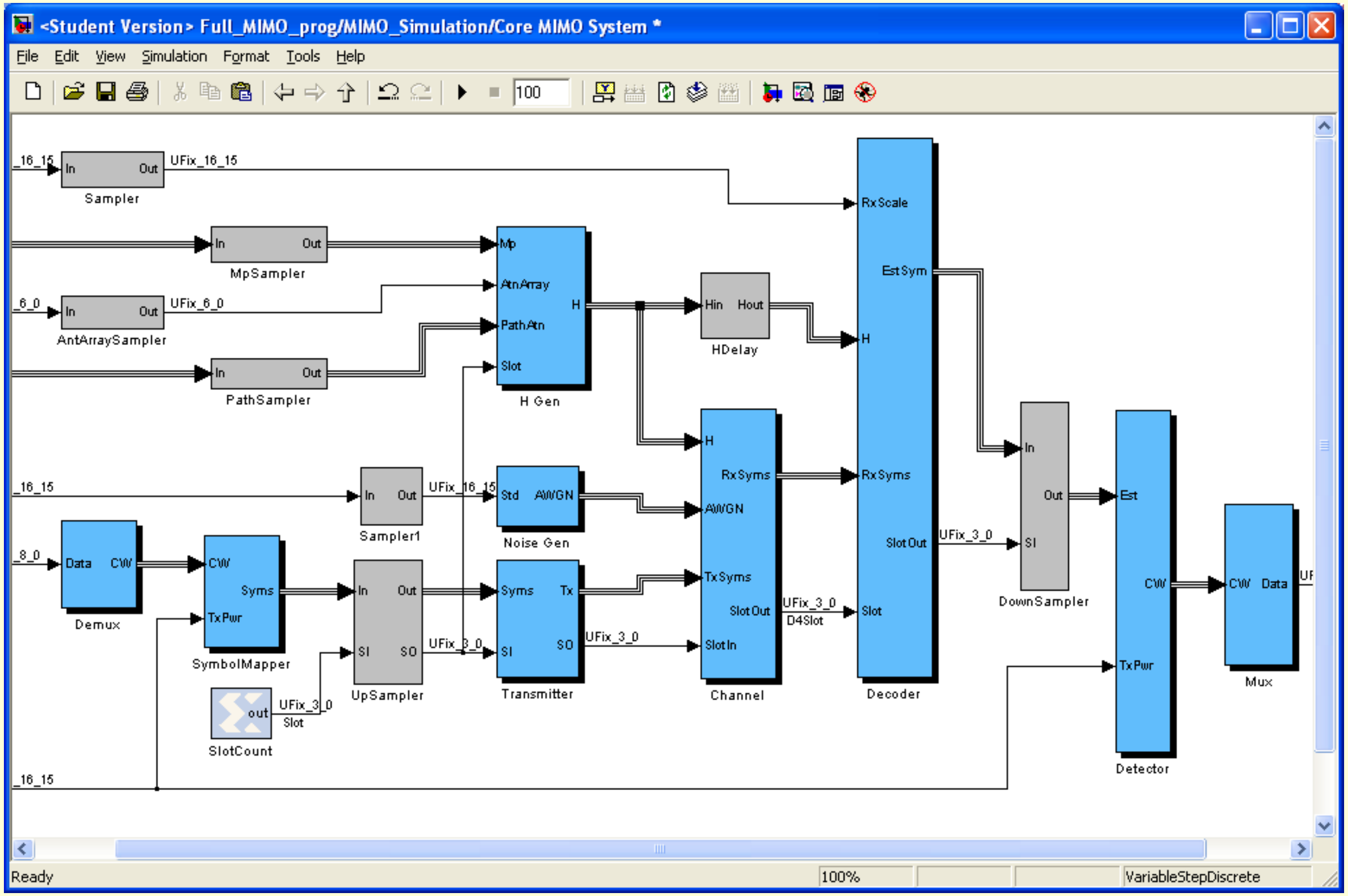
College of Engineering
TEMPLE UNIVERSITY

Simulation Support Hardware



College of Engineering
TEMPLE UNIVERSITY

Core MIMO System



Simulation Results

```
Simulation Run: HW-Case1-Tx0-Rx0
Started 9/7/2009 14:56:43
Minimum bits: 3.000000e+007

Multipath Channel: none
Channel Path Conditions (attenuation/phase)
--------------------------------------------
,      Rx0    ,,      Rx1    ,,    Rx2
Tx0:,  3.01,  0.00, -----,----- , -----,----- ,
Tx1:, -----,----- , -----,----- , -----,----- ,
Tx2:, -----,----- , -----,----- , -----,----- ,
SNR(dB),        Bits,  Errors,     BER
0.0,  3.001e+007, 2350879,7.835e-002
1.0,  3.001e+007, 1681490,5.604e-002
2.0,  3.001e+007, 1120005,3.733e-002
3.0,  3.001e+007,  682912,2.276e-002
4.0,  3.001e+007,  371572,1.238e-002
5.0,  3.001e+007,  176501,5.882e-003
6.0,  3.001e+007,   70853,2.361e-003
7.0,  3.001e+007,   22563,7.520e-004
8.0,  3.001e+007,    5632,1.877e-004
9.0,  3.001e+007,    1011,3.369e-005
Run time(sec),  173.277
Run bits     , 3.001e+008
Ave (nsec/bit) ,  577.481

Experiment complete ...
```

Significant Implementation

Requirements

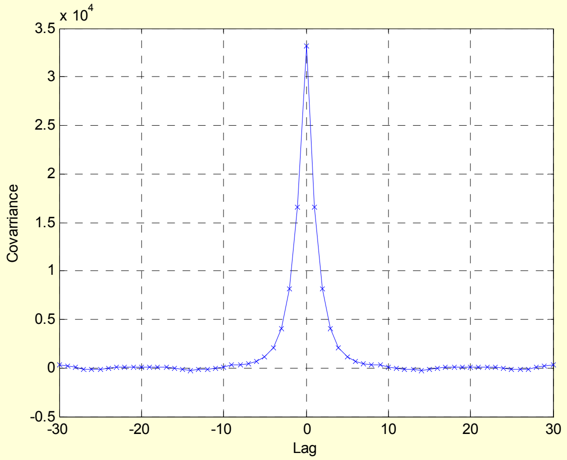
- Minimize hardware resource utilization
- Maintain processing rate
- Maintain sufficient accuracy to pass validation tests

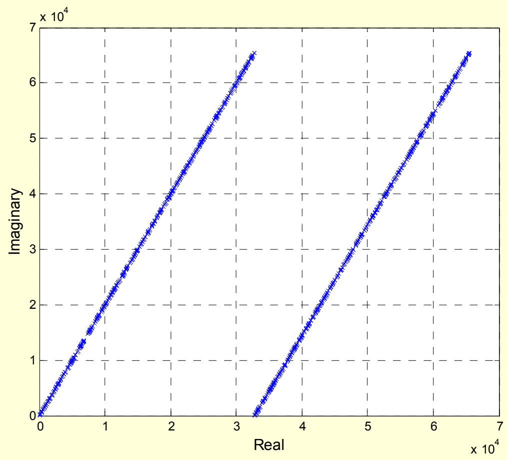
Skip-Ahead LFSR

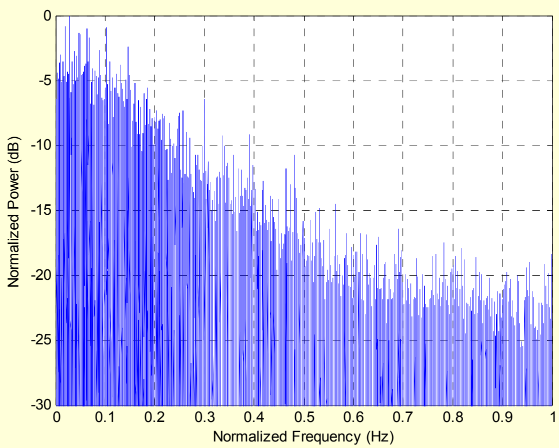
- All LFSR have correlation between successive values that affect simulation results when white randomness is assumed
- Skip-ahead technique advances LFSR *k* states in one step to avoid correlation and whiten random values
- Skip-ahead increases feedback network complexity
- Rules developed for minimizing feedback network complexity

Without Skip-Ahead

- Non-delta autocovariance
- Patterns when adjacent pairs are ploted
- Lowpass frequency characteristic

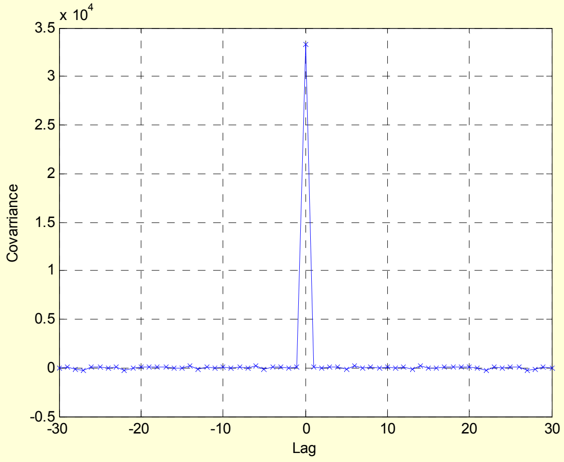


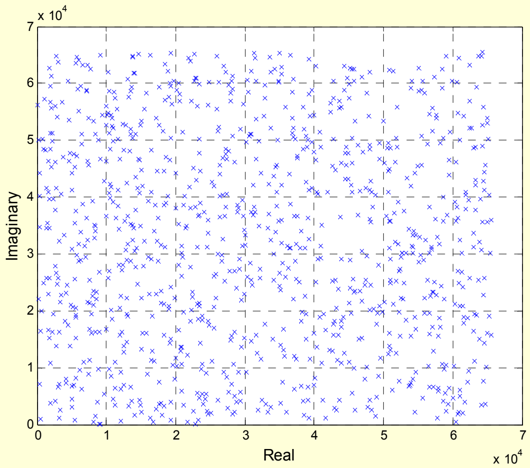


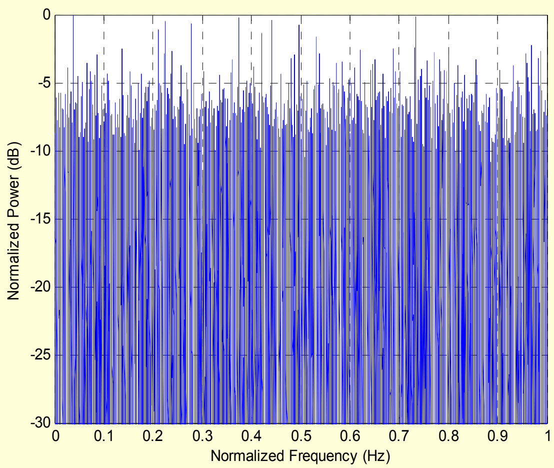


With Skip-Ahead

- Autocovariance is delta function
- No patterns in plots of adjacent pairs
- White frequency characteristic





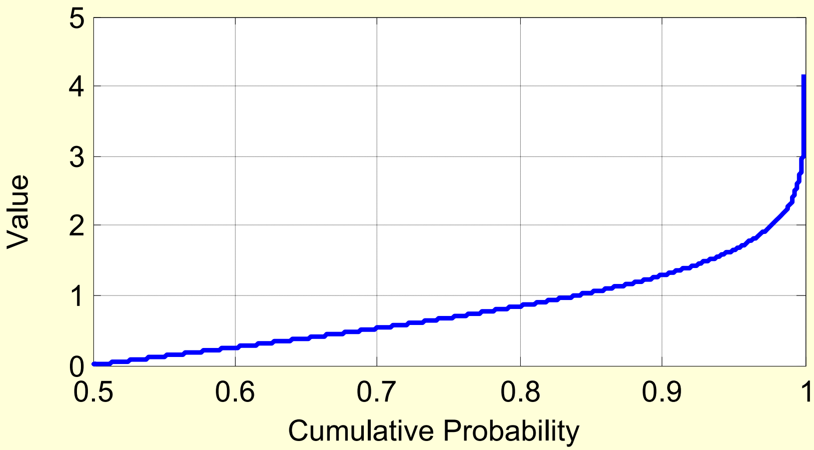


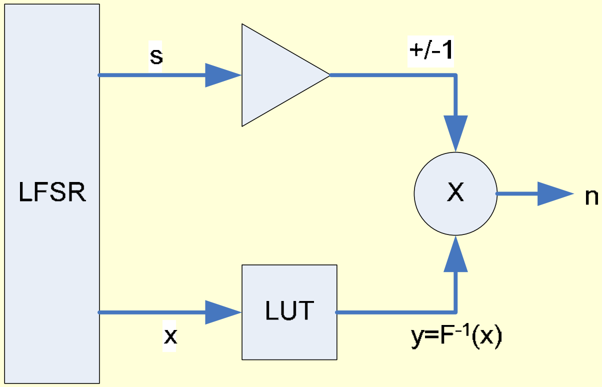
Composite Lookup Table

- Inverse transform method produces Gaussian random values from uniform random values
- Requires only basic mathematical functions
- Maintains processing rate of model
- Requires minimum hardware resource
- Can be designed to limited maximum error
- Rules developed for design of composite LUT

Basic Lookup Table (LUT)

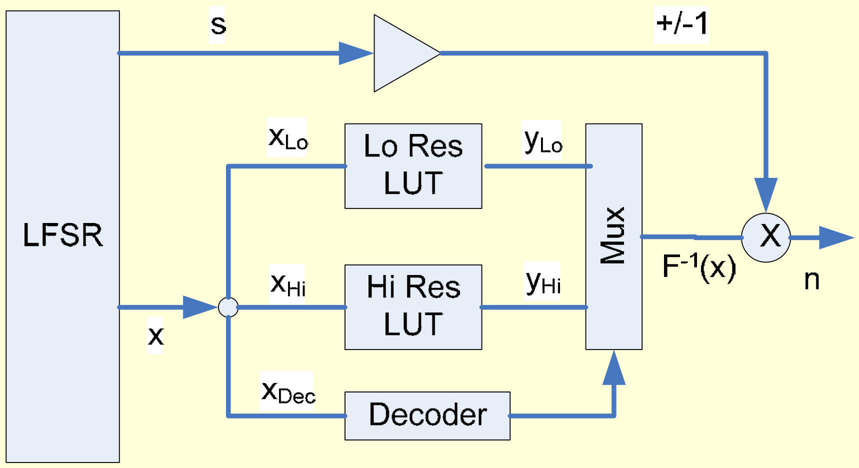
- LFSR produces 16-bit uniform values
- 1-bit used for sign, s
- 15-bits used to address LUT
- LUT holds Gaussian cumulative probability function





Composite Lookup Table

- LUT split into high and low resolution
- Decoder selects between tables
- Decoder is a single AND gate
- High and low tables addressed by partition of input bits
- Total table size is less than single LUT



College of Engineering
TEMPLE UNIVERSITY

Incremental Decoding

- Reduces hardware resource requirements for STBC decoder over a direct implementation
- Performs decoding incrementally over the eight timeslots of the code block transmission
- Maintains the processing rate of the model

Direct Decoder Implementation

- Receive signals r_k must be stored for all 8 timeslots
- All $\tilde{x}_i$ must be computed in one step

$$\tilde{x}_0 = \sum_{k=0}^{2} \left[r_k^{(0)} h_{0k}^* + r_k^{(1)} h_{1k}^* + r_k^{(2)} h_{2k}^* + \left(r_k^{(4)}\right)^* h_{0k} + \left(r_k^{(5)}\right)^* h_{1k} + \left(r_k^{(6)}\right)^* h_{2k} \right]$$

$$\tilde{x}_1 = \sum_{k=0}^{2} \left[r_k^{(0)} h_{1k}^* - r_k^{(1)} h_{0k}^* + r_k^{(3)} h_{2k}^* + \left(r_k^{(4)}\right)^* h_{1k} - \left(r_k^{(5)}\right)^* h_{0k} + \left(r_k^{(7)}\right)^* h_{2k} \right]$$

$$\tilde{x}_2 = \sum_{k=0}^{2} \left[r_k^{(0)} h_{2k}^* - r_k^{(2)} h_{0k}^* - r_k^{(3)} h_{1k}^* + \left(r_k^{(4)}\right)^* h_{2k} - \left(r_k^{(6)}\right)^* h_{0k} - \left(r_k^{(7)}\right)^* h_{1k} \right]$$

$$\tilde{x}_4 = \sum_{k=0}^{2} \left[- r_k^{(1)} h_{2k}^* + r_k^{(2)} h_{1k}^* - r_k^{(3)} h_{0k}^* - \left(r_k^{(5)}\right)^* h_{2k} + \left(r_k^{(6)}\right)^* h_{1k} - \left(r_k^{(7)}\right)^* h_{0k} \right]$$

Incremental Decoding

- Compute only three p_i or three q_i each timeslot
- Storage only one accumulator for each $\tilde{x}_i$
- Estimates accumulated according to table below
- Estimates complete after 8 timeslots

$$p_i = \sum_{k=0}^{2} r_k^{(t)} h_{ik}^*$$

$$q_i = \sum_{k=0}^{2} \left(r_k^{(t)}\right)^* h_{ik}$$

t	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)
$\tilde{x}_0$	$+p_0$	$+p_1$	$+p_2$	–	$+q_0$	$+q_1$	$+q_2$	–
$\tilde{x}_1$	$+p_1$	$-p_0$	–	$+p_2$	$+q_1$	$-q_0$	–	$+q_2$
$\tilde{x}_2$	$+p_2$	–	$-p_0$	$-p_1$	$+q_2$	–	$-q_0$	$-q_1$
$\tilde{x}_3$	–	$-p_2$	$+p_1$	$-p_0$	–	$-q_2$	$+q_1$	$-q_0$

Direct vs. Incremental Method

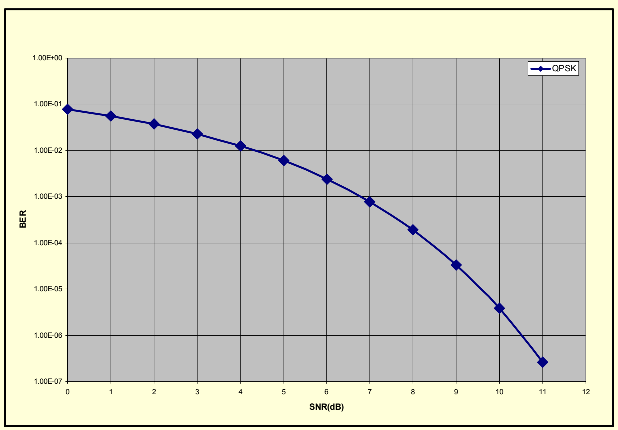
	Direct Method	Incremental Method
Storage Elements	48	8
Simple Multiplications	288	36
Simple Additions	280	36

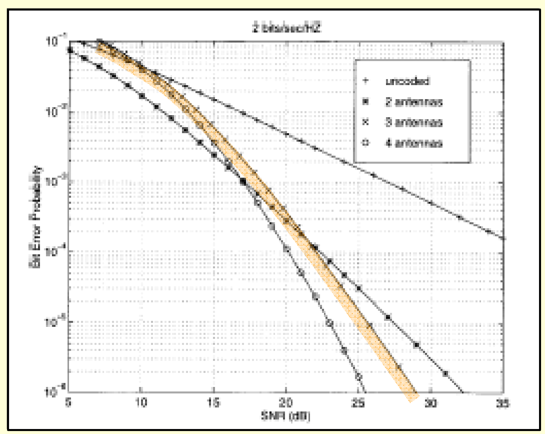
College of Engineering
TEMPLE UNIVERSITY

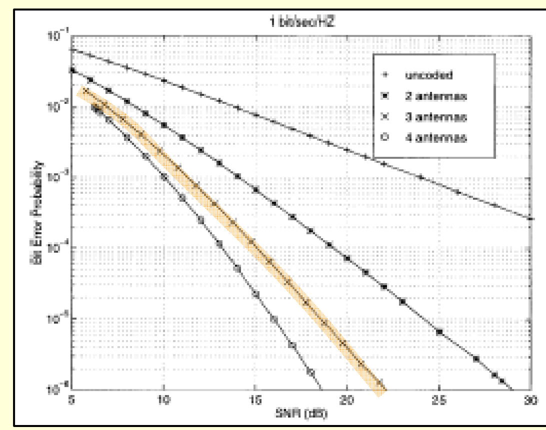
Validation

Validation Cases

- Model validation accomplished by reproducing published BER verse SNR data
  - One transmit to one receive antenna (Proakis, 1995)
  - Three transmit to one receive antenna (Tarokh, et. al. 1999)
  - Three transmit to two receive antenna (Tarokh, et. al. 1999)

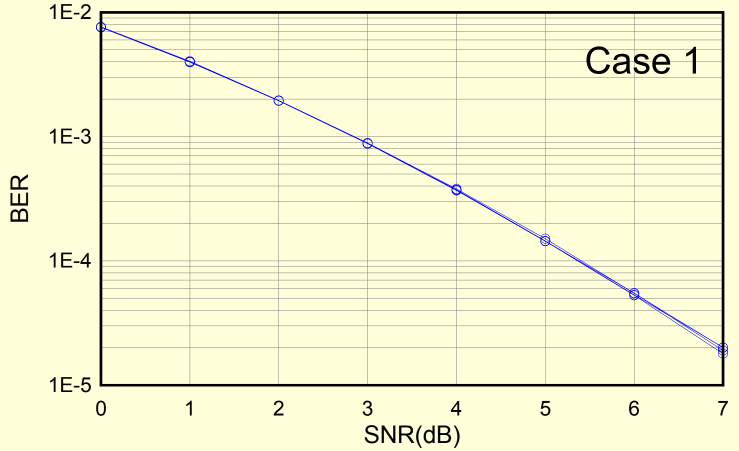


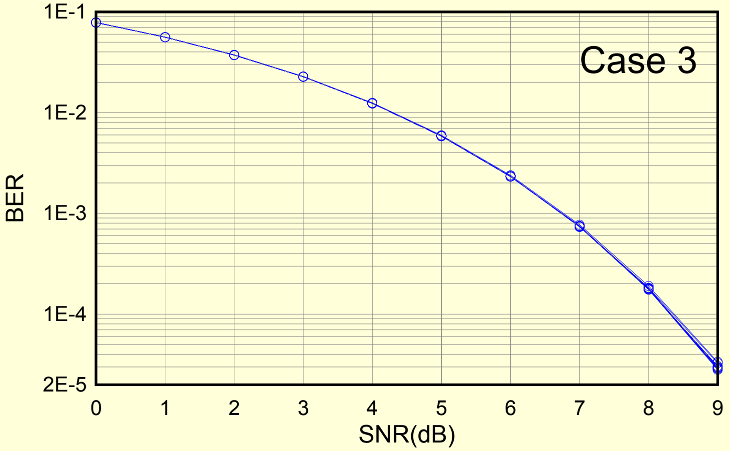


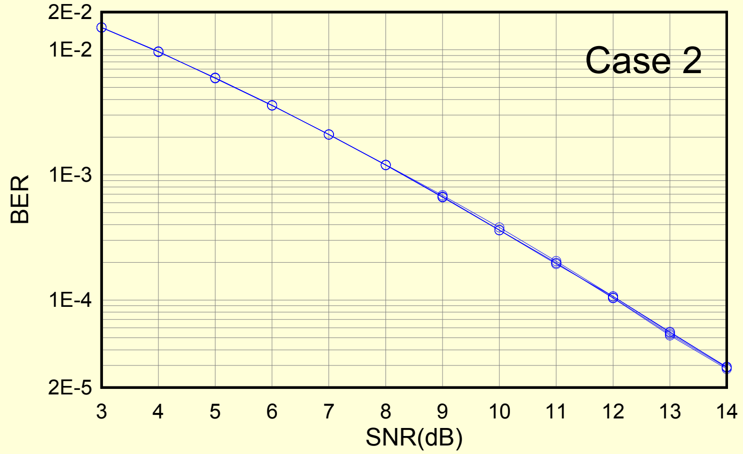


Software Model Validation

- Overlaying validation curves produced by the model with reference curves shows close match



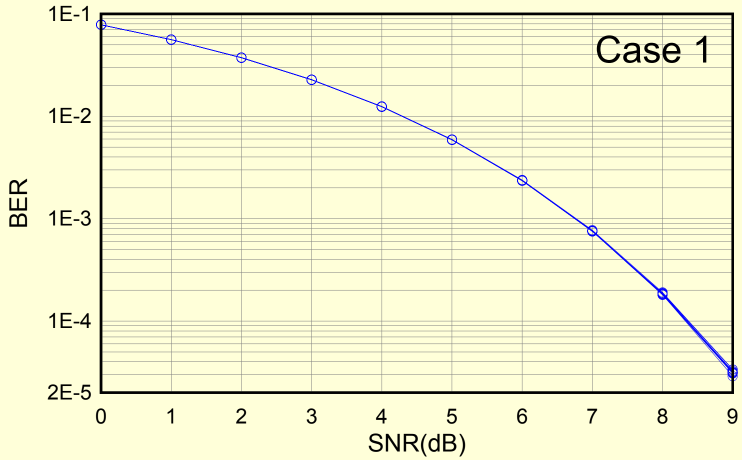


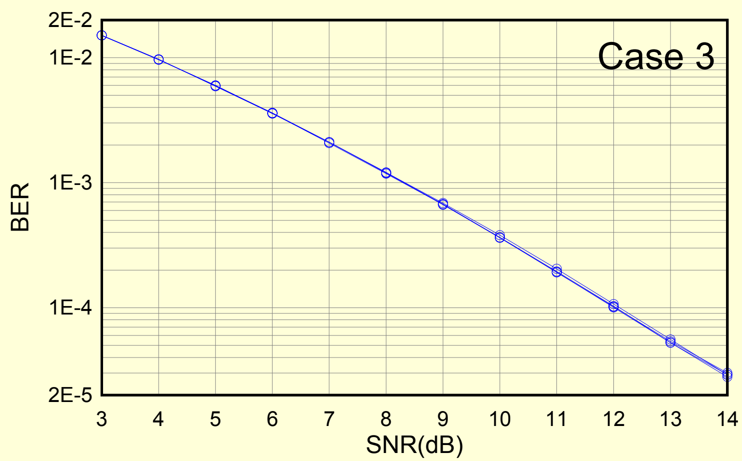


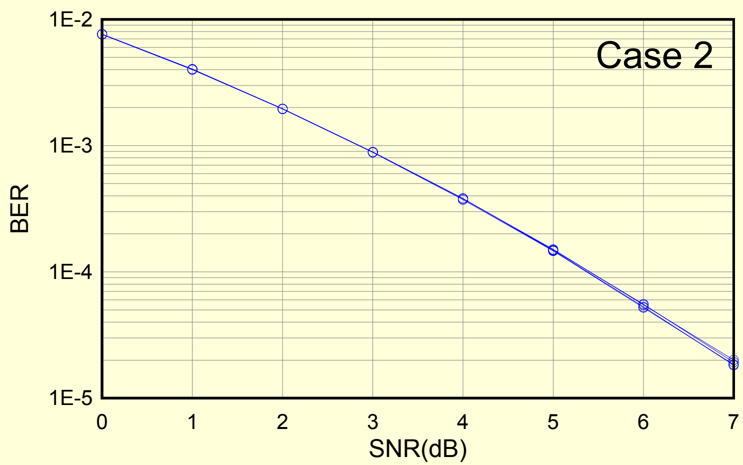
College of Engineering
TEMPLE UNIVERSITY

Hardware Model Validation

- Overlay of curves produced by hardware model also show close match to reference curves







College of Engineering
TEMPLE UNIVERSITY

Performance

College of Engineering
TEMPLE UNIVERSITY

Processing Rate

		Hardware Model	Software Model
Case 1	Bit Rate (nsec)	126	2948
	Overhead (sec)	11.2	1.0
Case 2	Bit Rate (nsec)	125	3119
	Overhead (sec)	11.2	0.9
Case 3	Bit Rate (nsec)	123	3216
	Overhead (sec)	11.2	0.9

* Average processing time per bit to simulate 10 point BER versus SNR curve

College of Engineering
TEMPLE UNIVERSITY

Resource Utilization

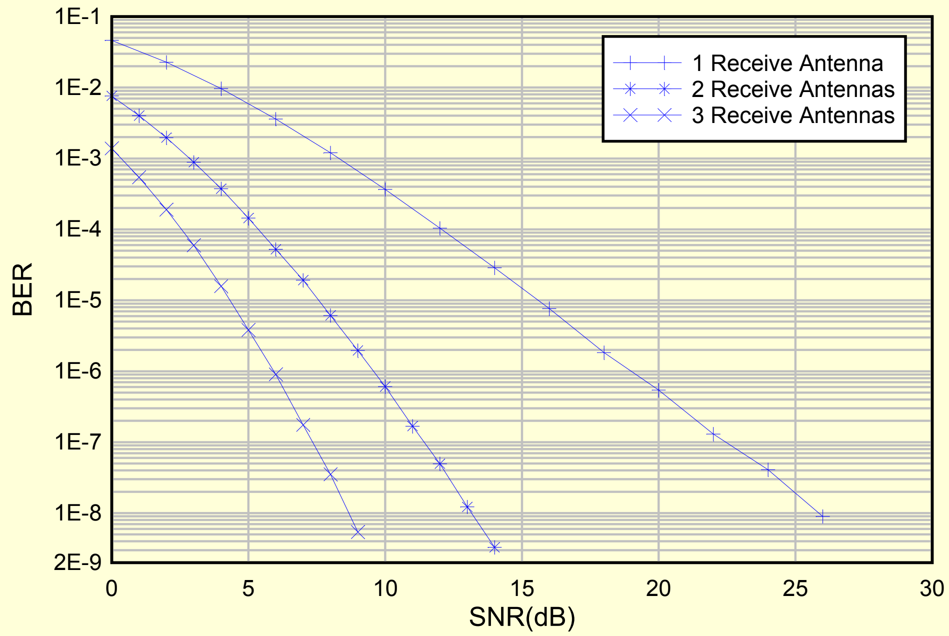
Resource	Available	Initial		Skip-Ahead LFSR		Final	
DSP48	192	153	79%	149	77%	149	77%
RAM16B	192	170	89%	166	86%	58	30%
Total Slices	15360	12900	84%	12966	84%	12756	83%
SliceM	7680	2590	33%	2701	35%	2599	34%

1. Applies to hardware model only
2. Shows number of resource units and percent of units available

Usage

Extended Curves

- BER versus SNR performance beyond original published curves
- Additional unpublished 3 receive antenna curve
- Hardware model performs simulations in minutes instead of hours

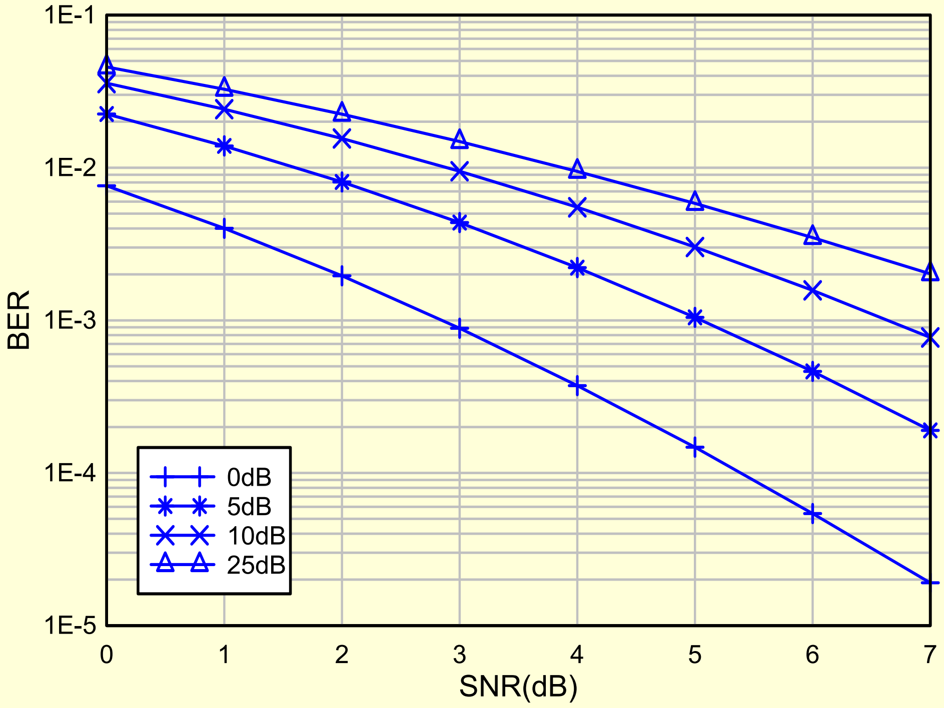


	1 Receive Antenna	2 Receive Antennas	3 Receive Antenna
Total Bits	1.2×10^{10}	1.5×10^{10}	1.1×10^{10}
Hardware Simulation Time	27 min	36 min	26 min
Software Simulation Time (Estimated)	10.2 hours	13.8 hours	9.5 hours

College of Engineering
TEMPLE UNIVERSITY

Progressive Fading

- 3 Transmit and 2 Receive antennas
- One receive antenna sees obstructed signals
- Signal attenuation increases for each curve (left to right)
- Note that greatest performance lost with first attenuation step



Summary

- Developed both software and hardware MIMO system models
- Developed method to efficiently whiten LFSR in PGA
- Developed method to reduce Gaussian cdf LUT size
- Developed incremental linear STBC decoding method
- Demonstrated advantages of hardware accelerated model in MIMO simulation
- Demonstrated method of controlled fading conditions
- Demonstrated viability of linearly decoded STBC MIMO systems using currently available technology

College of Engineering
TEMPLE UNIVERSITY

Questions ?

College of Engineering
TEMPLE UNIVERSITY®

Thank you