Analysis of SEU-induced Errors in an FPGA-based Digital Communications System

Brian Pratt, Michael Wirthlin

Brigham Young University

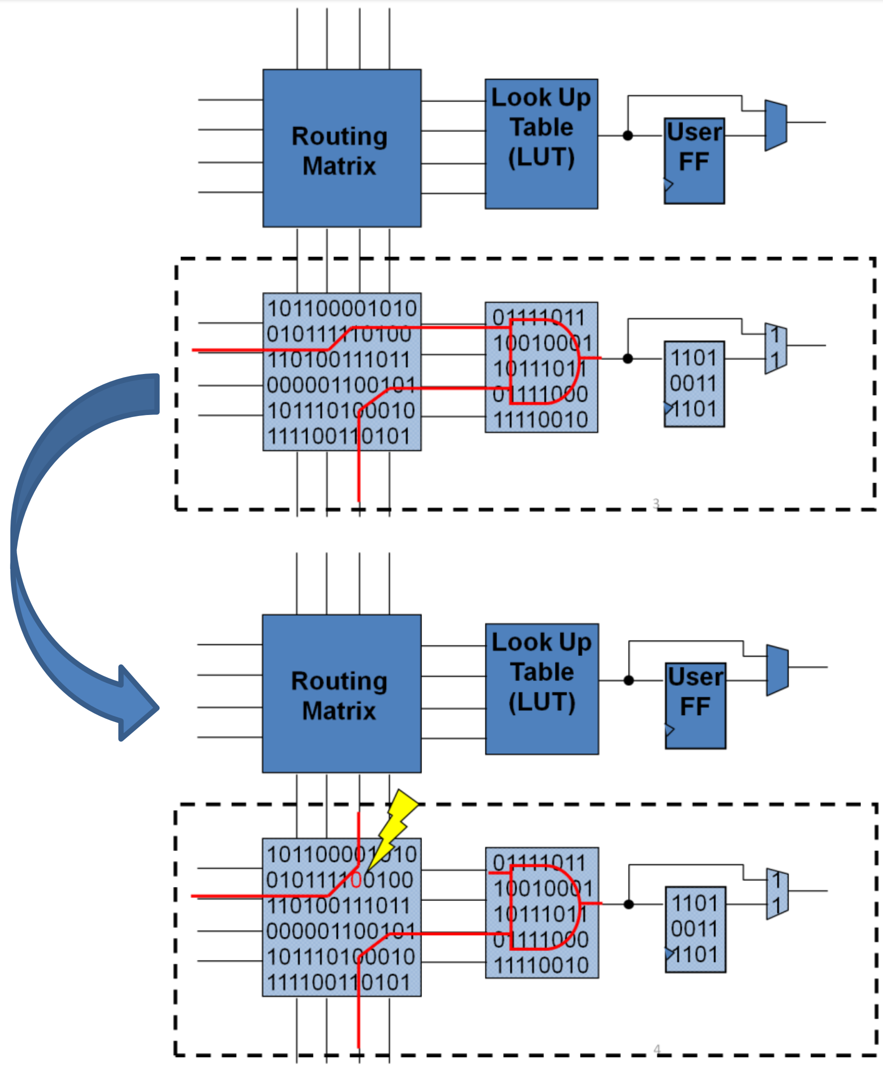
Michael Caffrey, Paul Graham, Keith Morgan

Los Alamos National Laboratory

This work was supported in part by the I/UCRC Program of the National Science Foundation under the NSF Center for High-Performance Reconfigurable Computing (CHREC). Approved for public release by Los Alamos National Laboratory under LA-UR-08-05615; distribution is unlimited.

FPGA Reliability

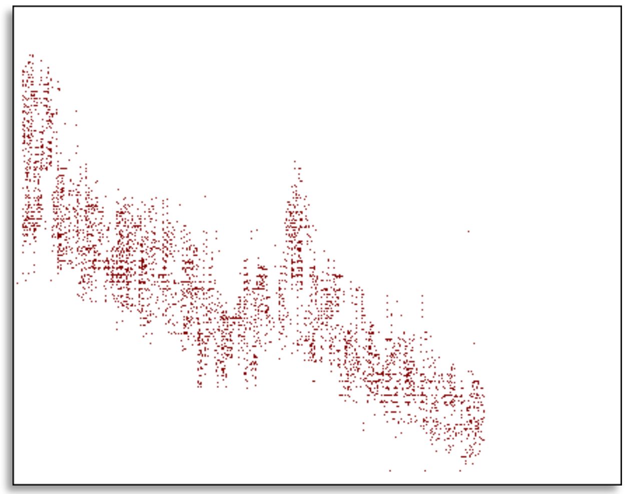
- FPGAs are susceptible to radiation-induced single-event upsets (SEUs)
- SEUs can change the hardware implemented or the contents of user memory



Dynamic Cross-section

FPGA

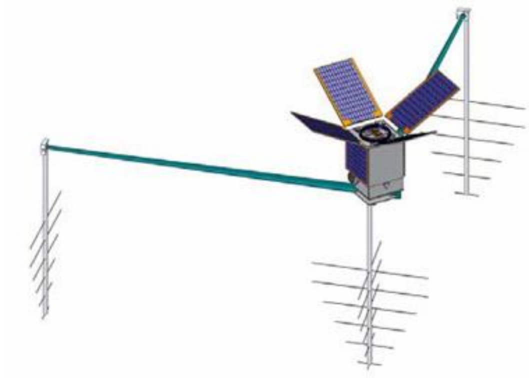
Virtex 1000

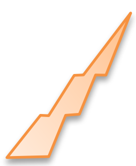


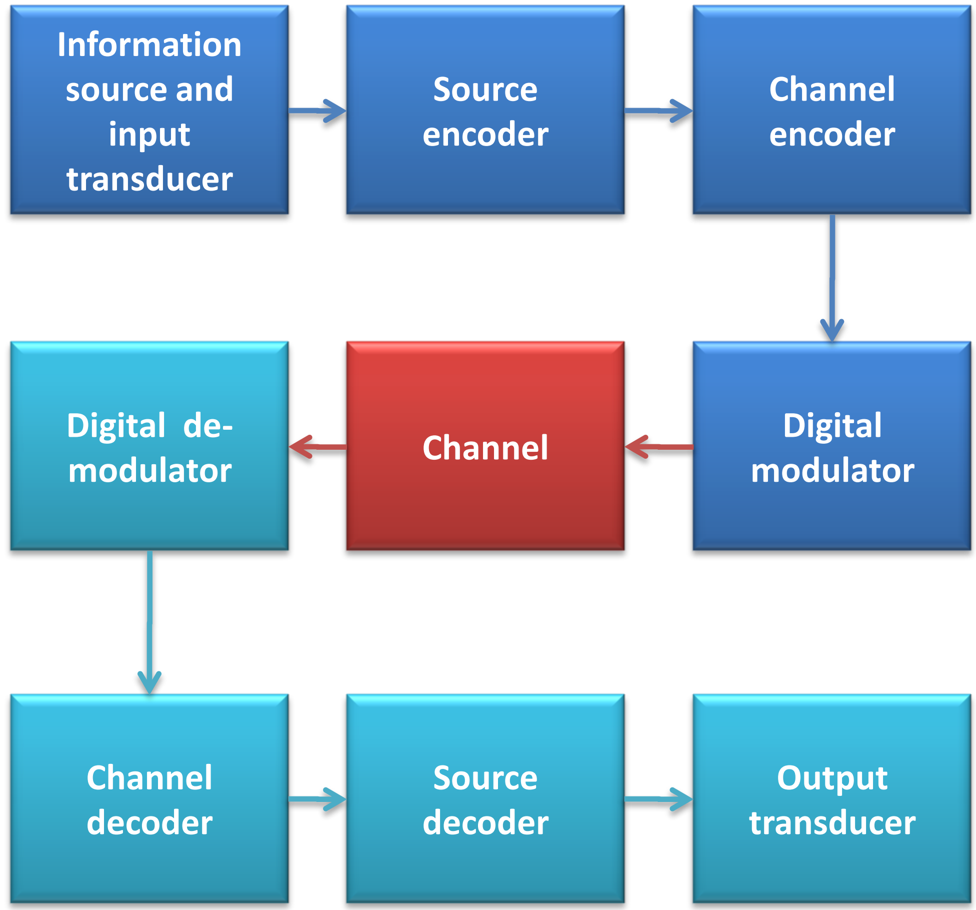
Full FPGA (static cross section):
12,288 Slices
5.8 Million configuration bits

FIR Filter (dynamic cross section):
1,869 Slices
149,696 configuration bits

Digital Communications

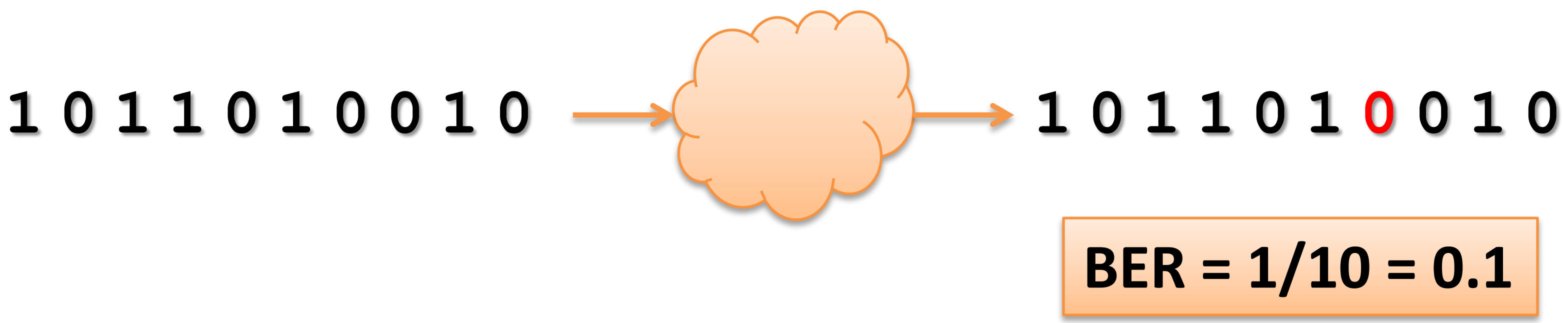






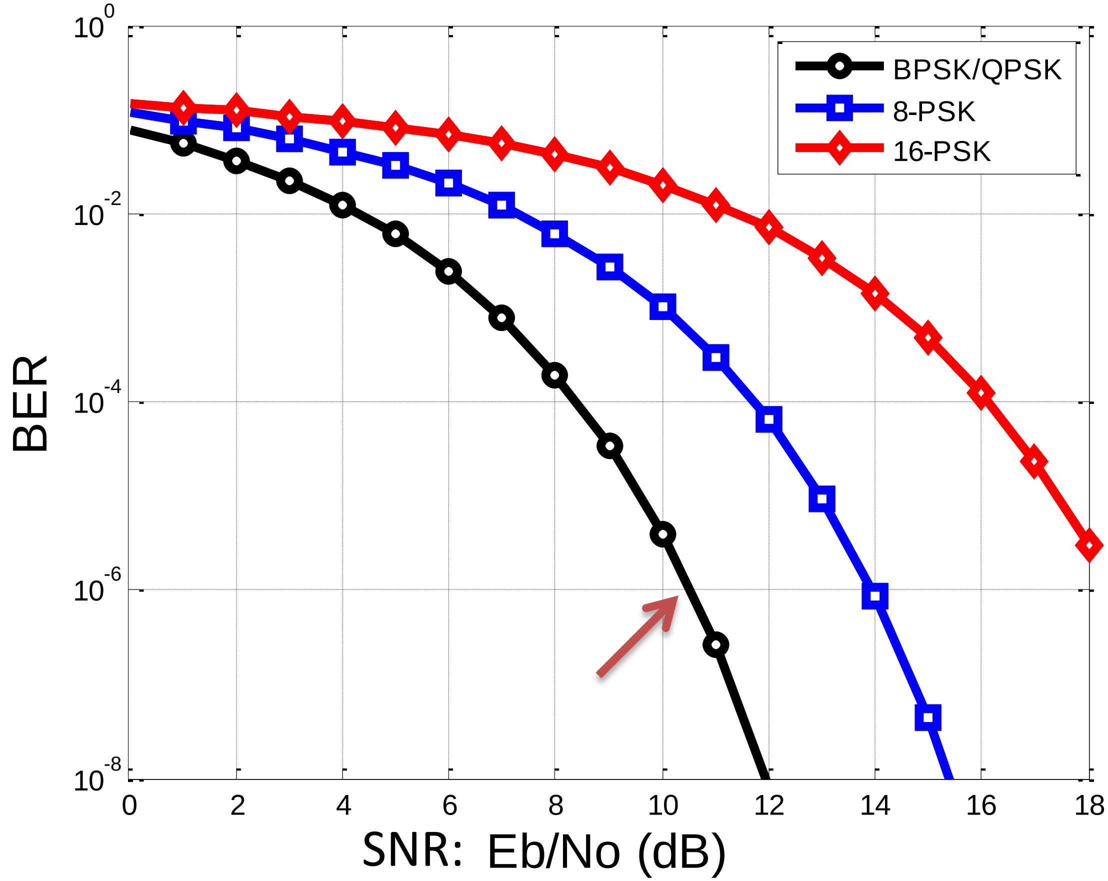
Source: Proakis, Digital Communications

Bit Error Rate (BER)



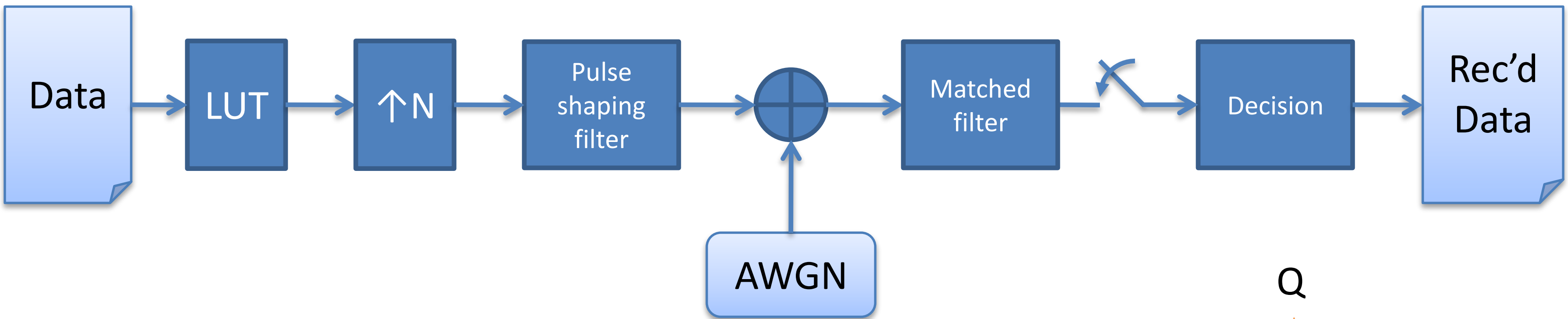
- The measure of performance of a digital communications system
- Also called the Bit Error Ratio
- BER is defined as the number of erroneously-decoded bits divided by the total number of bits sent

Sample BER Curves

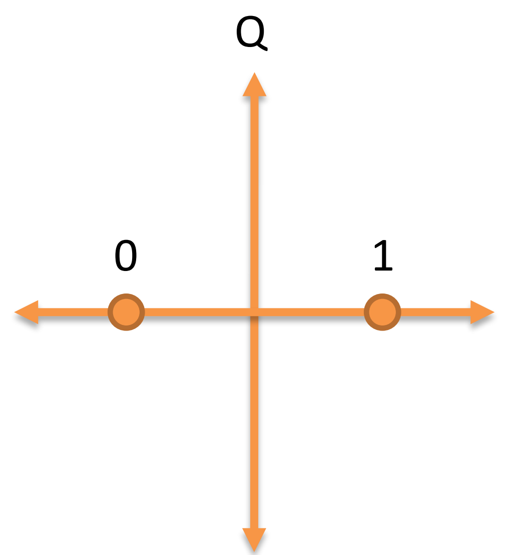


Example: For QPSK, 1 bit error per million message bits at SNR (E_b/N_o) of 10.6 dB

BPSK System

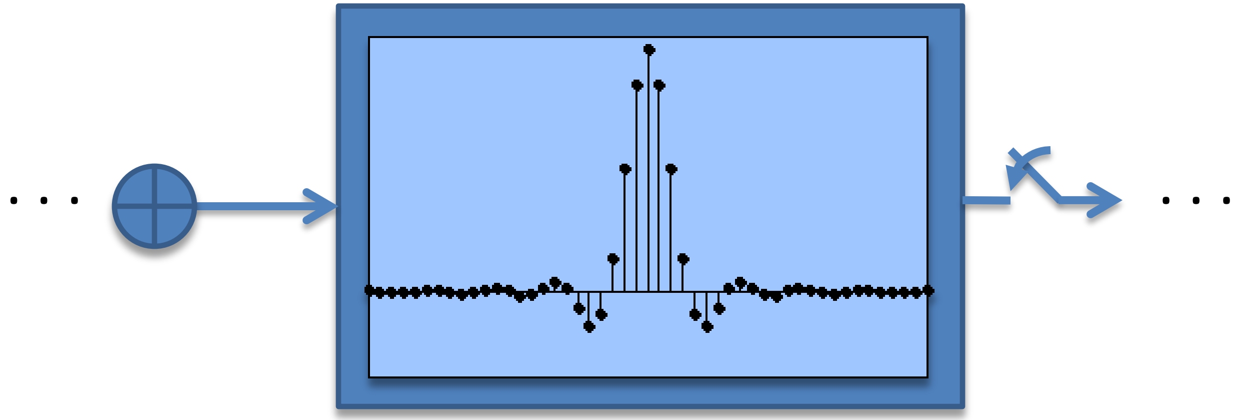


- BPSK = Binary Phase-Shift Keying
- Very simple system, also called binary PAM
- Similar to popular QPSK (which is only slightly more complex)



BPSK bit assignments

Matched Filter – FIR Filter



- Main component in a simple BPSK/QPSK receiver
- Matched to the pulse-shaping filter on the transmitting side to maximize performance

Test Methodology

Record output of uncorrupted filter

↓

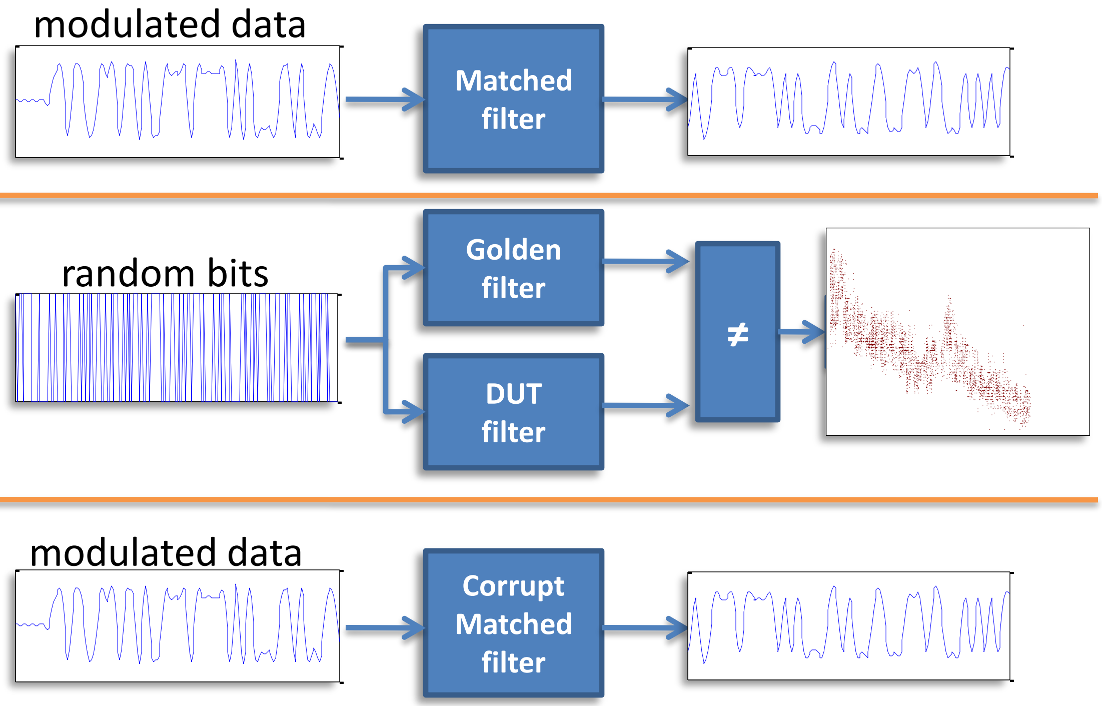
Find all *sensitive* config bits (those used by the design)

↓

Record output with each sensitive bit upset

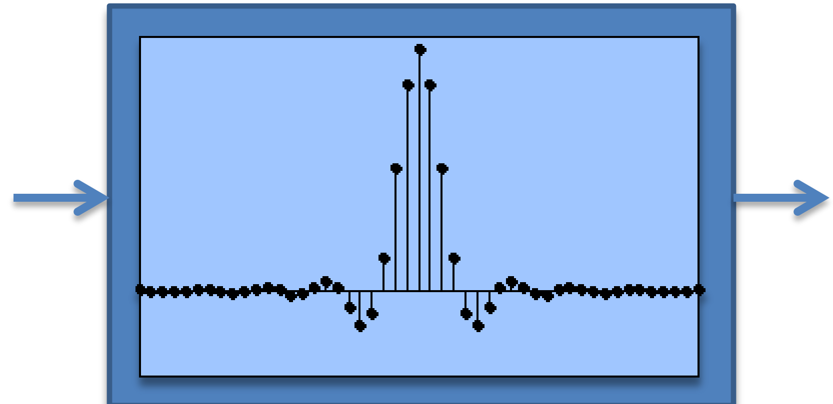
↓

Calculate loss in signal-to-noise ratio (SNR) at output



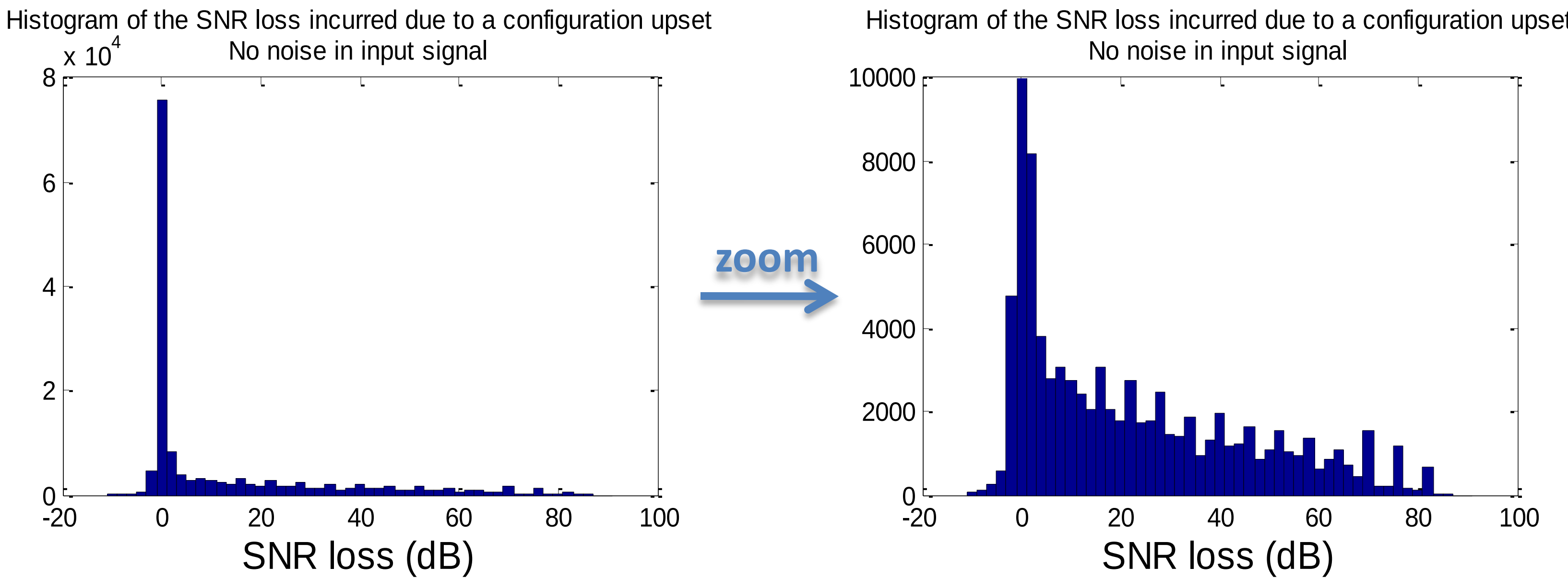

```
...
> noise = uncorrupt_out - corrupt_out;
> SNR_corrupt = power(uncorrupt_out)/power(noise);
> SNR_dB_corrupt = 10*log10(SNR);
> SNR_loss = SNR_dB_uncorrupt - SNR_dB_corrupt;
...
```

Test Design – FIR Filter



- 49 taps
- 24 multipliers (symmetric coefficients)
- Square-root raised cosine (SRRC) pulse shape with 50% rolloff
- 16-bit fixed-point input (Q2.14 format)
- 18-bit fixed-point output (Q4.14 format)
- 15% of Slices occupied on Virtex 1000 FPGA
- Total sensitive configuration bits: 149,696/5,810,024

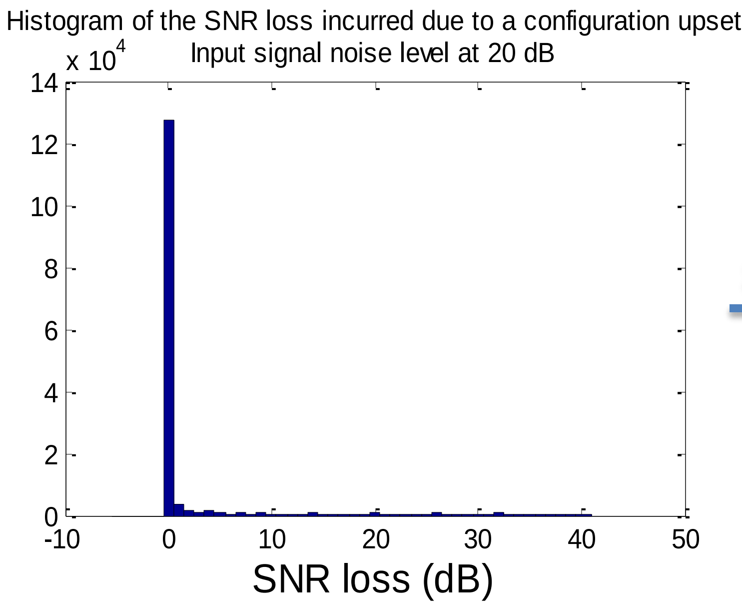
Results – No input noise



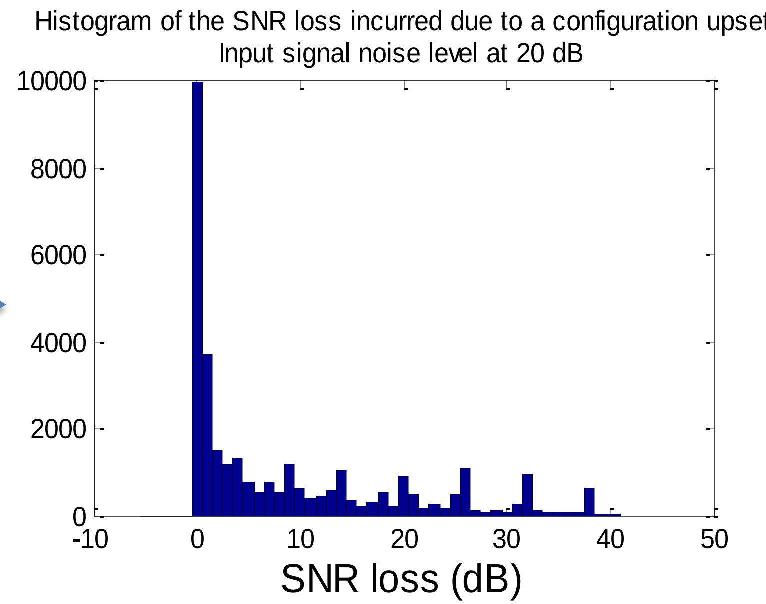
69,160 /149,696 trials reduce output SNR by less than 0.1 dB

16,443/149,696 upsets caused **no difference** in output (11%)

Results – 20 dB SNR at input



zoom



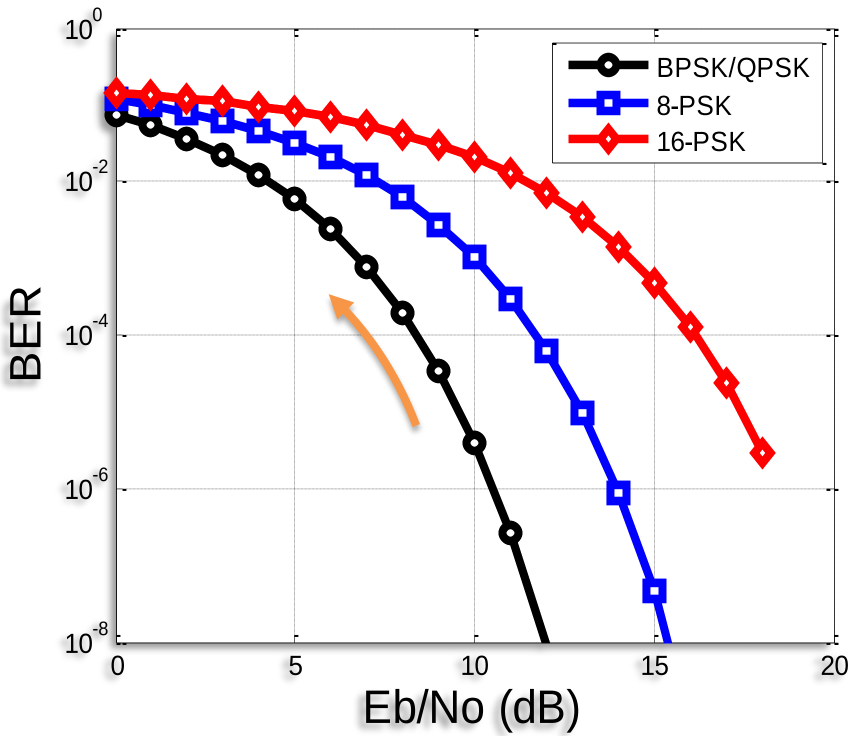
121,370 /149,696 trials reduce output SNR by less than 0.1 dB

Results Table

Input SNR	Less than 0.1dB loss in SNR	Less than 1dB loss in SNR	Less than 3dB loss in SNR	Less than 6dB loss in SNR
No noise	69,160 trials (46.2%)	81,419 trials (54.4%)	89,619 trials (59.9%)	95,134 trials (63.6%)
20 dB	121,370 trials (81.1%)	129,223 trials (86.3%)	133,441 trials (89.1%)	136,230 trials (91.0%)
10 dB	128,741 trials (86.0%)	135,997 trials (90.8%)	139,586 trials (93.3%)	142,135 trials (94.9%)
5 dB	132,484 trials (88.5%)	139,126 trials (92.9%)	142,230 trials (95.0%)	143,825 trials (96.1%)

- Total trials: 149,696
  - Number of sensitive configuration bits in the design

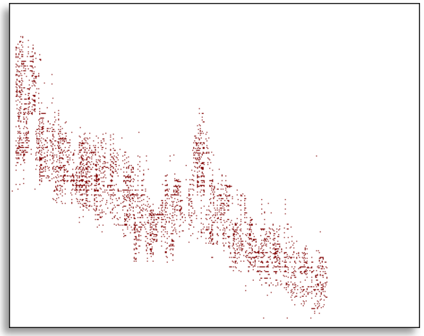
Impact of SNR Loss

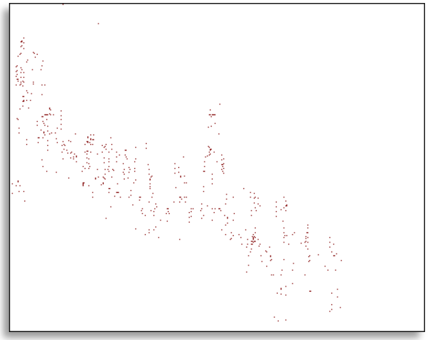


SNR level	BER for BPSK/QPSK
10.6 dB	1 in 1.2 million
10.5 dB	1 in 923,000
10.4 dB	1 in 707,000
10.3 dB	1 in 545,000
10.2 dB	1 in 422,000
10.1 dB	1 in 330,000
10.0 dB	1 in 258,000

Application-specific Cross-section

FPGA
Virtex 1000





Full FPGA
(static cross section):
12,288 Slices
5.8 Million config bits
(100%)

FIR Filter
(dynamic cross section):
1,869 Slices
149,696 config bits
(2.5%)

FIR Filter in a 20dB SNR environment tolerating 1dB additional SNR loss:
20,473 config bits
(0.35%)

Conclusions

- When designing for reliability, knowledge of the application can be *very* important
- Systems with inherent error/noise tolerance may tolerate SEU-induced upsets
- Full TMR and similar approaches may be overkill for certain systems

Future Work

- Evaluate different types of errors
  – What type of faults cause catastrophic failures?
- Evaluate lower-cost mitigation techniques
  – Partial replication
  – Error-control coding