NAND Flash Reliability and Optimization

Barry Fitzgerald

Agenda

- Introduction
  - Research group, project goals
- Flash reliability
  - Endurance/retention, test system, test process
- Machine Learning
  - Genetic Algorithms, Genetic Programming
  - Application to NAND Flash Modeling/Optimization
- Research-to-date & future work

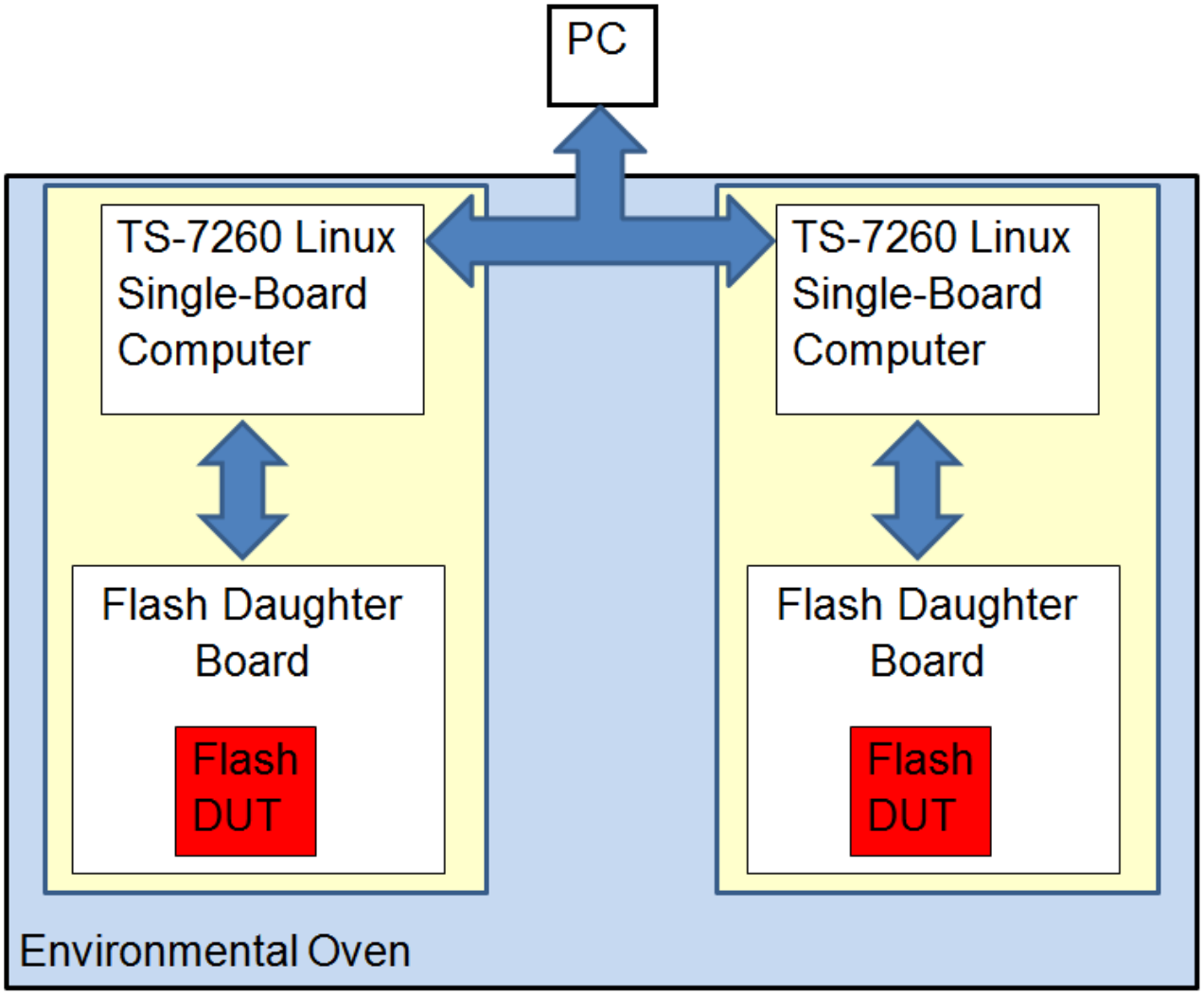
Introduction

- Research group
  - Collaboration between University of Limerick and Limerick Institute of Technology
- Project goals
  - Model NAND Flash reliability (endurance & retention)
  - Predict future performance & degradation
  - Classify & optimize devices

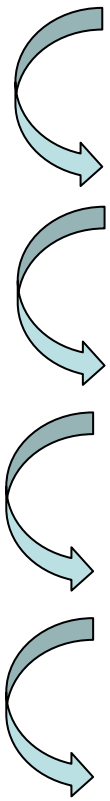
Flash Reliability

- Endurance: number of P-E cycles device can withstand
  - SLC: 100K; MLC: 5K-10K; TLC: 500
- Retention: length of time device will retain data
  - JESD47H.01 (Flash-level spec)
    - 1 year for 100% of max cycle count
    - 10 years for 10% of max cycle count
  - JESD218A (SSD-level spec)
    - 3 months at 40C

Test System


PC
TS-7260 Linux Single-Board Computer
TS-7260 Linux Single-Board Computer
Flash Daughter Board
Flash DUT
Flash Daughter Board
Flash DUT
Environmental Oven

Endurance/Retention Test Process



Endurance Stressing	• Weeklong at 85C • Random pattern copied from donor block
Donor Block Refresh	• Read Disturb
Pre-Retention Test	• ‘Difficult’ pattern – adjacent cells in opposite state
Retention Bake	• 13 hours at 85C equivalent to 3 months at 40C (Arrhenius equation)
Post-Retention Test	• Difficult pattern

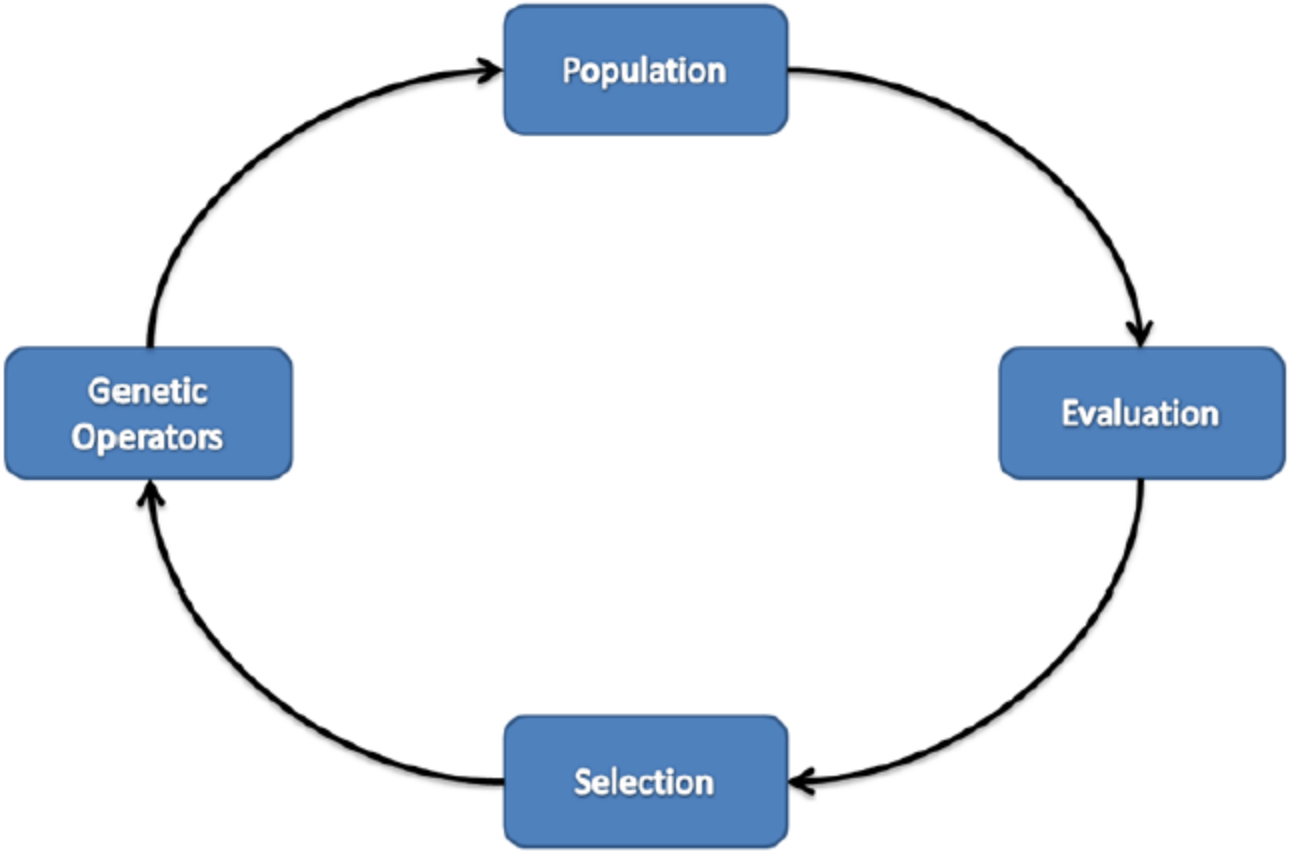
Machine Learning

- Machine Learning
  - Algorithms that improve through experience
- Evolutionary Algorithms
  - Use concepts from biological natural evolution
- Genetic Algorithms (GA) & Genetic Programming (GP)
  - GA - solutions represented as bit strings
  - GP - solutions represented as tree structures

GAs for Flash Optimization

- Control register settings
  - Store operating parameters – voltages & timings
  - Unavailable to user – obtained under NDA
  - Default parameters not optimal
- GAs use binary strings such as register settings
- Solutions tested on actual hardware by writing blocks to destruction

GA Process

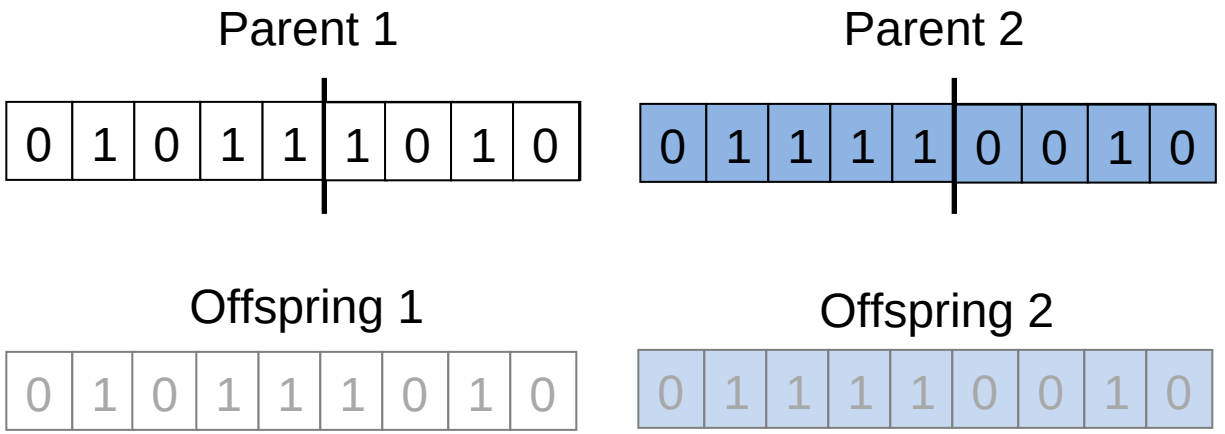


GA Process Steps

- Population
  - Individuals (register settings) chosen randomly
- Evaluation
  - Fitness – number of write/erase cycles completed
- Selection
  - Roulette wheel – fitness proportional selection
- Genetic Operators
  - Crossover and mutation

Genetic Operators

- Crossover
  - Random crossing point chosen in parent strings
  - Data after crossing point is swapped between both parents to form two new offspring
  - Aim is to produce children fitter than their parents



Genetic Operators

- Mutation
  - One bit changed from 1 to 0 or vice versa

0	1	0	1	1	1	0	1	0

  - Works against stagnation and loss of diversity
  - Used sparingly

Genetic Operators

- Mutation
  - One bit changed from 1 to 0 or vice versa

0	1	0	1	0	1	0	1	0

  - Works against stagnation and loss of diversity
  - Used sparingly

- GA properties include probability that crossover & mutation will occur

Genetic Programming

- Each individual is computer program
- Individuals represented as tree structures
  - Leaves (terminals) are variables and constants
  - Internal nodes (functions) are mathematical operations
- Genetic operators (crossover & mutation) are applied to evolve solutions to problem
  - Example: learning a prediction function

GP for Flash Modelling

- Use GP to model device degradation
- Use models to predict future performance and degradation
  - Can measurements made early in life be used to predict when end of life will occur?
- Use predictions to classify devices
  - Can these predictions be used to implement a production-level binning solution?

GP Trial Implementations – Endurance Classifier [1]

- Blocks cycled to destruction & program/erase times measured as function of cycle number
- Timing data used as GP input and number of cycles completed used as GP output
- Achieved up to 95% correct classification when presented with unseen data

GP Trial Implementations – Retention Classifier [2]

- Devices put through full endurance/retention test process
- No. of pre-retention errors used as GP input, no. of post-retention errors used as GP output
- Achieved up to 89% correct classification when presented with unseen data

Future Work

- Full endurance/retention predictor based on data that can be quickly acquired
  - Develop fast-cycling algorithm that is equivalent to distributed-cycling algorithm [3]
    - Requires measurement of V_T as function of retention time
  - Use predictor to propose industrial-grade binning solution
- NAND Flash optimization using GAs

Summary

- NAND Flash Testing
  - Testing requirements, test system, test process
- Machine Learning Introduction
  - Genetic Algorithms (GAs), Genetic Programming (GP)
- GP for NAND Flash modeling & classification
- GAs for NAND Flash optimization

Acknowledgements

- University of Limerick group
  - Dr. Conor Ryan, Dr. Tom Arbuckle, Damien Hogan
- Limerick Institute of Technology group
  - Joe Sullivan, Sorcha Bennett

Questions?

Bibliography

1. "Evolving a Storage Block Endurance Classifier for Flash Memory", Damien Hogan, Tom Arbuckle, Conor Ryan. To appear (August 2012) in Proceedings of 11th IEEE International Conference on Cybernetic Intelligent Systems (CIS 2012).

2. "Evolving a Retention Period Classifier for use with Flash Memory", Damien Hogan, Tom Arbuckle, Conor Ryan, Joe Sullivan. To appear (October 2012) in Proceedings of 4th International Conference on Evolutionary Computation Theory and Applications (ECTA 2012).

3. "Investigation of the Threshold Voltage Instability after Distributed Cycling in Nanoscale NAND Flash Memory Arrays", Christian Monzio Compagnoni et al. Published in Proc. IRPS (IEEE 2010)