
SIGLEAD

# Intelligent Read Threshold Tracking to Mitigate Read Retry Requests

Oliver Hambrey

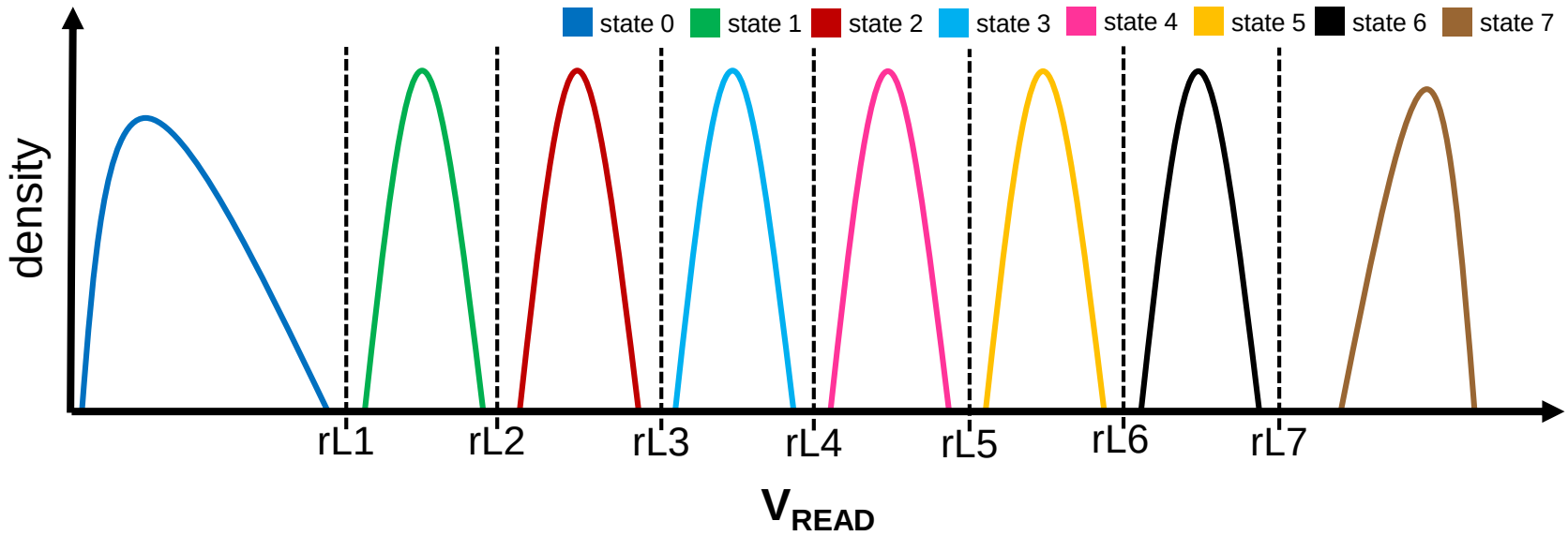
Siglead Europe Limited

# Bit Errors and Error Correction

- NAND flash memories often read in error
  - over/under programmed cells
  - stress induced leakage current (SILC)
- methods for mitigating bit errors include:
  - error correction codes (ECC)
  - digital signal processing (DSP)
  - read threshold adjustment

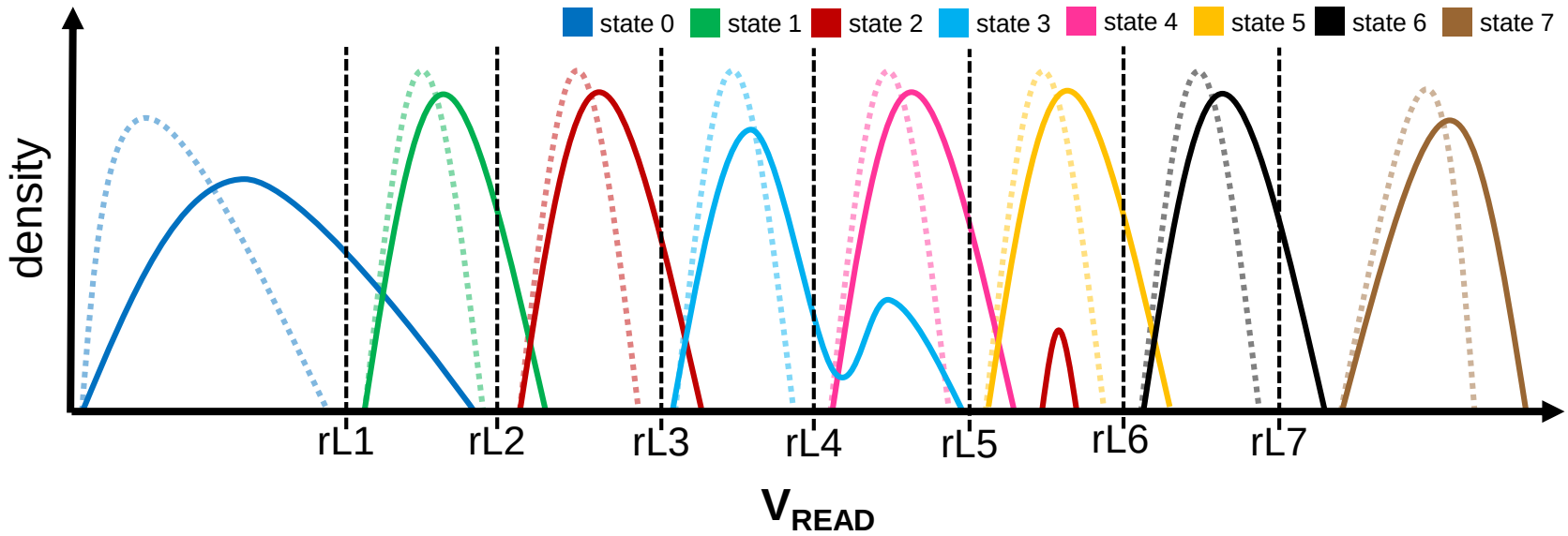
# Evolving Voltage Distributions (1)

- distributions change during NAND lifespan



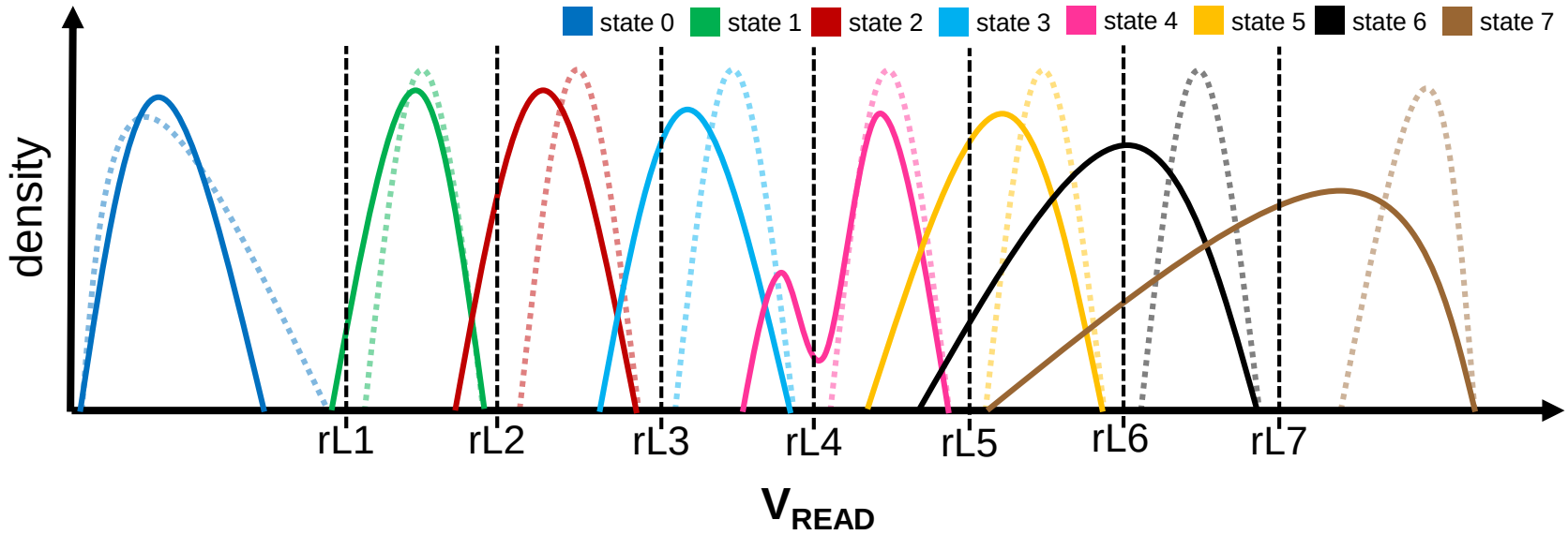
# Evolving Voltage Distributions (2)

- distributions change during NAND lifespan
  - upwards shift (disturb, program error)



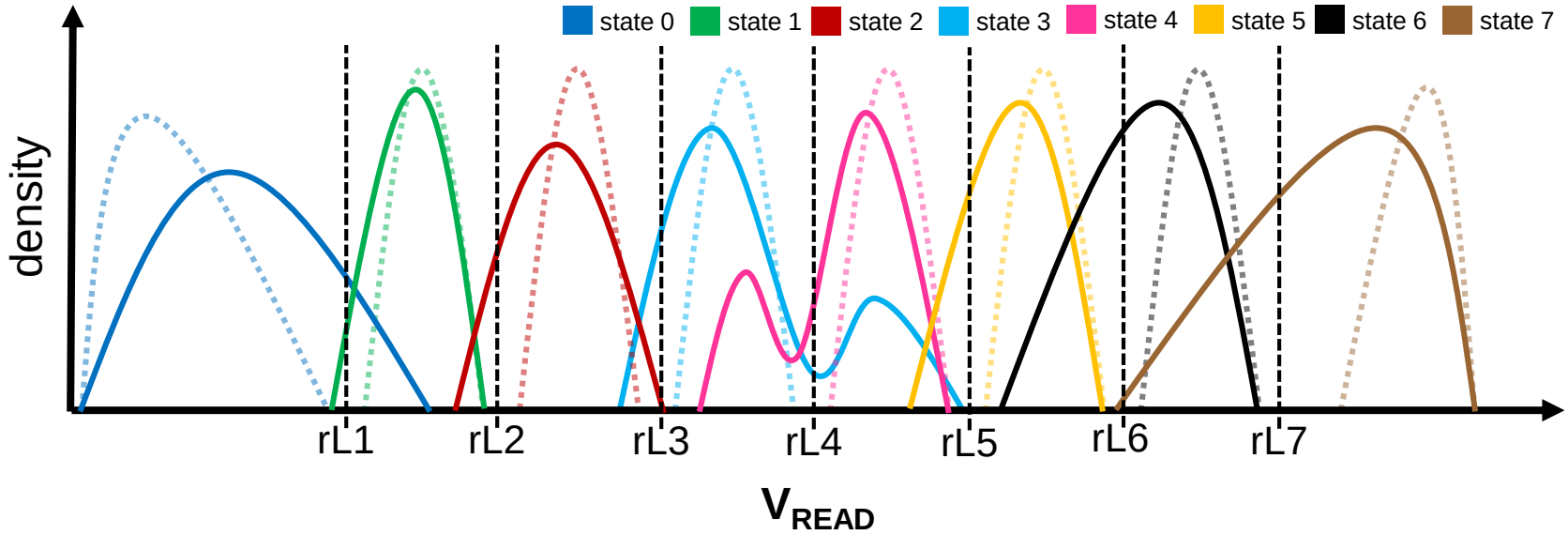
# Evolving Voltage Distributions (3)

- distributions change during NAND lifespan
  - downwards shift (SILC, program error)



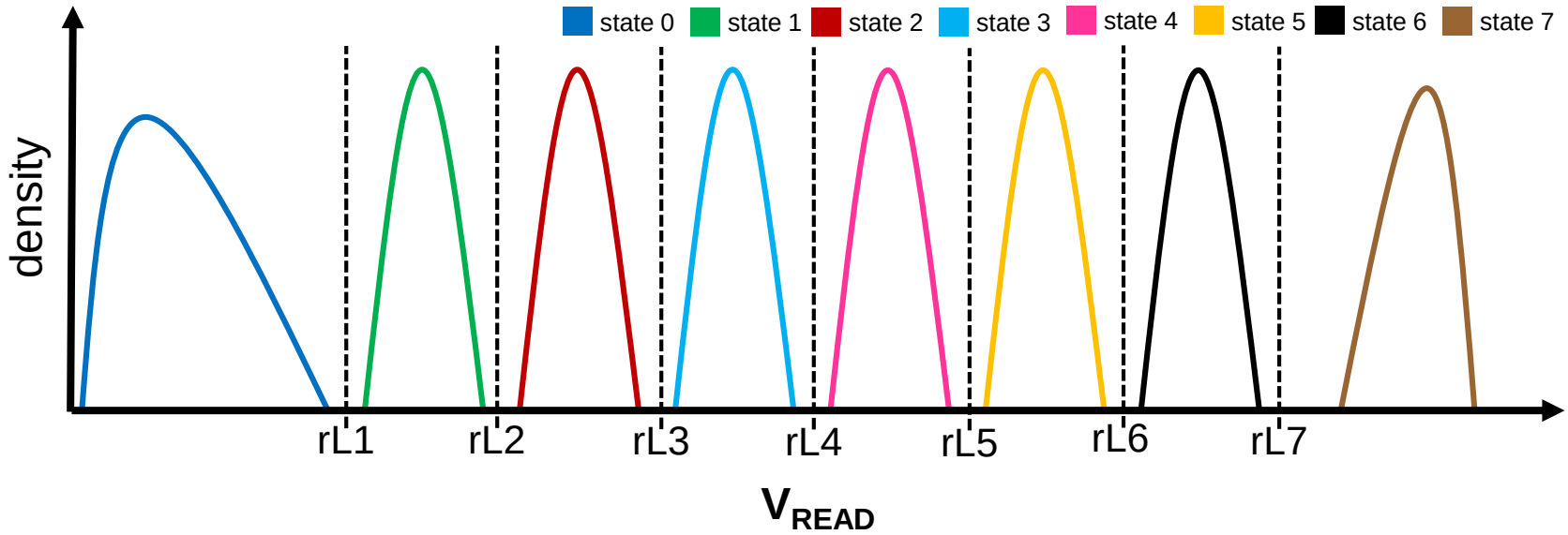
# Evolving Voltage Distributions (4)

- distributions change during NAND lifespan
  - typically combination of all (disturb, SILC, program error)



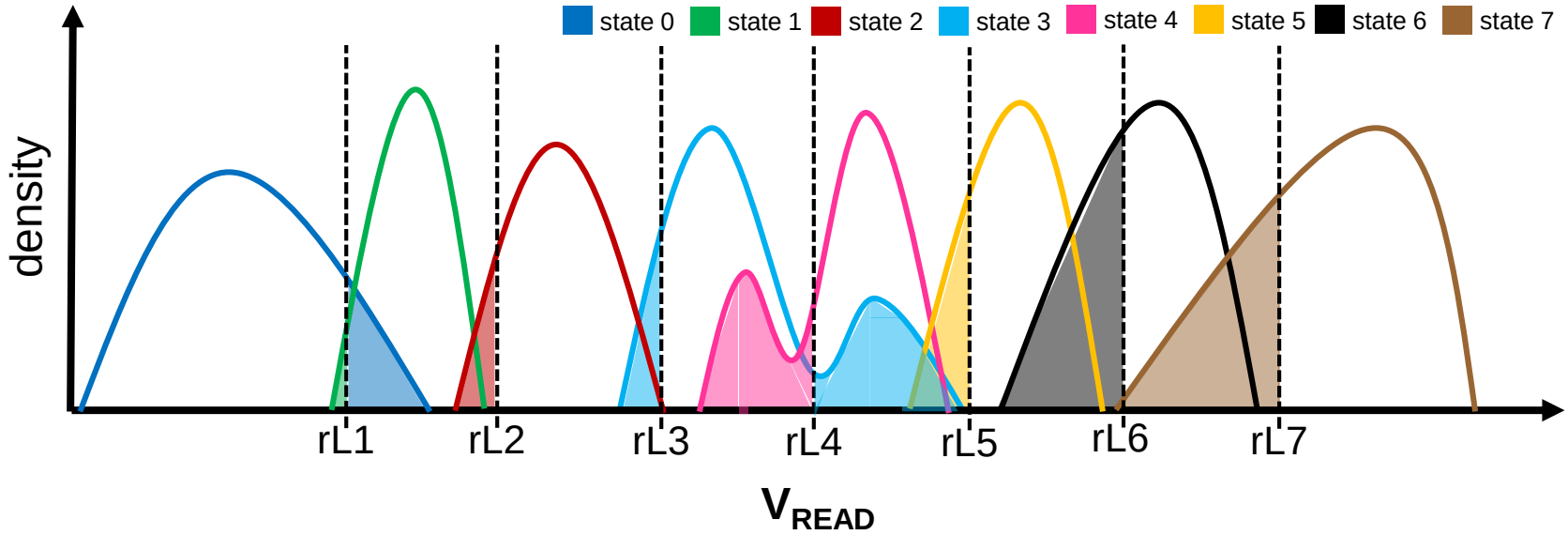
# Read Threshold Placement (1)

- beginning of life read thresholds well placed
  - states distinguishable, bit error rate is low



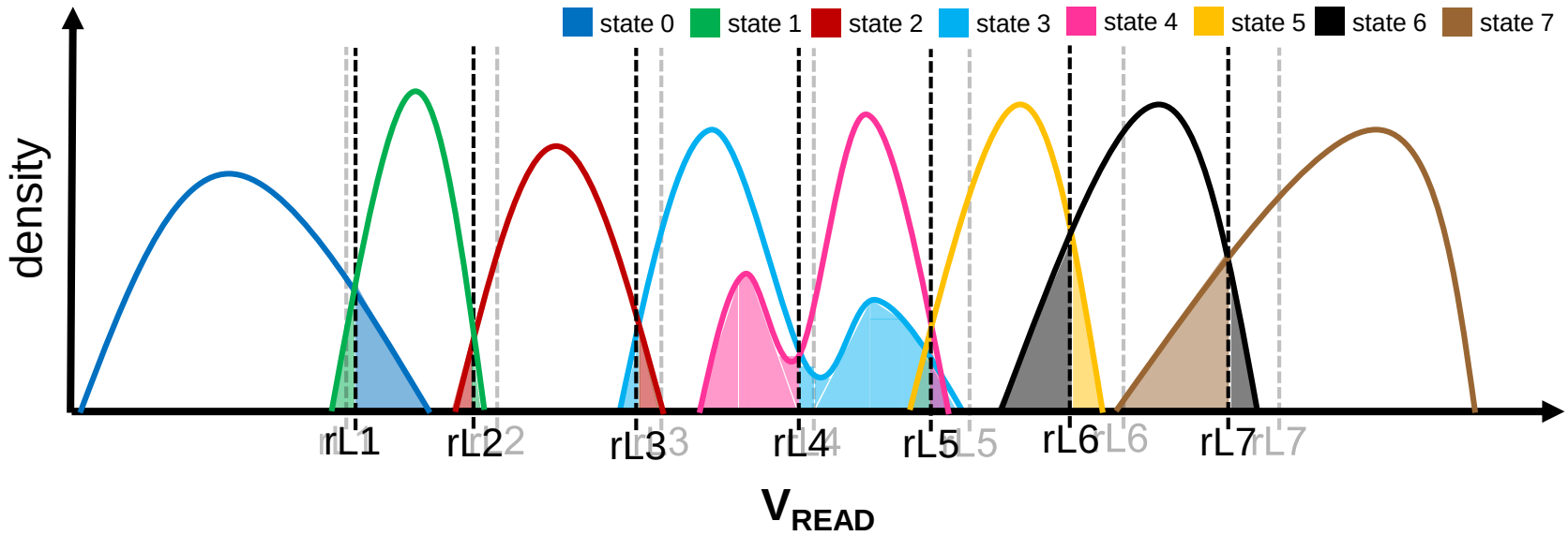
# Read Threshold Placement (2)

- over lifespan voltage distribution shifts
  - read thresholds may not optimally distinguish states



# Read Threshold Placement (3)

- change read thresholds to counteract shift
  - states can be optimally distinguished, reducing bit error

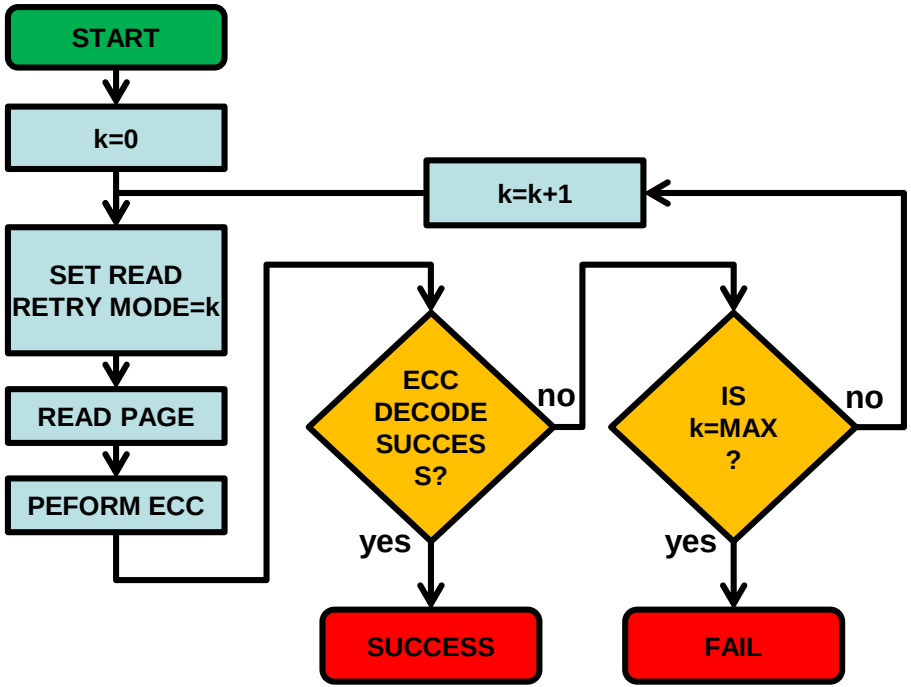


# Easy, Right?

- in practice, how can/should we change read thresholds?
  - track voltage distributions (optimal, slow, complex)
    - **READ DISTURB**
  - read retry (sub-optimal, slow, simple)
- adaptive on-the-fly read threshold tracking
  - near-optimal
  - fast
  - simple

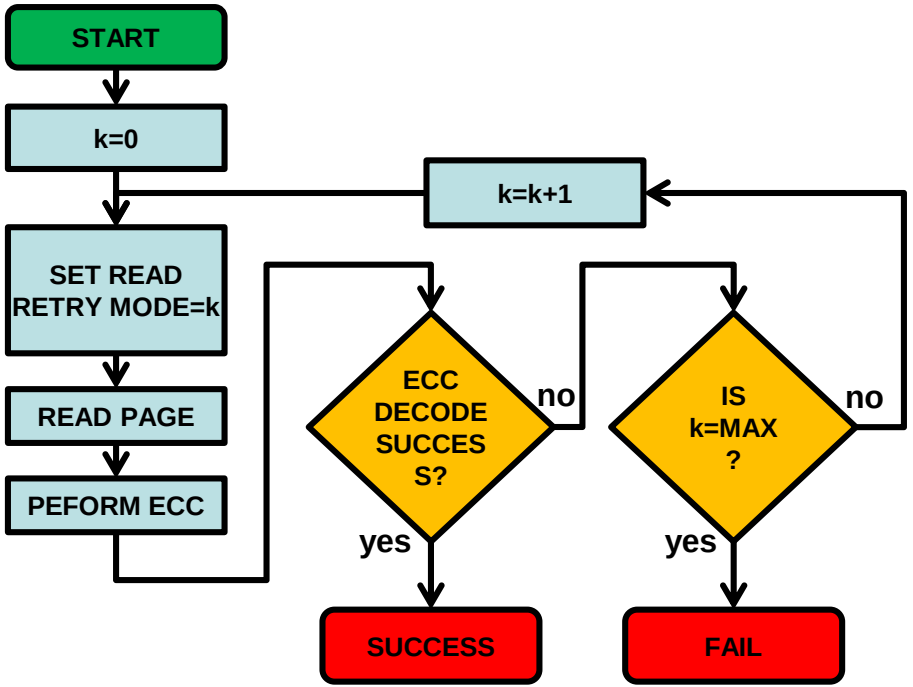
# Read Retry (1)

- NAND vendor provides sequence of pre-set read thresholds
  - threshold values may/may not be disclosed
- in event that ECC decoding fails, re-read page with next preset
- continue iteratively until either
  - ECC decoding successful
  - all presets are exhausted

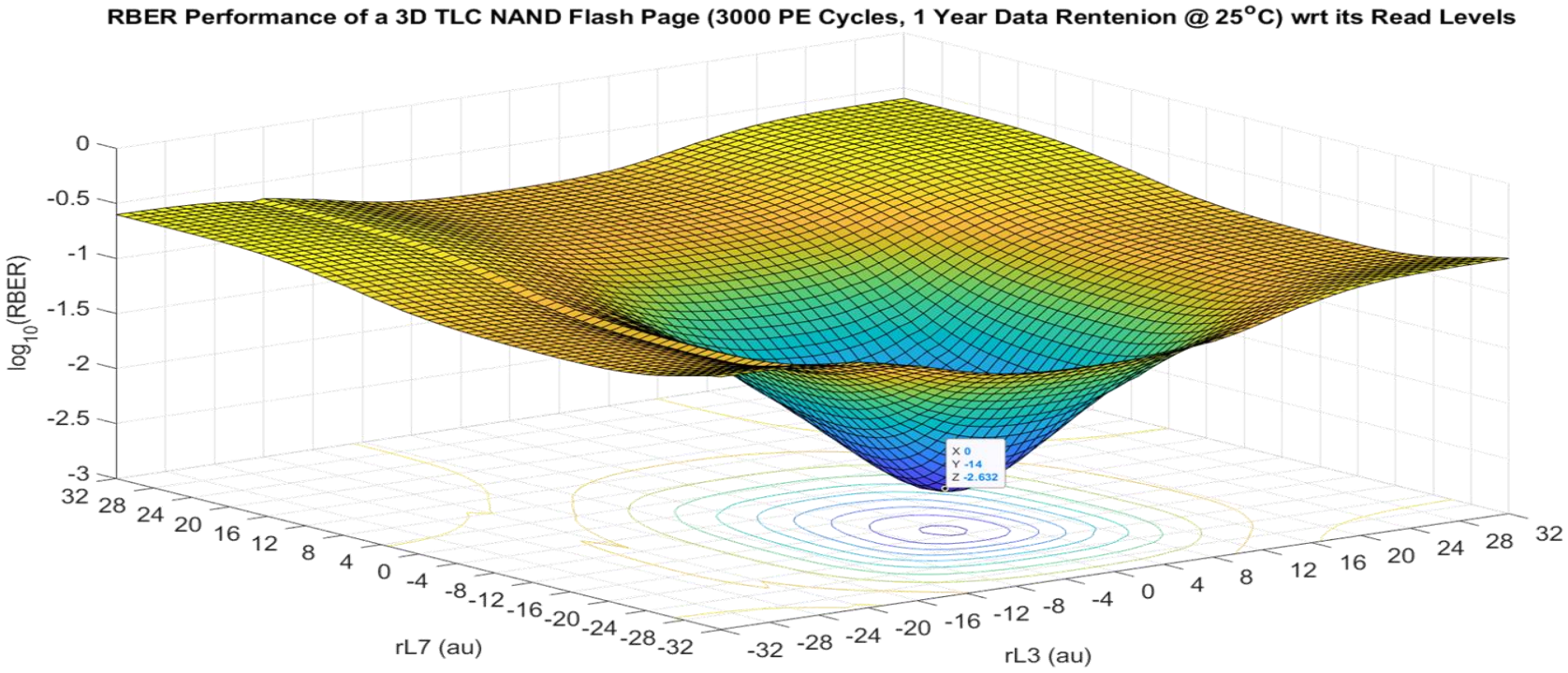


# Read Retry (2)

- what's good?
  - read retry is simple
- what's bad?
  - no guarantee any of the presets will be suitable
  - re-reading the page several times is slow
- can we do better?



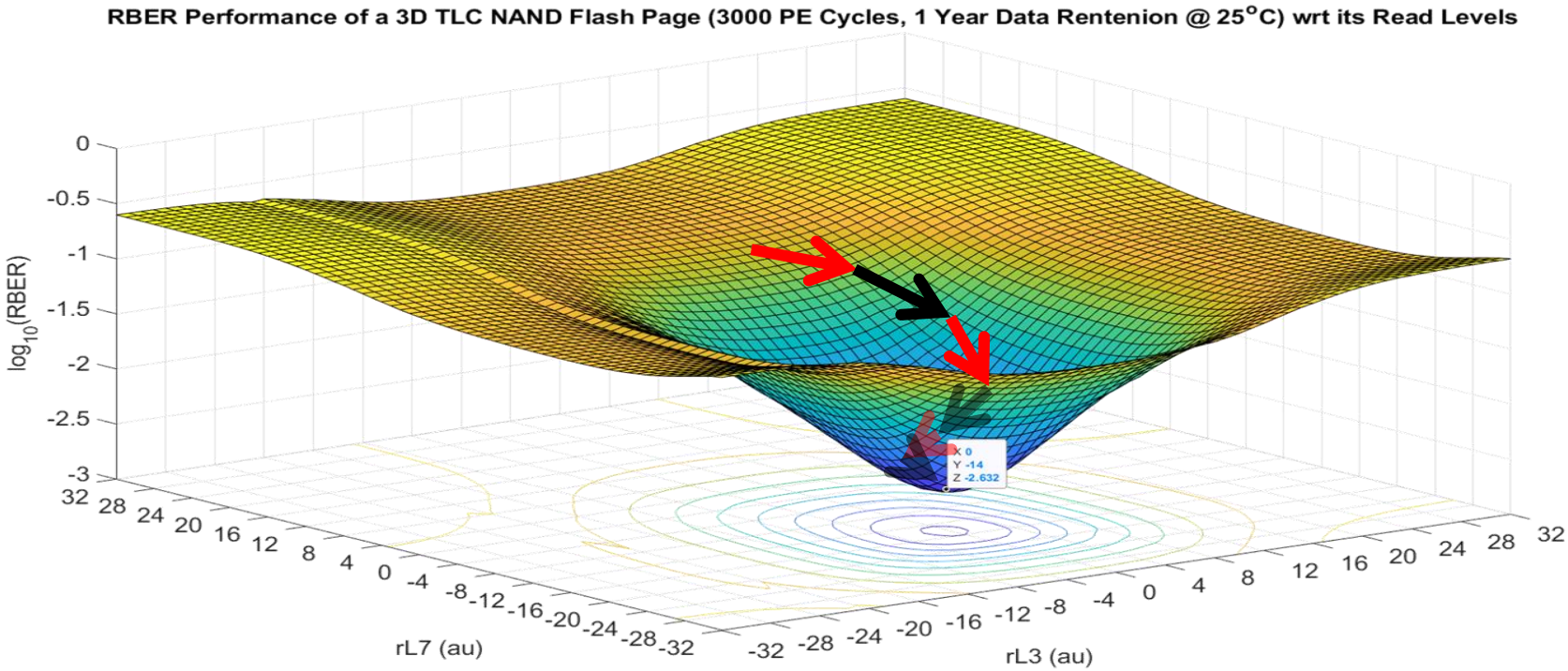
# Read Threshold Space (1)



# Read Threshold Space (2)

- need intelligent way to find location of trough
  - near-optimal:
    - quickly gets close to trough location
    - if trough moves, new location is quickly rediscovered
  - fast:
    - only 1 read from NAND
  - simple:
    - needs minimal amount of side information
    - uses only basic arithmetic

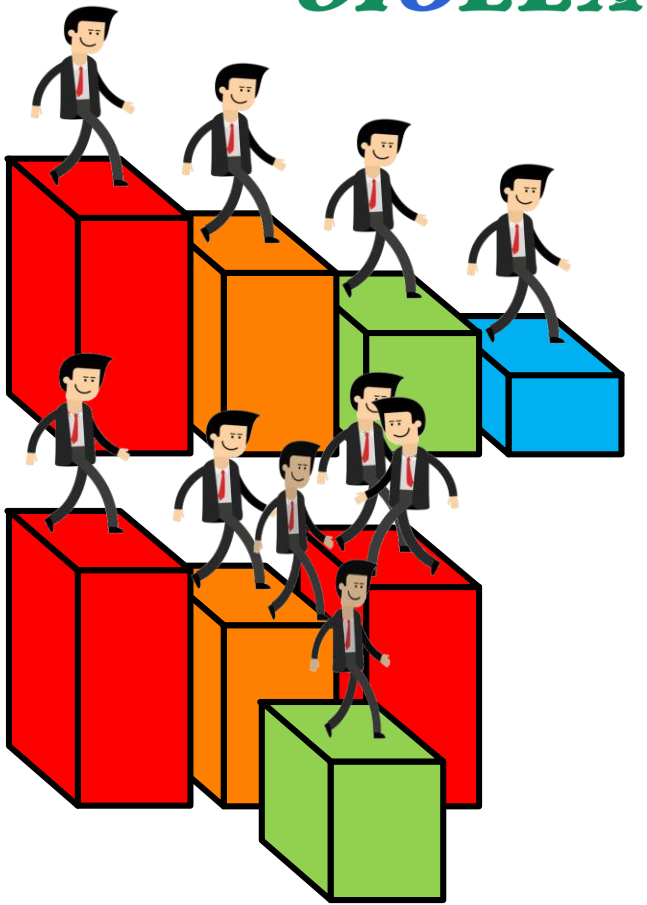
# Read Threshold Space (3)



# Simple Movement

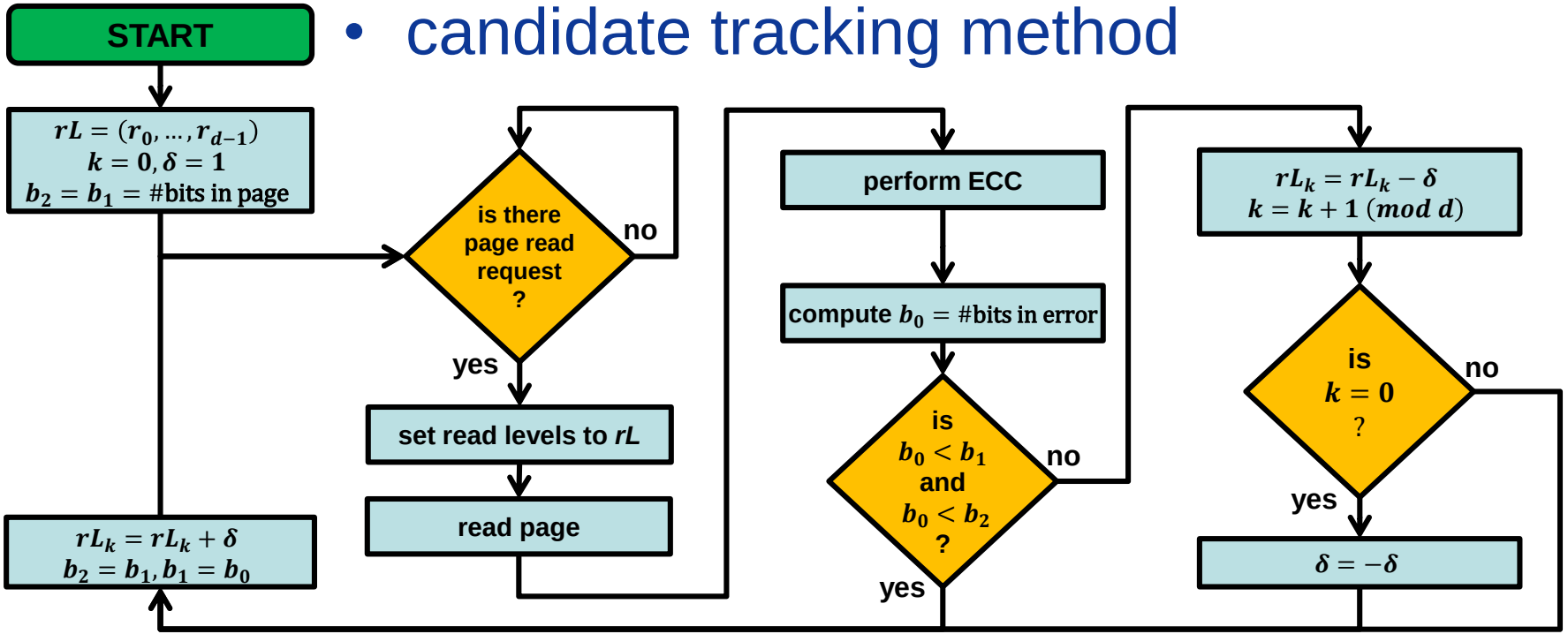


- bit errors getting smaller
  - KEEP MOVING FORWARDS
- bit errors getting larger
  - STEP BACKWARDS
  - CHANGE DIRECTION
  - MOVE FORWRDS



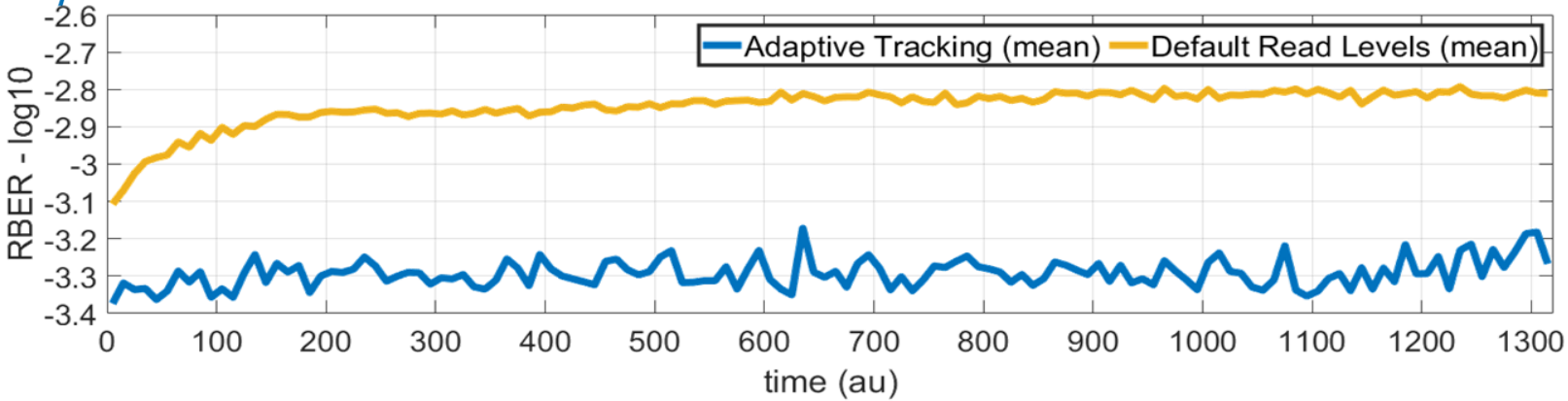
# Adaptive Read Threshold Tracking

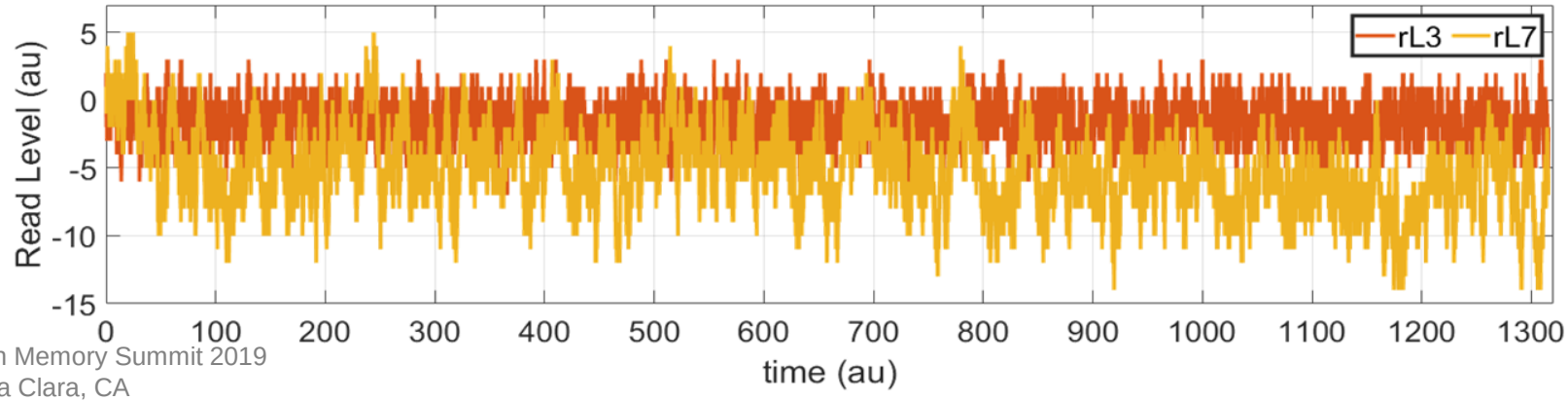
- candidate tracking method



SIGLEAD

# Does This Work?



# Summary

- simple, intuitive on-the-fly methods for adaptively tracking NAND flash read levels…
  - exist
  - reduce RBER
  - negate need for re-reading
  - building voltage distribution not required
- **BUT:** this is a basic introduction to this idea!
  - there are aspects I have not covered
  - there are challenges to overcome I have not mentioned

SIGLEAD

# Buy a NAND Analyzer

## you can investigate too!

**SigNASII**
lightweight, versatile


**SigNAS3**
high quantity testing
(still versatile)

www.siglead.com