Characterizing NAND Devices at High Speed

Why and How ?

Tamás Kerekes - NplusT

Why Characterizing the NAND ?

“Why characterizing” is not a question

- NANDs have inherent failures
- Workaround algorithms are needed
- Efficient workarounds need correct tuning
- Tuning requires a deep knowledge of the specific NAND failure modes
- Datasheet does not contain this information

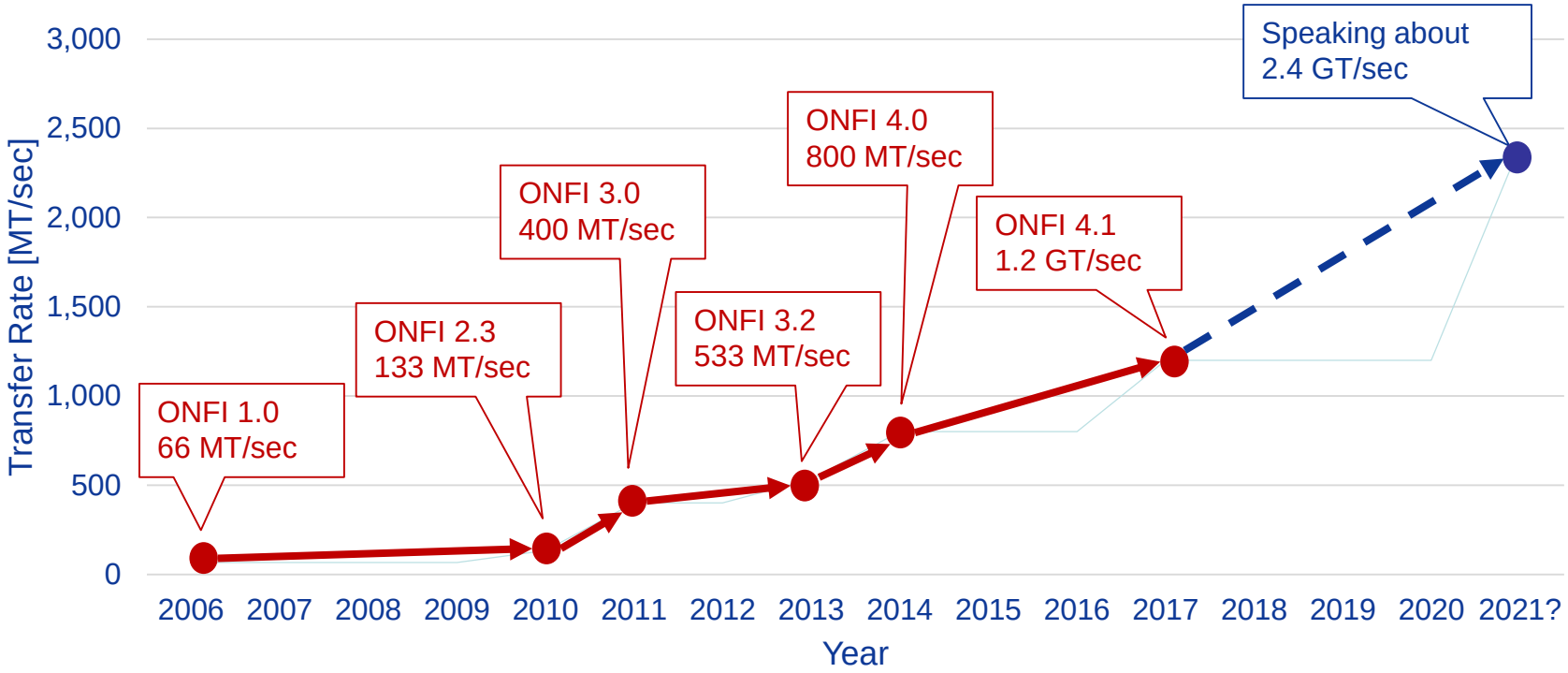
to build a good SSD,

you cannot live without NAND characterization

(if you are not convinced, let’s talk about)

second question … why at high speed ?

NAND interface speed trend



Basic concept of the characterization

characterization in an application-like environment

+

speed in the applications is rocketing

=

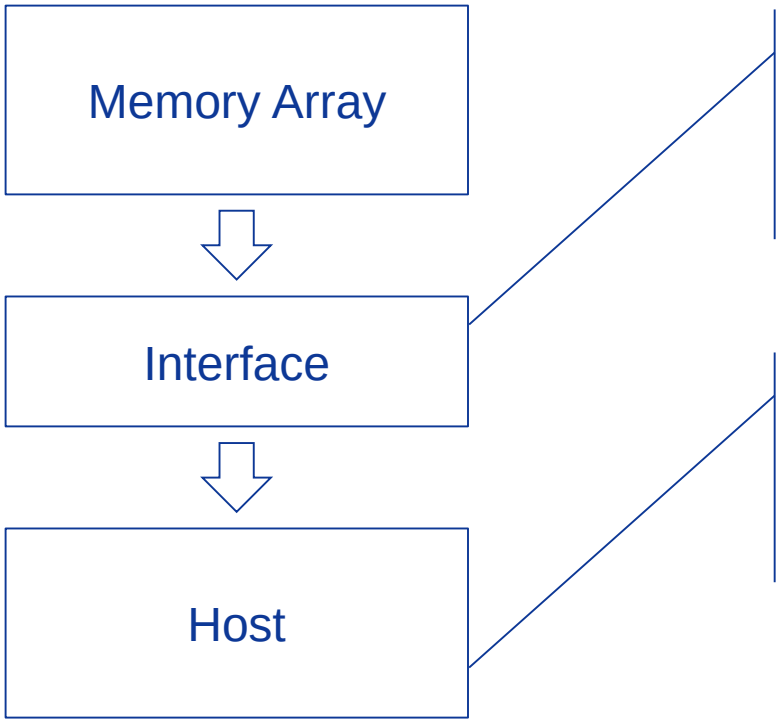
high speed characterization is desirable

(useful, mandatory)

Benefits of the high speed

- Quality of the result
  - Read/write speed might have an impact on the flash array operation
- Execution efficiency
  - Lower execution time → lower overall cost
  - … and engineers don’t like to wait

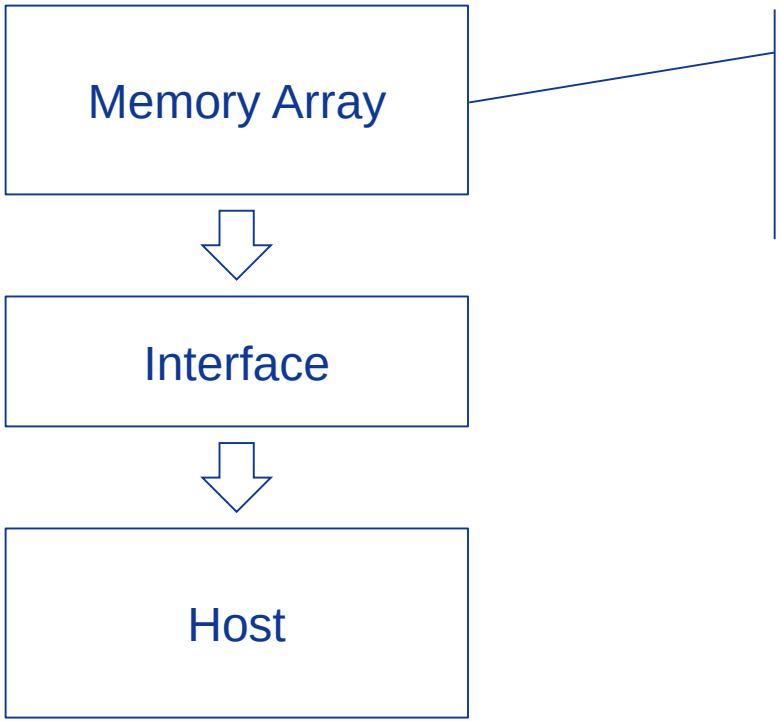
Impact of high-speed readout



Due to the high-speed readout, the power consumption increases which changes the thermal status of the entire package.

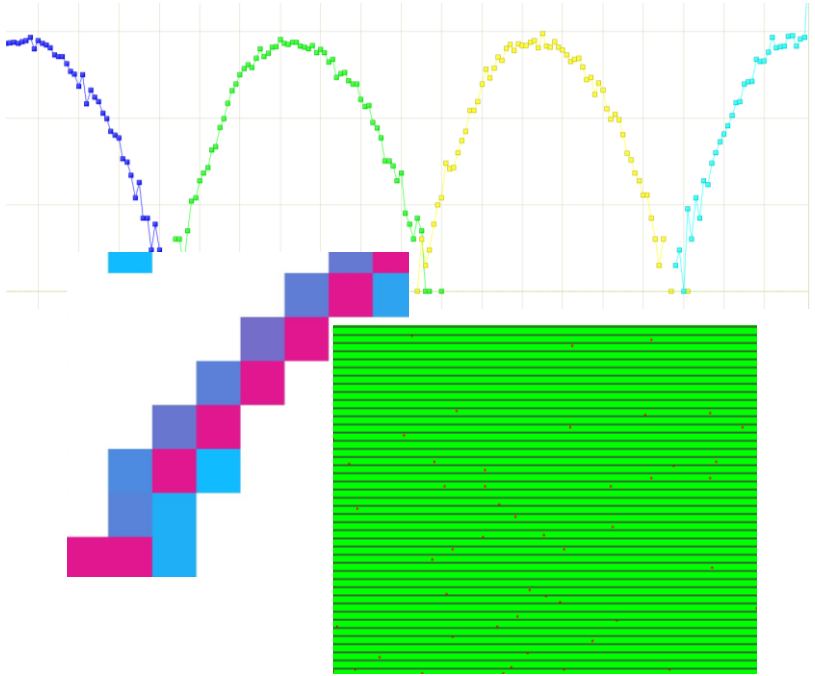
The timing of the operations change due to the shorter transfer times. Relaxation conditions might impact array failure rates.

Impact of the low test overhead



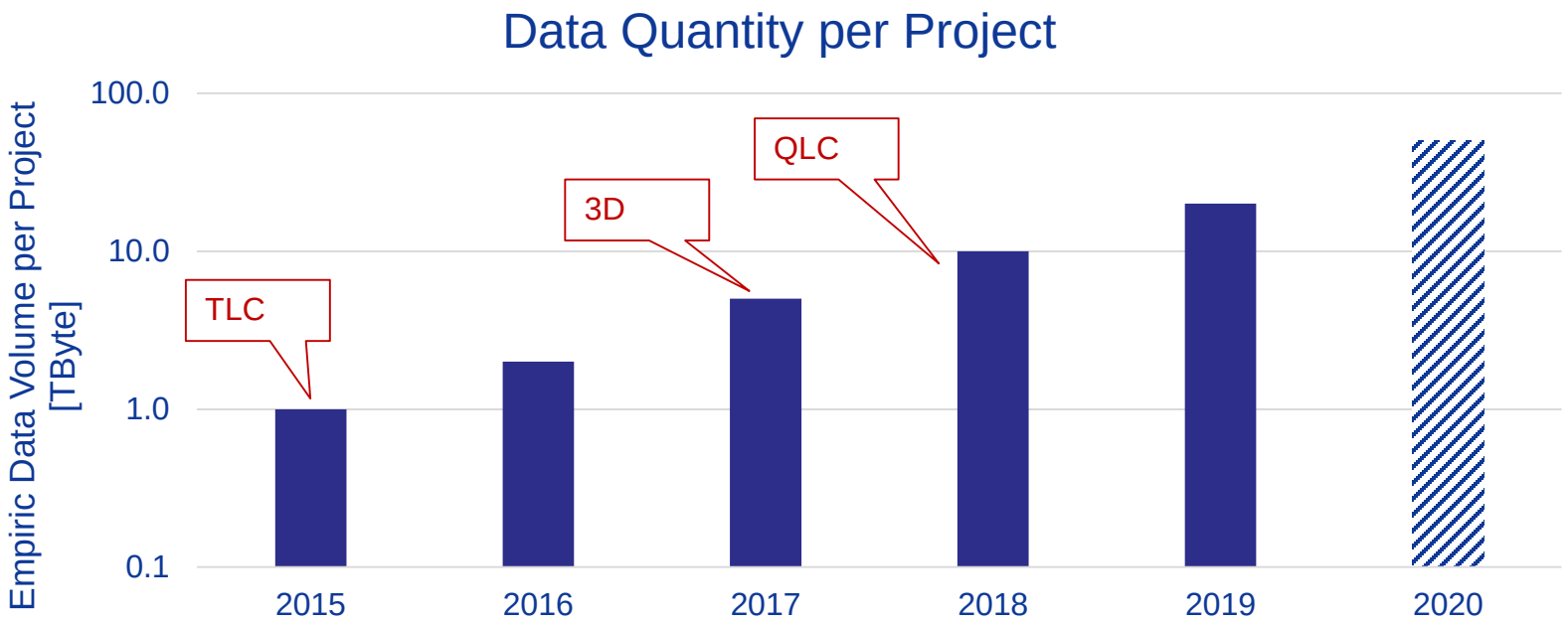
High test overhead might create different timing conditions from a real-life application, causing misleading results.

Execution efficiency - bitmapping

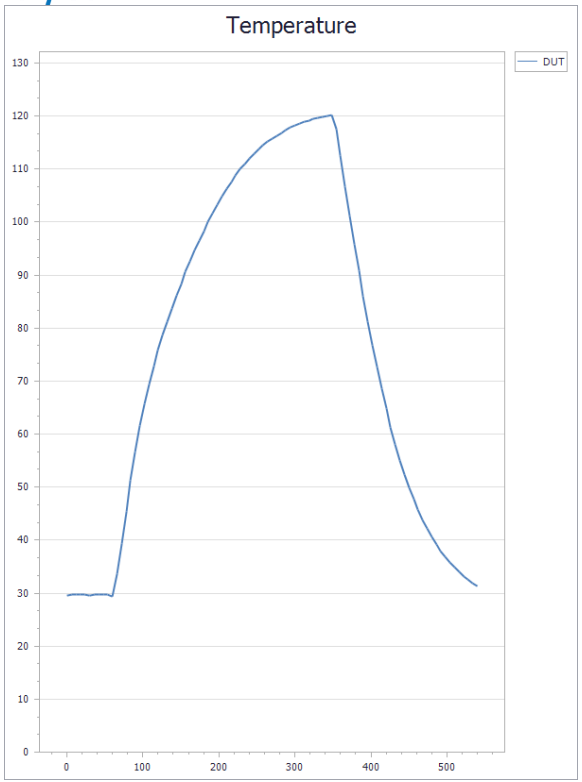


- Complex experiments like Vt analysis, value shift, bitmapping require the bit-per-bit analysis of the readout contents
- Several dozens of GBytes of data might be generated by a single experiment on a single block
- These numbers are even worse at ECC/LDPC analysis
- This quantity of data sets challenges for the data collection system

Data volume per project



Execution efficiency – temperature

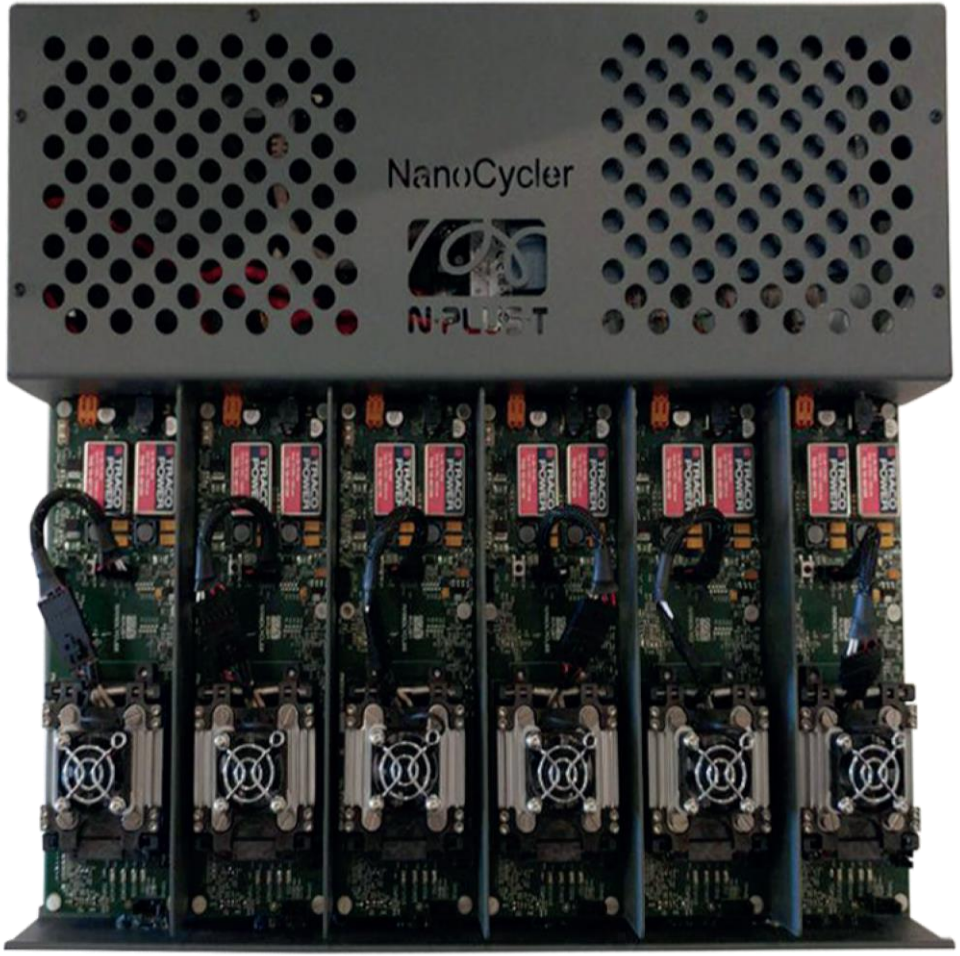


- Temperature might change frequently during the experiment
  - e.g. cycling at high temperature and readout at room temperature
- Fast temperature setting and stabilization saves significant execution time

How a Characterization Tester should look like ?

Flash Memory Summit

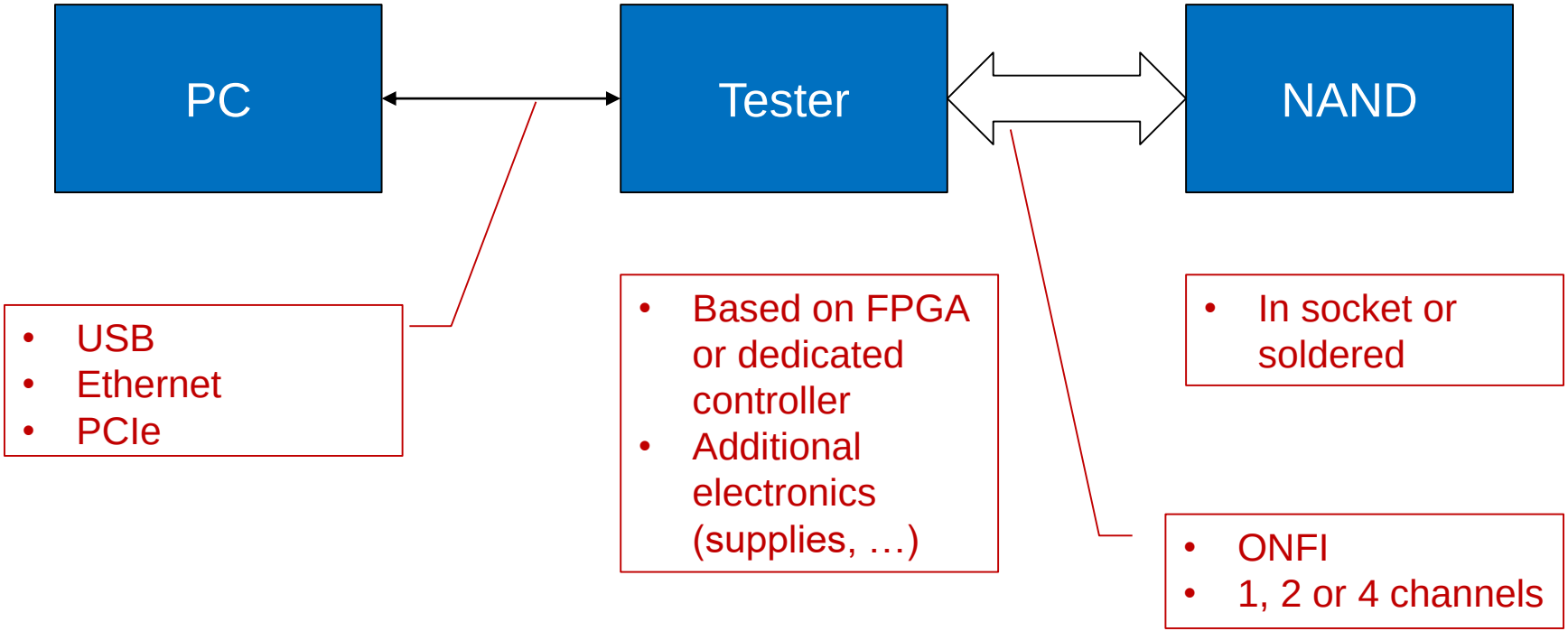
NanoCycler



High-impact characteristics

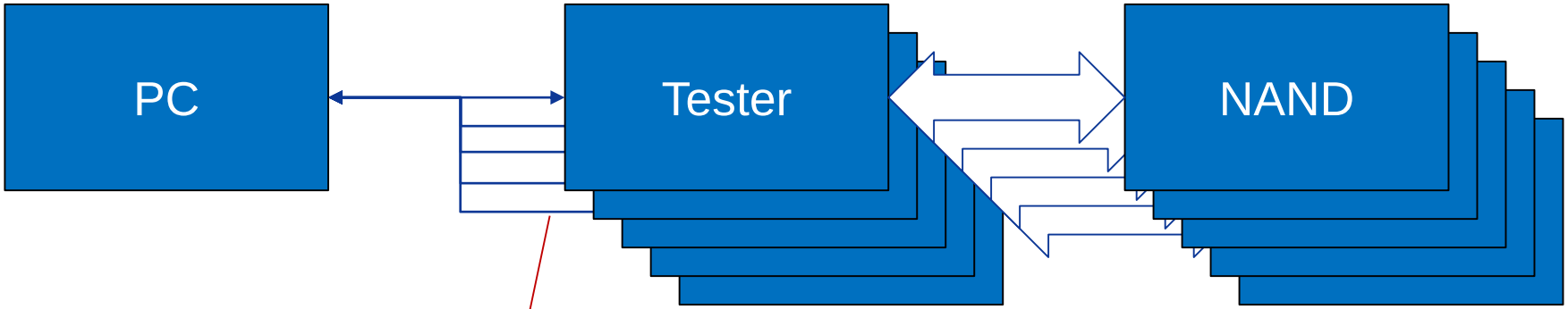
- High-speed Tester-NAND communication
- Low overhead of the single test functions
- Efficient data collection
- Fast and accurate temperature control

Basic/typical architecture



Scalability – another requirement

Need of executing multiple – and different ! – tests in the same time



- ~~USB~~
- **Ethernet**
- ~~PCIe~~

Tester – NAND communication

High-Speed Hardware:

- Design rules, terminations, references, …
- Layout
- Socket electrical behavior
- Hi-performance FPGA

Powerful Firmware:

- PHY (FPGA)
- DQS alignment algorithms

Reducing test overhead

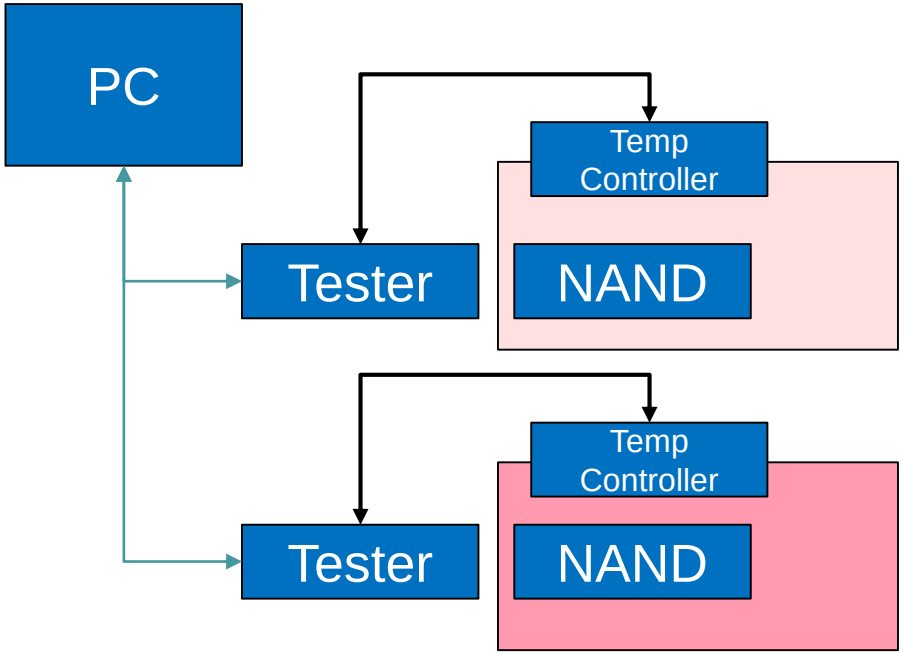
- High-speed link for user-defined pattern download
- Test flow executed locally (and efficiently)
- On-the-fly pattern generation
- On-the-fly pattern matching and RBER counting
- Local storage for user-defined patterns

Efficient data collection

- High-speed link for fast test result upload

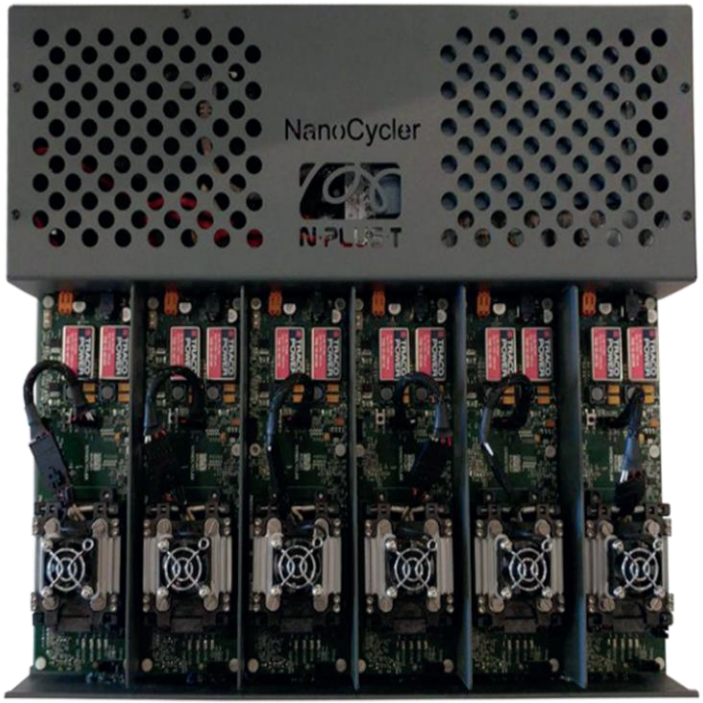
- Local storage for test results optimizing the global network load

Distributed temperature control



- Each NAND shall have its own micro-chamber
  - Fast
  - Accurate
  - Independent per socket
  - Controlled by the tester
- Chamber-based solutions do not match characterization requirements
  - Slow
  - Not accurate
  - Common temperature per socket

NanoCycler



- One tester unit per socket
  - µP with LINUX and SD
  - C++ or Python engine + test library
  - FPGA, proprietary test IP
  - Fast local bus between µP and FPGA
  - Mini-chamber with temperature control
- Scalable up to 48 tester units
  - 1Gb Ethernet to PC
- Experiment setup and data analysis on the PC

what if the 1Gb Ethernet becomes the bottleneck of bitmap-intensive applications ?

we are working on the answer

see you at FMS 2020