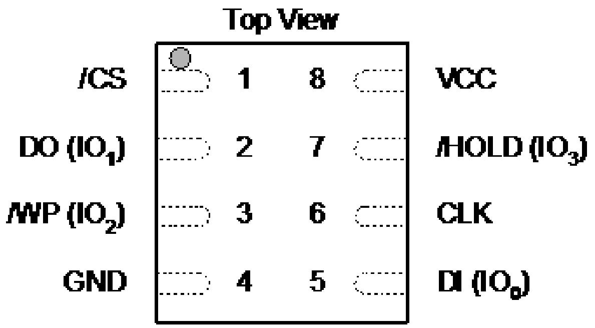
New Architecture for Code-Shadowing Applications

Dual-buffer SPI-NAND Arch providing SPI-NOR compatibility

Anil Gupta

Technical Executive, Winbond Electronics



Key features of New Architecture (Dual-buffer SPI-NAND):

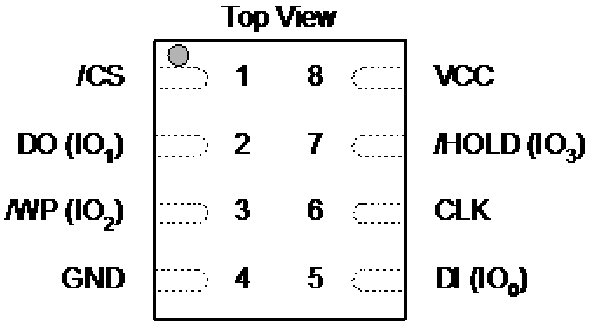
- Package and pin-out compatibility with SPI-NOR (both with 8-pin foot-print)
- Read command and clocking compatibility with SPI-NOR Flash
  - High speed continuous/sequential read across Page and Block boundaries

➢ New Arch compatible with SPI-NOR Flash in “code shadowing” applications

  - SPI-NOR not likely to be displaced in XIP applications for fast random access

➢ Fully integrated on-chip solution for high speed continuous/sequential read :

  - Dual-buffer Arch provides gapless interleaved read
  - On-chip ECC (error-correcting code)
  - On-chip LUT (look-up table) for BBM (bad block management)
  - Eliminates need for ext. controller chip for providing ECC and LUT (for BBM)



Major benefits of new SPI-NAND Arch:

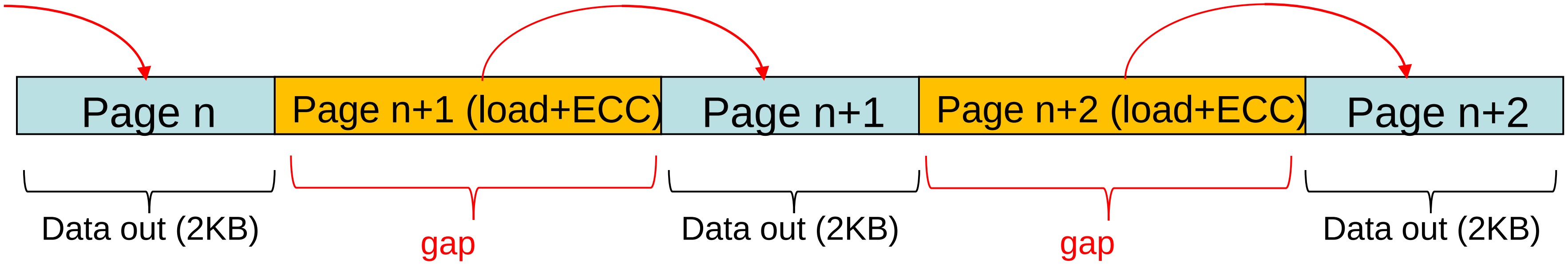
- Highly cost effective solution for “code shadowing” applications
  - NAND cell size ~4.5F2 much smaller compared to NOR cell size ~12F2
  - NOR floating gate technology reaching scaling limit ~45nm, whereas NAND has scaling path down to 1Xnm
    1. NOR has one CO (contact) per cell, whereas NAND has one CO per string
       - String may be 32, 64, or 128 cells
    2. Programming by CHE (channel hot electron) in NOR, but tunneling in NAND
       - Cell channel length has scaling limit due to CHE
- Large SPI-NAND densities in 8-pin package, e.g. 1Gb on 4X nm and 2Gb on 3X nm in 6x8 WSON package
  - In the SPI-NOR Flash, 128Mb/256Mb are largest densities to fit 6x8 WSON
- SPI-NAND is attractive alternative in code-shadowing for higher densities
  - SPI-NOR densities < 256Mb are already able to offer small footprint (e.g. 6X8 WSON)

Note: NOR offers fast random access because of higher cell current, and NOR doesn’t require BBM

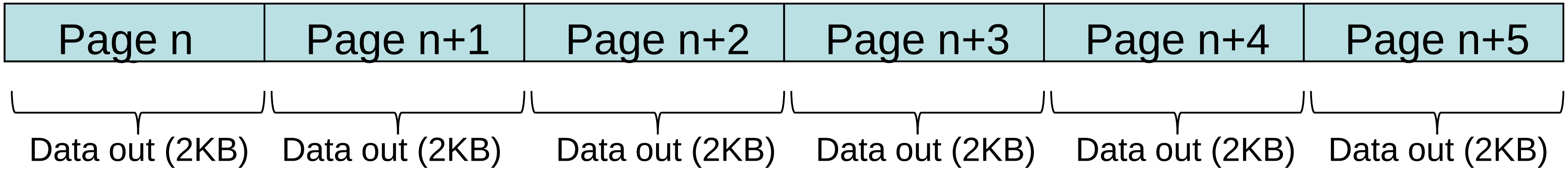
Code shadowing: Legacy Arch vs New Arch

(comparing both Single Array/Plane Arch)

1. **Legacy Architecture: read with gaps**



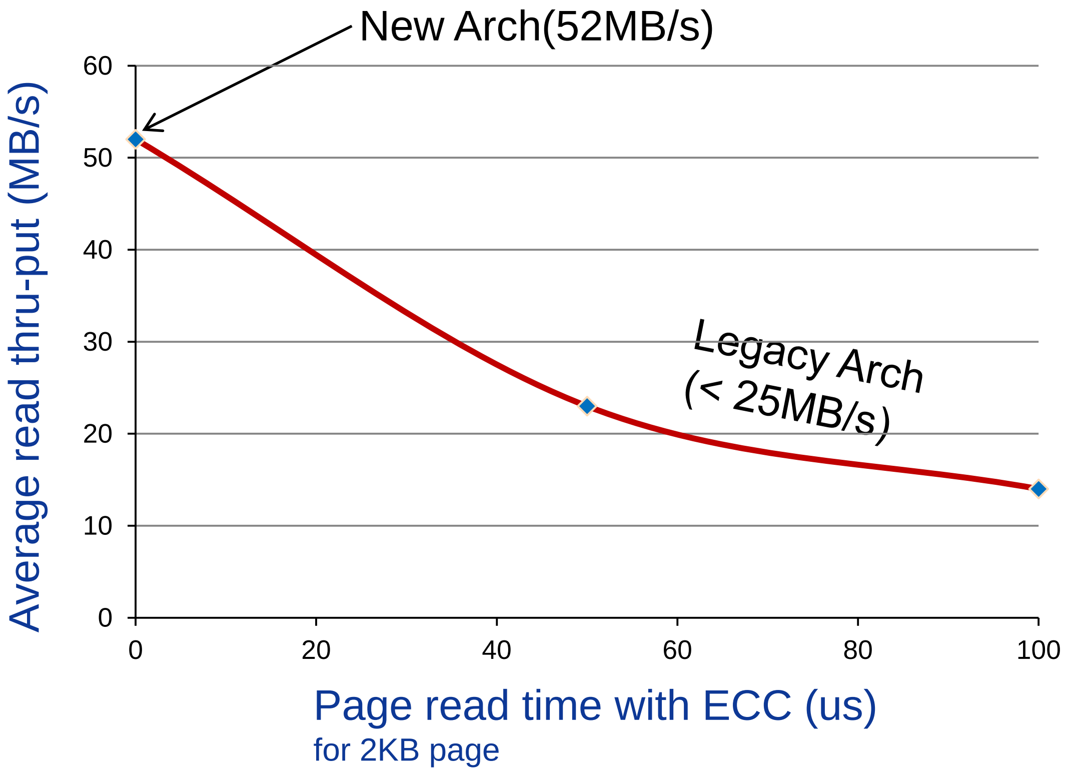
2. **New Dual-Buffer Arch: continuous/sequential read without gaps**



1. Legacy Arch suffers large gaps (e.g. 50us+) between reading out data from consecutive pages
   - Gaps due to Page read : load data from NAND Array to Page Buffer plus ECC computation
2. New Arch provides "gapless" continuous read across Page and Block boundaries
   - Ideal for Code Shadowing applications when large amount of data is sequentially read out

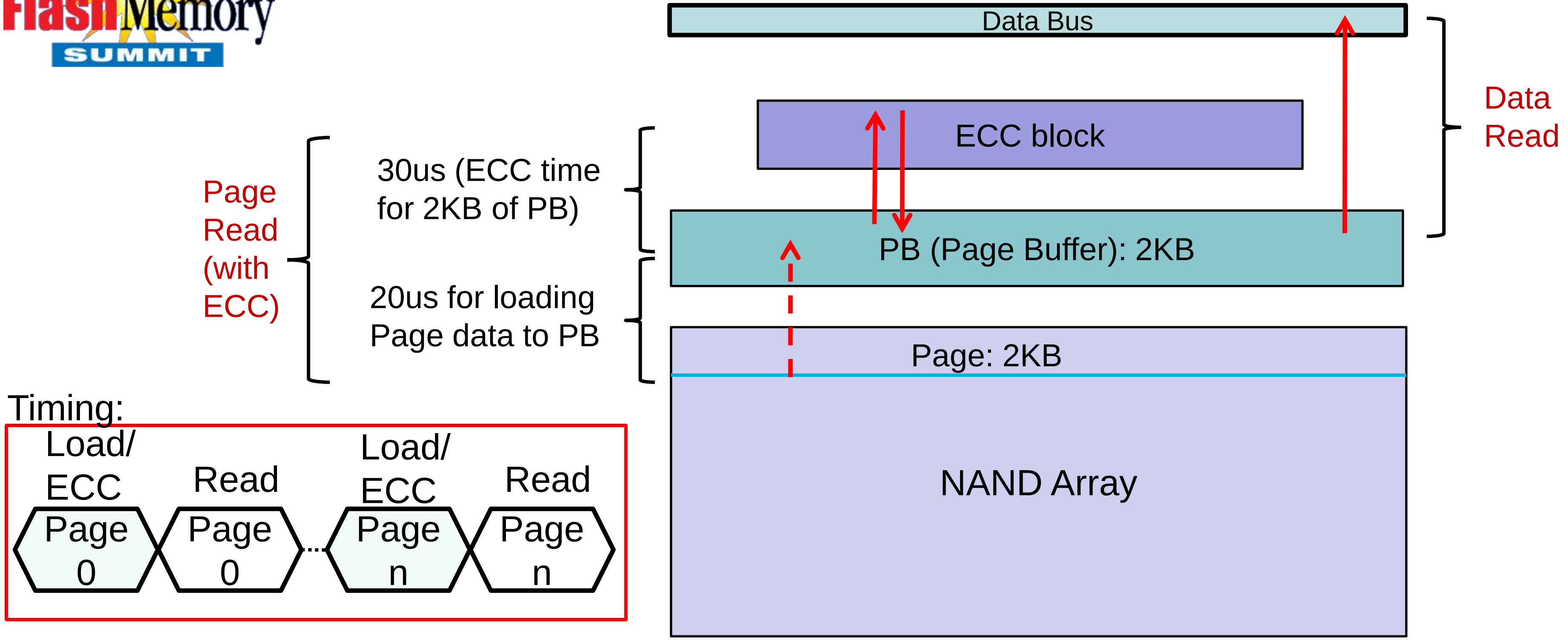
Average read thru-put (MB/s) vs. Page Read time with ECC

for Quad (4-bit I/O) operation and 104MHZ Clock Freq



- Legacy Arch: Average read thru-put (< 25MB/s) much slower than 52MB/s in New Arch
  - There may be 50us+ gaps (Page read time with ECC), before each 2KB data read out
- New Arch: Average read thru-put 52MB/s is exactly half of the 104MHZ Clock Freq (Quad I/O)
  - Page read time with ECC time is completely hidden from the read thru-put

Legacy Single Array/Plane NAND Arch



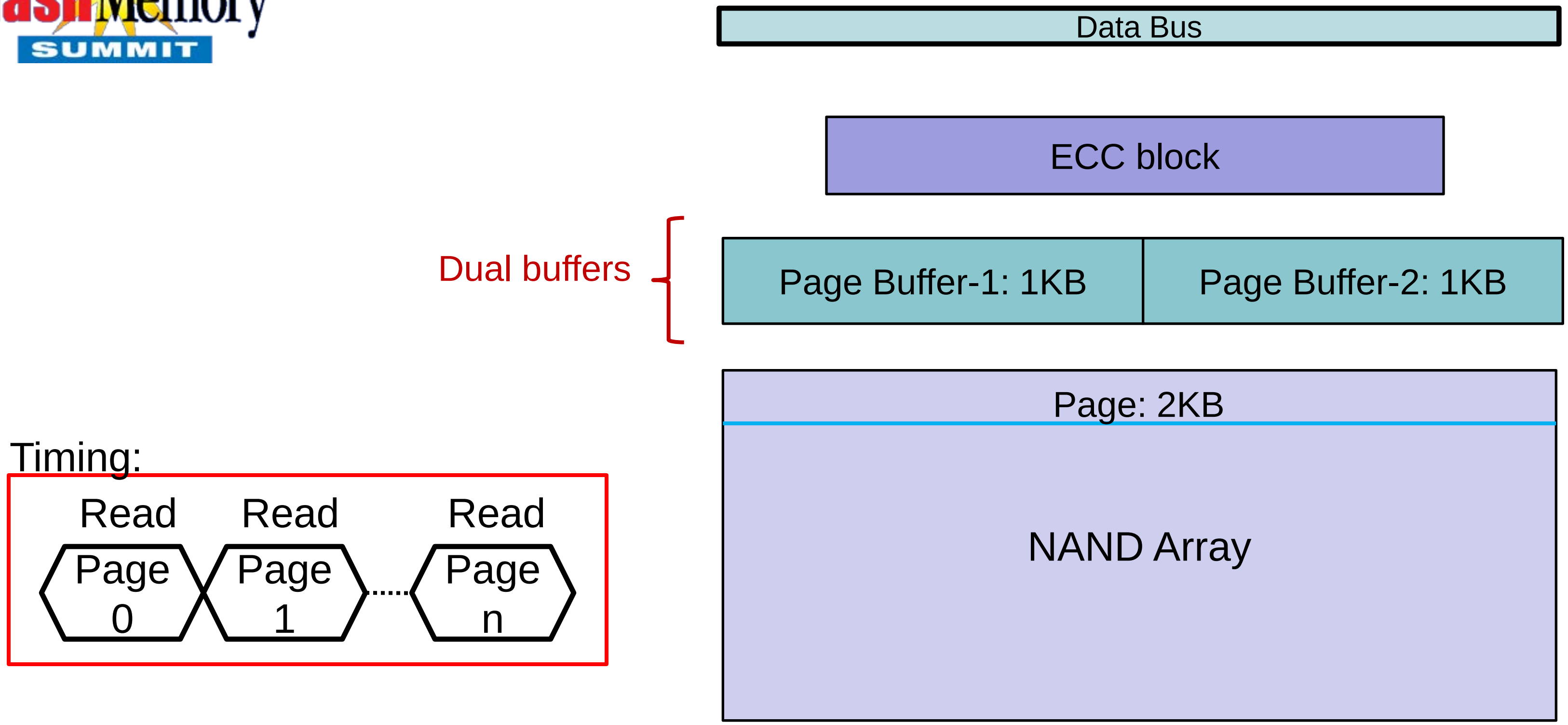
Two common NAND operations:

a. Page read operation : 20uS (load time) + 30us (ECC time) ~ 50us
   - Page read time in NAND products without on-chip ECC ~ 20us

b. Data read operation : read out 2KB page data from PB through data bus

- Next Page read incurs 50us wait again ➔ big overhead in Code Shadowing applications !
  - Data read for next page can only start after completion of next page read operation

New Arch : Dual-buffer SPI-NAND

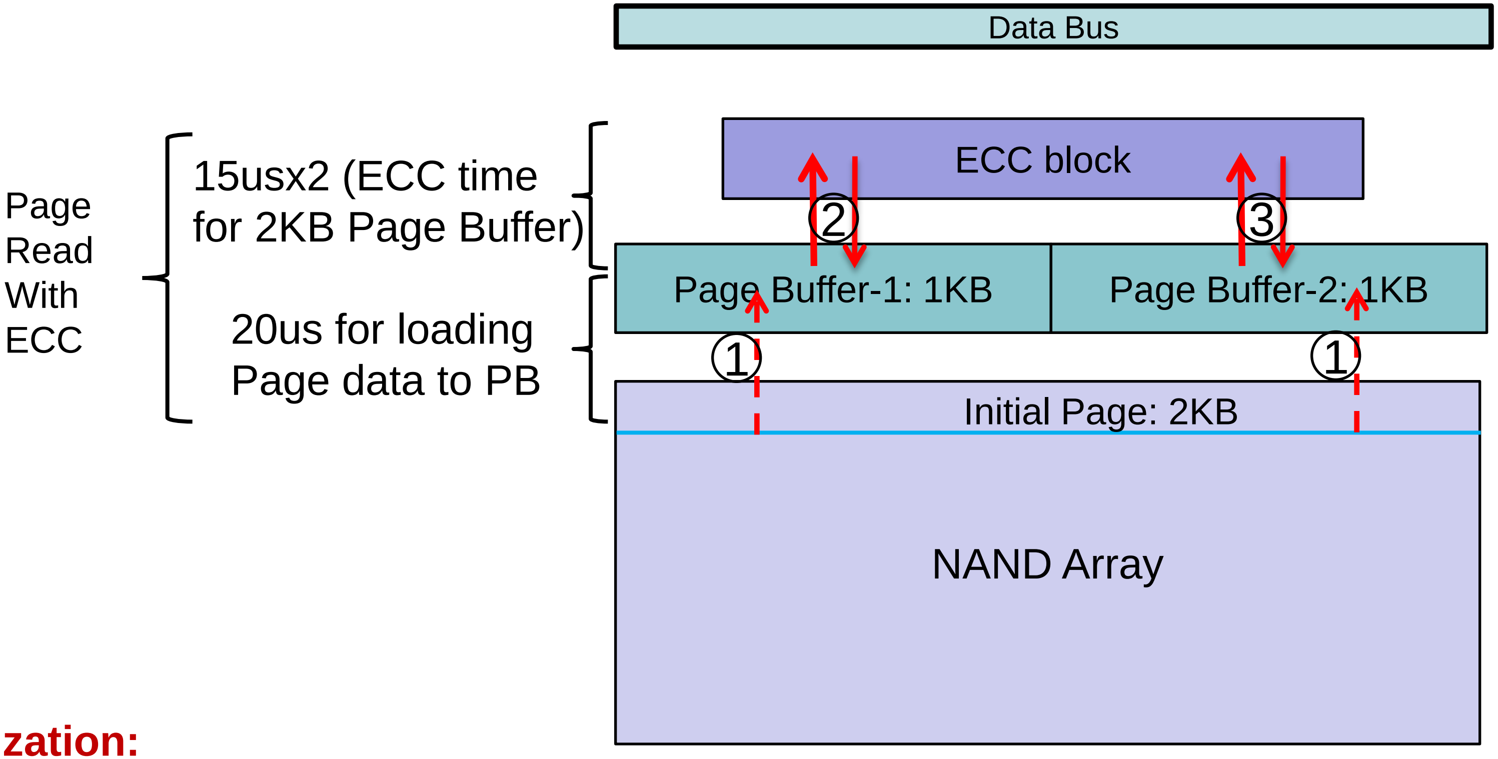


- New Arch uses Dual-buffers, while retaining die size benefit offered by Single Array/Plane
- "Continuous read" operation offered with no gaps at page and block boundaries !
    - Continuous read is achieved by interleaved read from dual Page Buffers
    - Page read time (load data to PB + ECC) is completely hidden from the read thru-put

➢ Continuous read in new Arch comprises 2 steps : (i) Initialization, and (ii) Interleaved read

Continuous read operation in new Arch:

1st step : “Initialization”



Initialization:

- Above operation 1+2+3 (50us, i.e. 20us+15us+15us) is one-time latency prior to starting of each continuous read operation (note: step 3 during initialization is optional)

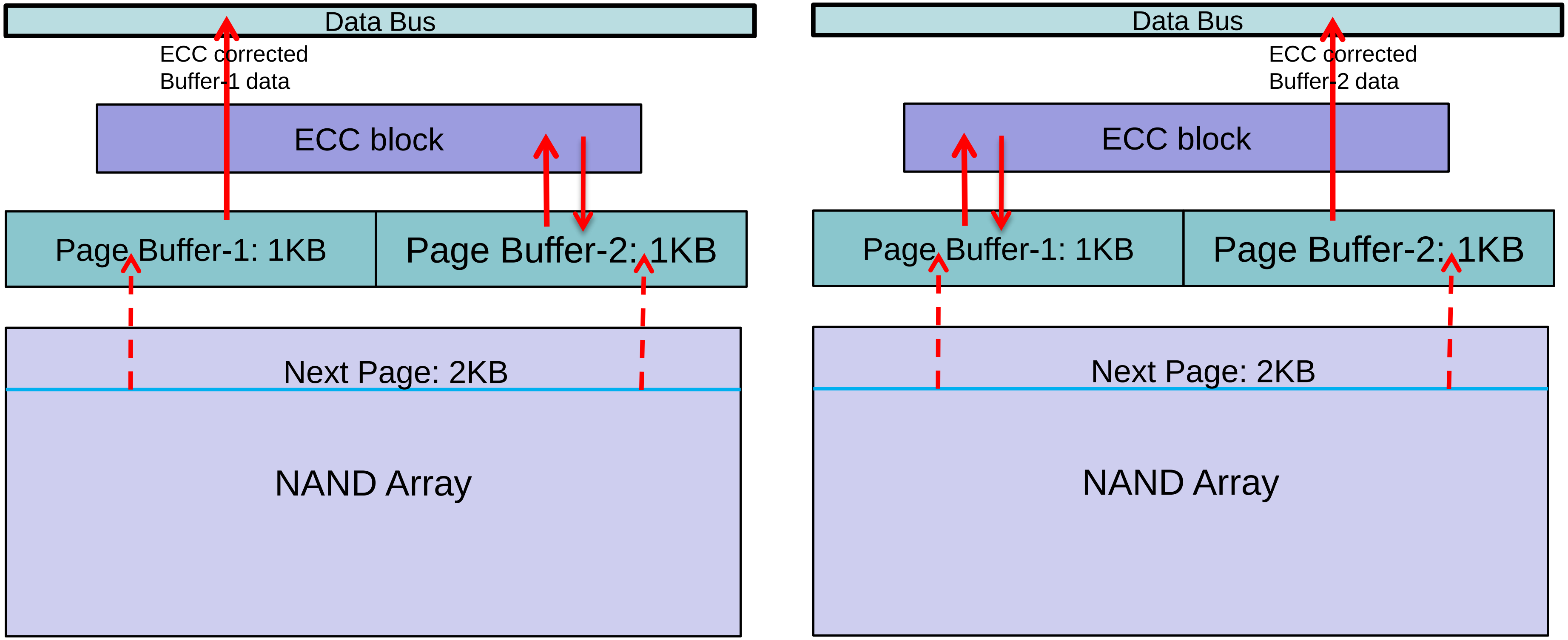
➢Initialization can be hidden in P/U sequence, with Page-0 read (including ECC) during power-up

  - Many code-shadow applications starting download from Page-0/byte-0 may incur no latency

- Appls with code-shadow starting from other than Page-0 may incur initial 50us page read latency

Continuous read operation in new Arch:

2nd step : repetitive "Interleaved read"



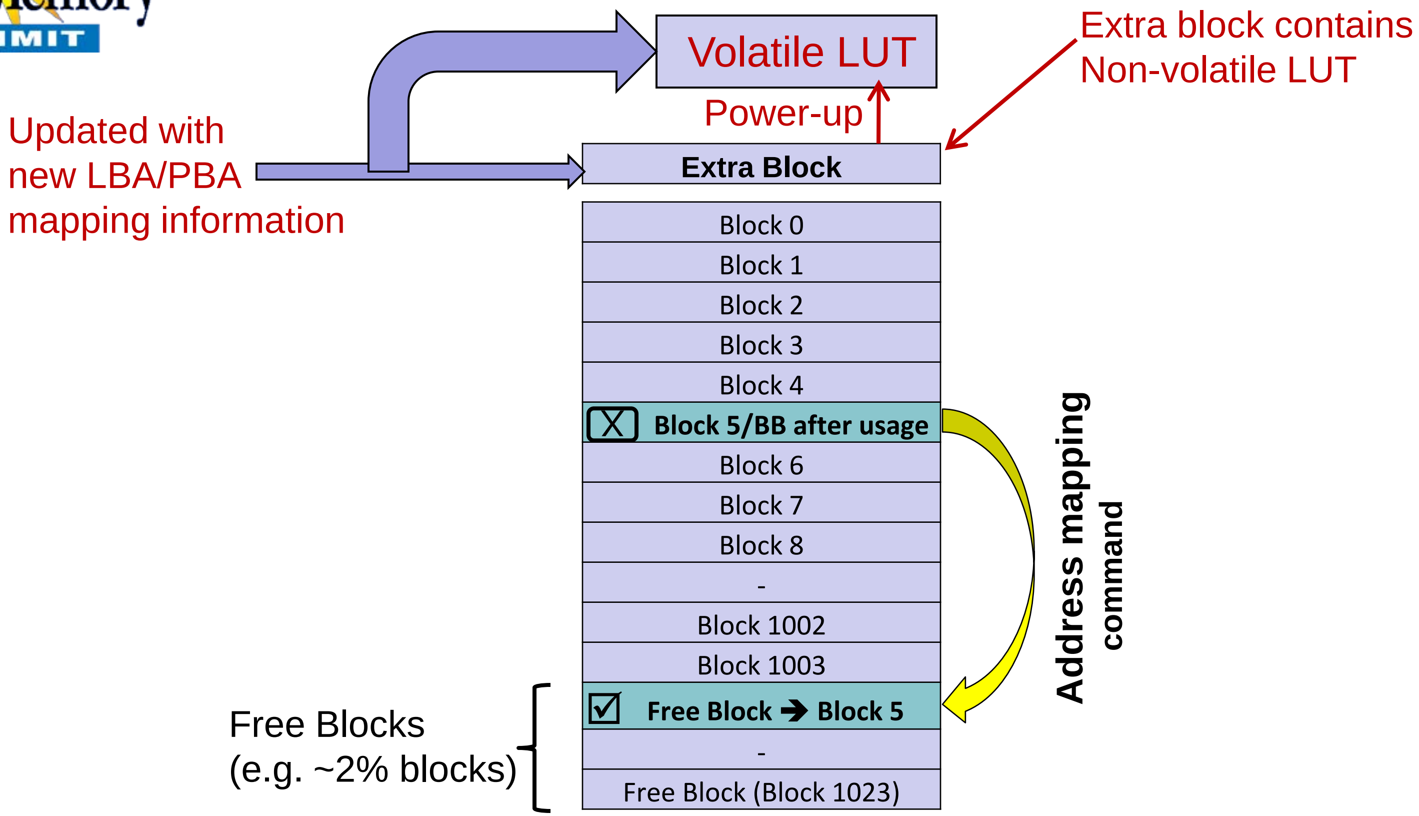
Interleaved read:

- Buffer-1 data is outputted while ECC is carried out on Buffer-2 data, and vice versa
  - Data from next page can transfer concurrently to PB, since PB has 2-levels (L1/L2 latches)
- Pipelined operation with 3 concurrent steps: Data out from PB, ECC calculation, load data to PB

➢ Data read out continuously without gaps, by alternately reading from Buffer-1/Buffer-2

New Arch: On-chip Bad block management

LBA to PBA mapping

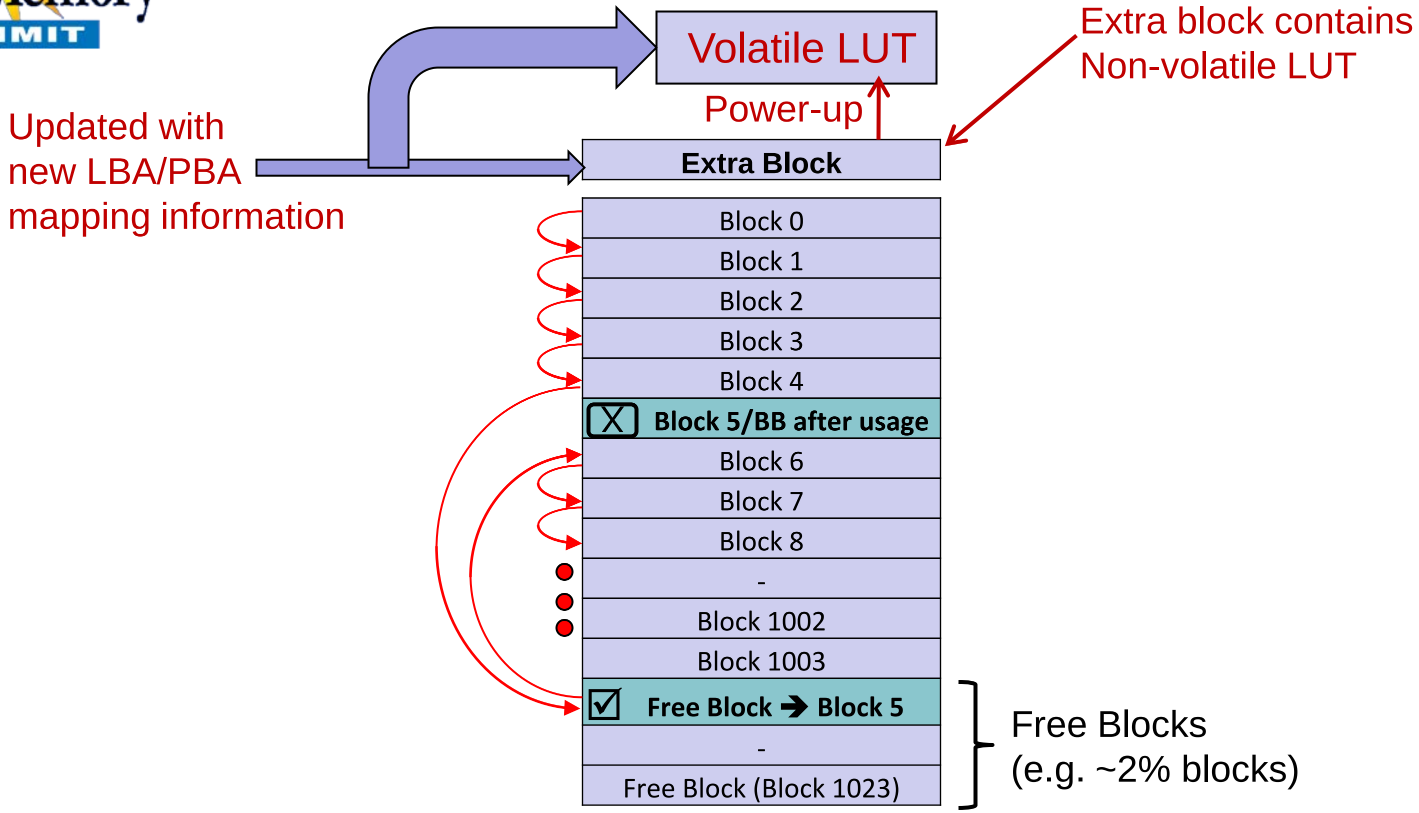


- User may issue “address mapping” command, e.g. on Erase/Verify or Program/Verify fail
  - Along with address mapping command, user provides LBA to PBA mapping information
  - Mapping information (LBA to PBA) saved in both non-volatile and volatile LUT
  - On each P/U (power-up), mapping information from non-volatile LUT read into volatile LUT

➢ **Need for ext. controller chip with LUT (for LBA/PBA mapping info.) has been eliminated**

New Arch: On-chip Bad block management

Continuous read (after LBA to PBA mapping)



- Red arrows show continuous read traversing from block-to-block
  - Bad blk skipped and read redirected to good replacement blk (using on-chip LUT for BBM)

➢ **Fast continuous read across block boundaries not possible with LUT in ext. controller**
  - On-chip volatile LUT with fast access facilitates high speed cont. read in case of bad block
  - On each P/U (power-up), mapping information from non-volatile LUT read into volatile LUT

Summary:

- A new Arch has been described for Code-Shadowing applications
  - New dual-buffer Arch provides Interleaved continuous/sequential read
  - Small die size due to Single Array/Plane key to fit 1Gb/2Gb in 8-pin pkg.

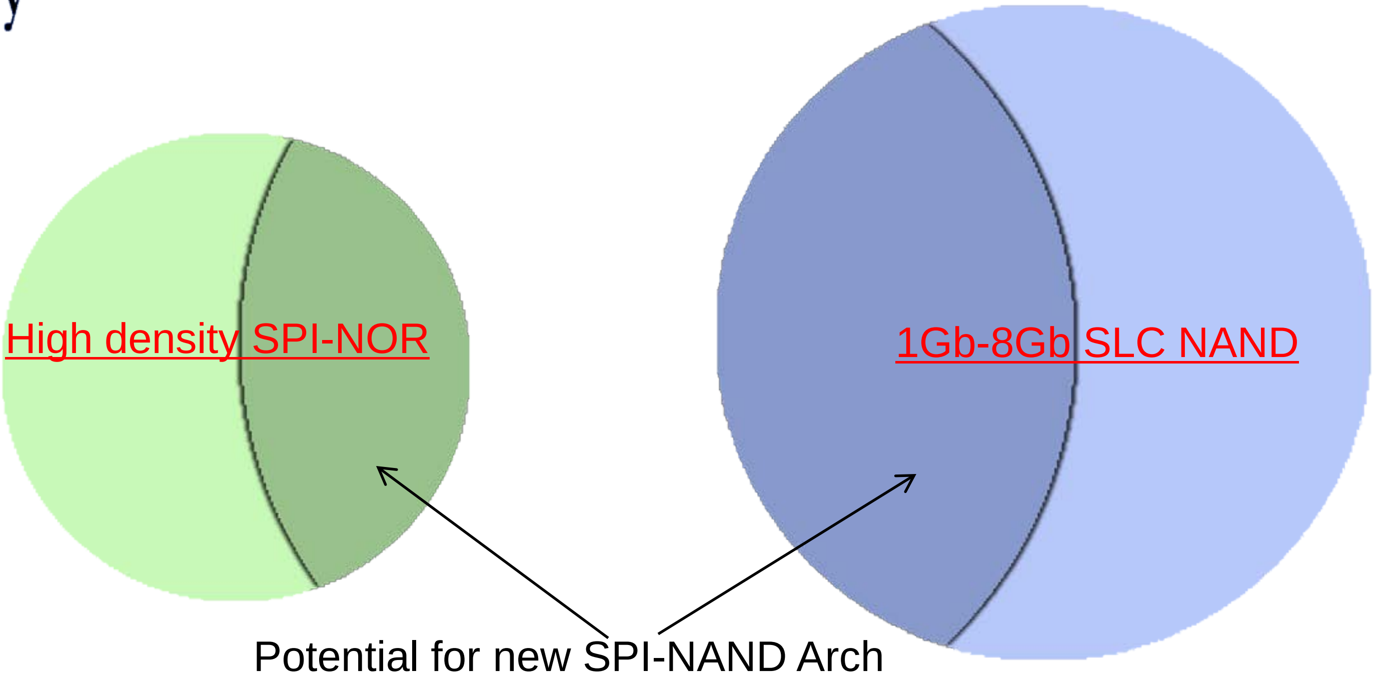
- Data is read out continuously without gaps at Page and Block boundaries
  - Continuous read operation at 104MHZ and even at 133MHZ is feasible

- Continuous reading out of data is compatible with SPI-NOR Flash
  - Package, pin-out, command, and clocking compatibility with SPI-NOR

- Unlike traditional NAND, no need of ext. cont. chip for ECC and LUT (BBM)
  - Fully integrated on-chip solution has been proposed

➢ **Building blocks of New Arch (dual buffer, on-chip LUT for BBM) have been described to achieve high speed continuous/sequential read in code-shadowing applications**

Future potential, projection, and discussion



- New SPI-NAND Arch will overlap with 1Gb-8Gb SLC NAND, in addition to high density SPI-NOR
- Read thru-put (Quad, sequential) better than SLC NAND (8/16 pin), and 8-pin foot-print attractive
- SPI-NOR introduced in 90's evolved to replaced many Parallel-Flash appls due to 8-pin foot-print
  - Evolution was accelerated due to higher thru-put offered by multi I/O SPI, e.g. Quad I/O

➢ Similar trend could take place in migration from 1Gb-8Gb SLC NAND to new SPI-NAND Arch

Thanks