# Data Storage Using a Non-integer Number of Bits per Cell

Naftali Sommer
June 21st, 2017

The Conventional Scheme

- Information is stored in a memory cell by setting its threshold voltage

  - Many cells are programmed and read simultaneously

- Conventional Flash memory devices store an integer number of bits per cell

  - Number of voltage states is a power of 2

- Easy to program, read and protect by Error Correcting Codes (ECC) using simple schemes


1 bit/cell

2 bits/cell
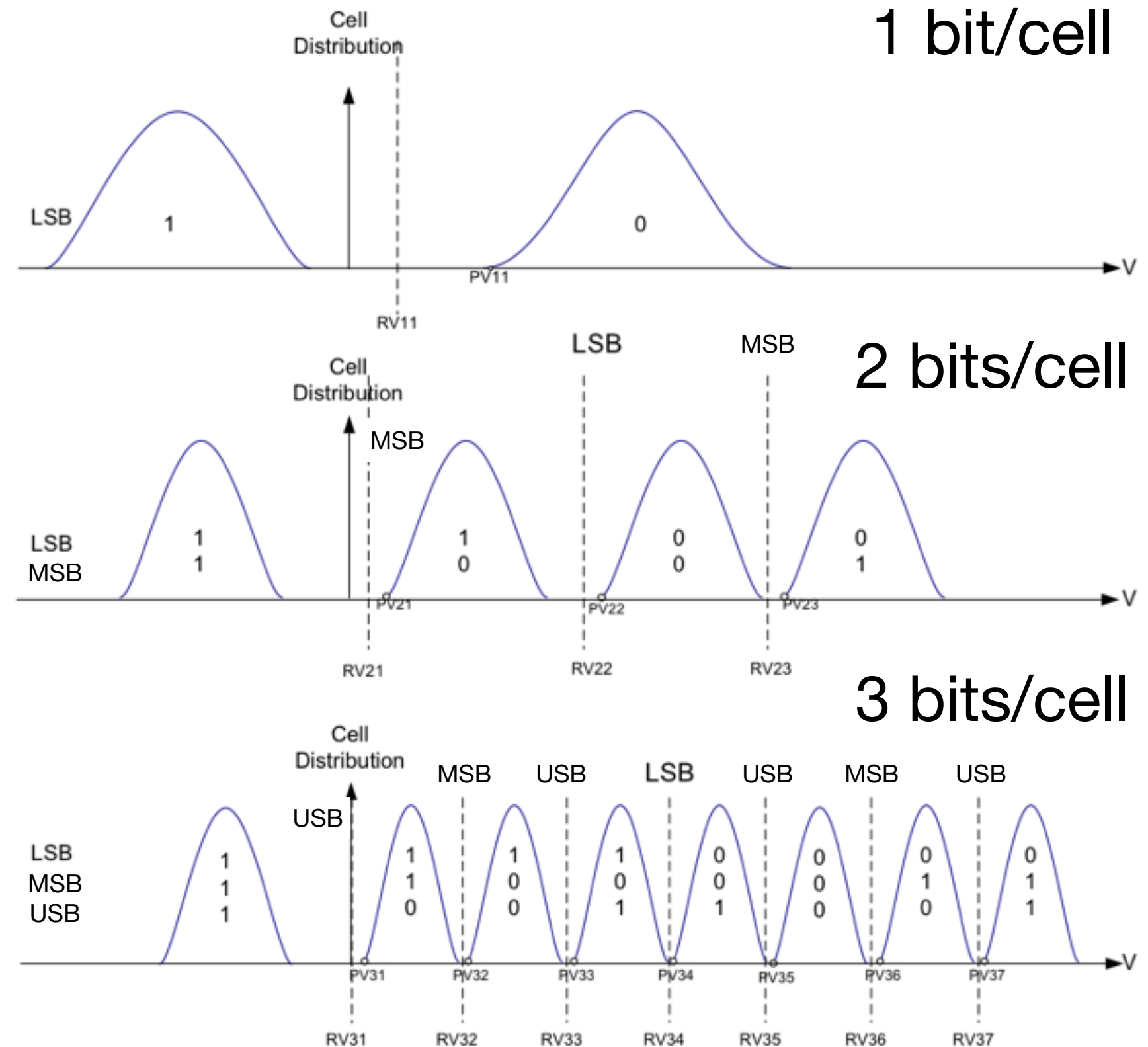
3 bits/cell

The Conventional Scheme - 3 Bits/Cell Example
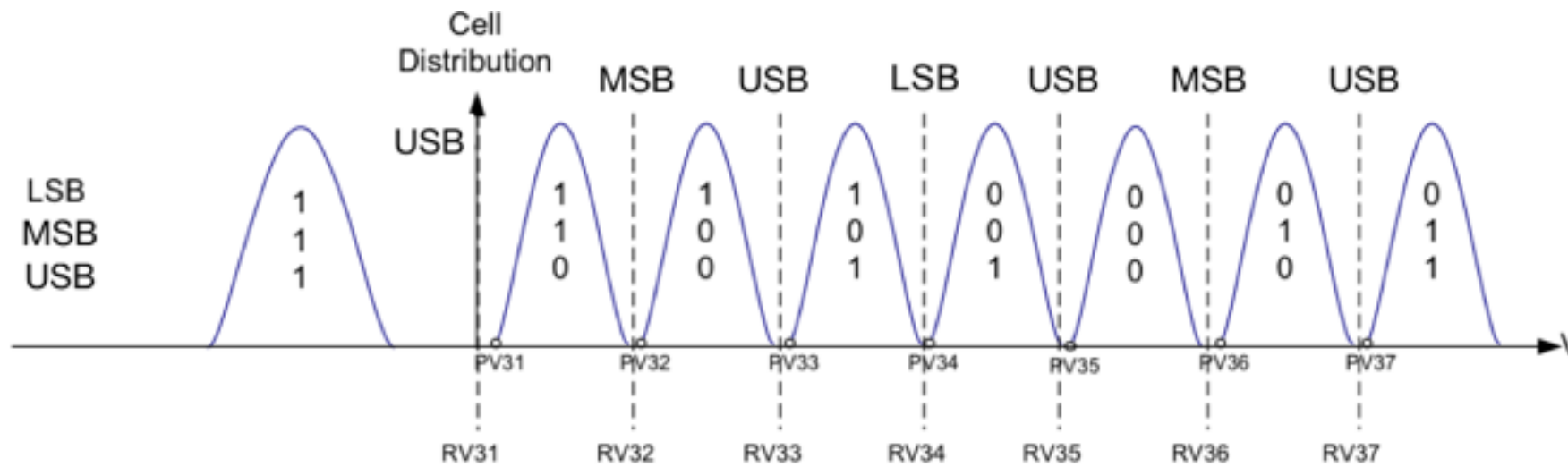
- Multi-level coding - 3 independent logical data pages are written to the same cells

  - Denoted LSB, MSB, USB

- Due to the Gray mapping, each page can be independently read with a relatively small number of sense operations

  - LSB - single sense (RV34), MSB - 2 senses (RV32, RV36), USB - 4 senses

  - Average read latency of (1+2+4)/3 = 2.33 senses per page

  - Beneficial for random read scenarios

- A binary Error Correcting Code (ECC) can be independently applied to each page

# Motivation for a Non Integer Number of Bits/cell
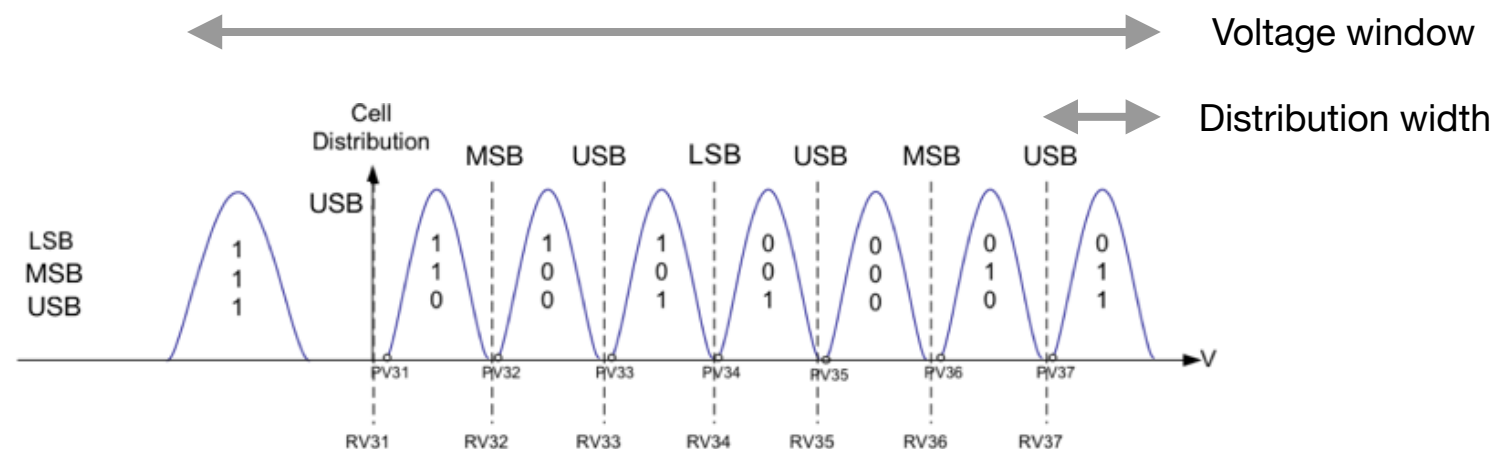
- The distribution width and the available voltage window define the equivalent Signal to Noise Ratio (SNR)

- This SNR limits the net number of bits/cell that can be reliably stored

- Example: assume that the available SNR enables net storage of 2.5 bits/cell

  - 2 possible schemes:

Number of voltage states	Number of bits/cell	ECC code rate	Net number of bits/cell
8	log2(8) = 3	0.83	2.5
7	log2(7) = 2.8	0.89	2.5

  - Advantages of the 7 states scheme

    - Simpler implementation of the ECC - lower power consumption and die size

      - Higher code rate, less errors to correct

    - Faster to program

6/20/17

Problem Statement

- We would like to design a practical mapping scheme for a non power-of-2 number of states (e.g. 9, 7, 6, 5, 3) that will enable:

  - Simple ECC scheme

  - Short read latency for random read

- Potential solution - use non-binary symbols, each symbol mapped to a cell

  - Practical non-binary ECC schemes exist (e.g. LDPC, BCH)

  - However - random read operations will have high latency

    - For example, 6 sense operations are needed to distinguish between 7 states

      - Much higher than the 2.33 sense operations shown in the previous slides for 8 voltage states

- Main challenge - how to map several independent pages to the same cells, such that different pages are read with disjoint sense operations

Proposed Scheme for the 7 States Case
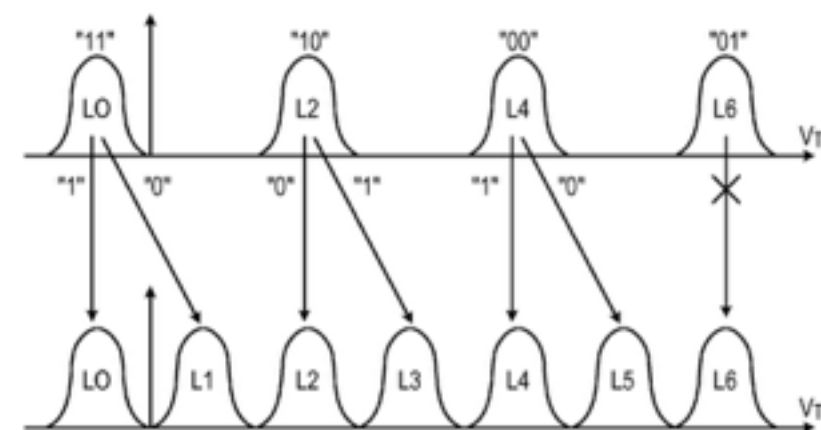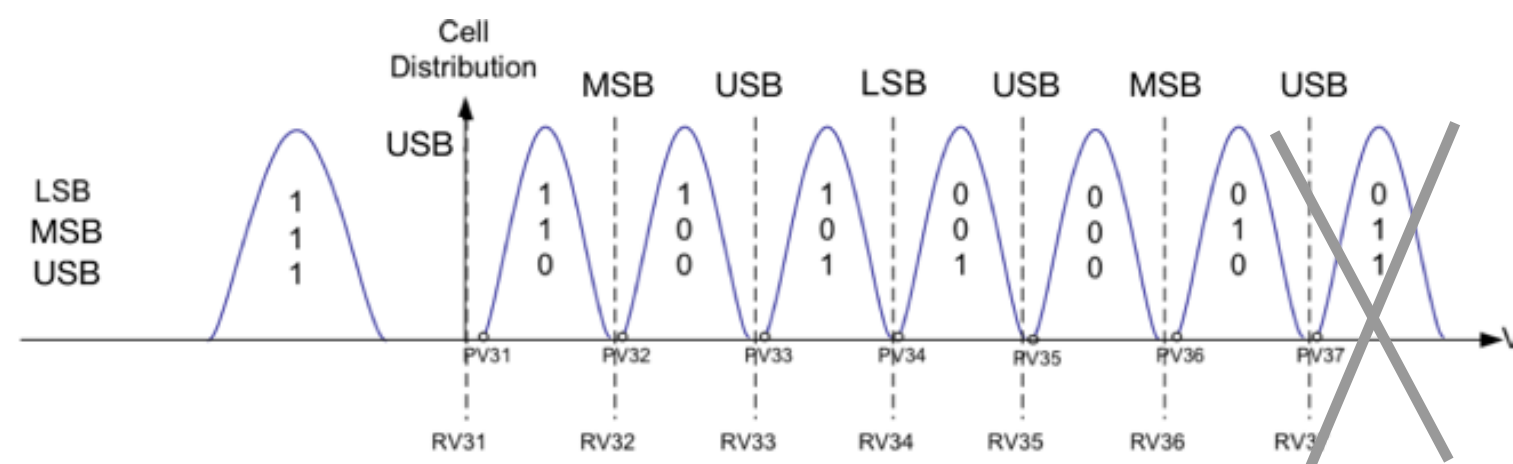
- Proposed scheme - map 3 independent pages as done for 8 states, but don't use the highest level

  - Problem - what to do if {LSB, MSB, USB} = {0, 1, 1}?

  - Solution - write LSB & MSB pages as before, but map USB bits only to cells for which {LSB, MSB} = {11, 10, 00}

    - Only 3/4 of the cells can be used for storing the USB page

      - USB page size will be 3/4 of the LSB and MSB page sizes

      - For example: LSB and MSB are 16 KByte, USB is 12 KByte

  - Conditionally invert the LSB and MSB data before programming in order to make sure that at least 3/4 of the cells will enable a USB bit programming

- Equivalent to storing 2.75 bits/cell

  - Close to the theoretical limit of log2(7) = 2.8 bits/cell

Proposed Scheme for the 7 States Case - Cont'd
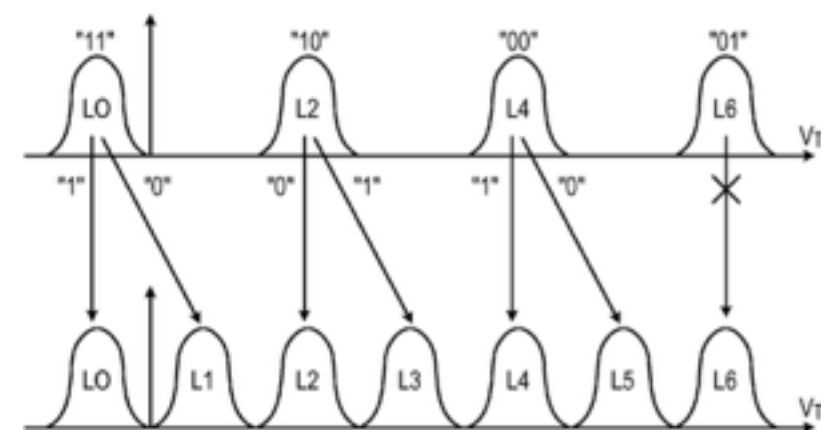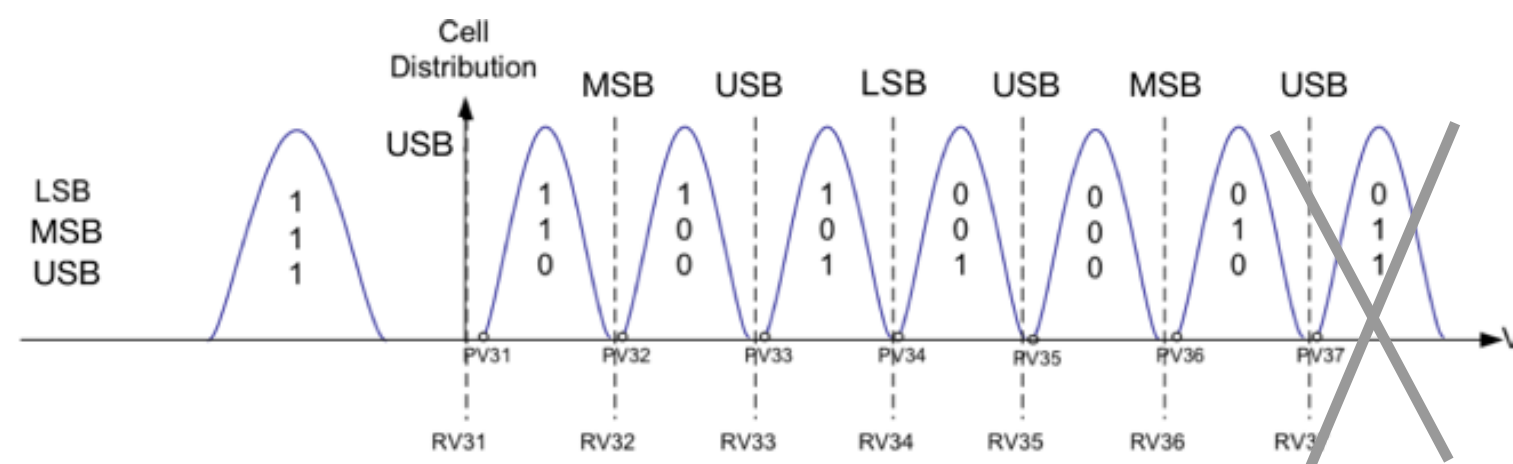
- A binary ECC is applied independently to each page, similar to the conventional scheme

- Read latency

  - First 2 pages are read with low-latency in the same way as in the conventional scheme

    - Single sense for LSB, 2 senses for MSB

  - For reading the 3/4-sized USB page, we first need to read and ECC decode the first 2 pages

    - Need to know in which cells the 3/4 page is stored, and must know it error-free

  - A total of 6 sense operations are needed to read the USB page

  - Average read latency: (1*1 + 1*2 + 0.75*6)/2.75 = 2.73 sense operations

    - Worse than the 2.33 of the conventional scheme, but still much better than the 6 of the non binary symbols scheme

Extensions

- The scheme can be extended to any number of voltage states

  - Examples:

    - 2.5 bits/cell using 6 voltage states by programming 2 full pages + a half-sized 3rd page to half of the cells

    - 1.5 bits/cell using 3 voltage states by programming a full page + a half-sized 2nd page to half of the cells

    - 3.25 bits/cell using 10 voltage states by programming 3 full pages + a 1/4-sized 4th page to 1/4 of the cells

- The scheme can be extended to other types of Gray mapping