Institute for Communications Engineering
Department of Electrical and Computer Engineering
Technical University of Munich

# Cyclic Construction for Masking Partially Stuck-at-1 Memory Cells

Haider Al Kim, Antonia Wachter-Zeh, Sven Puchinger

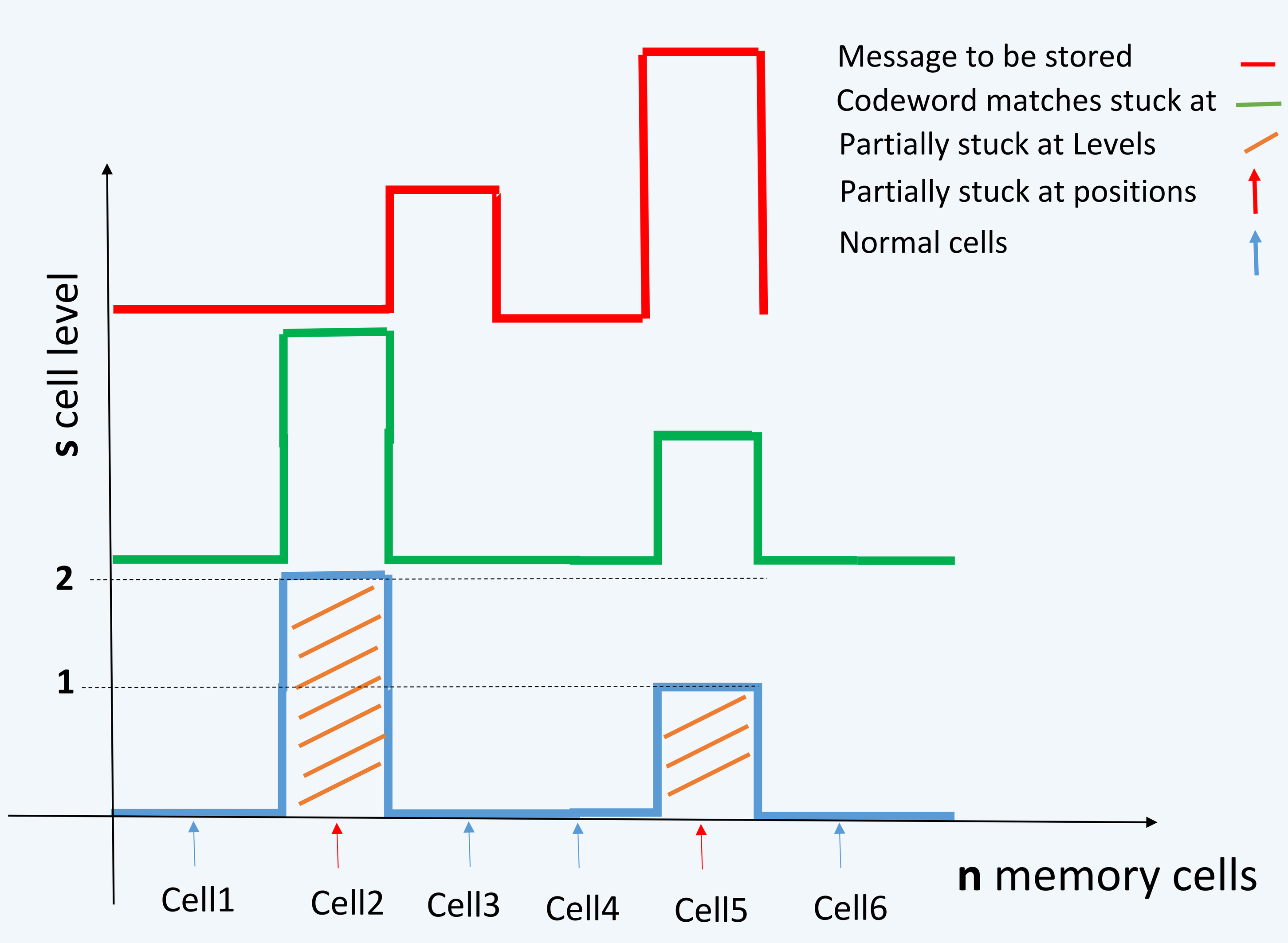
haider.alkim@tum.de, antonia.wachter-zeh@tum.de, sven.puchinger@tum.de

## Introduction

The dominance of non-volatile memories and PCMs(phase change memories) as memory solutions for variety of applications have brought the attention to the pros and cons of these types of memories.

- **Problem Description** : the memory cells are stuck at a state where it cannot change its level [1]. Different PCM cells' levels (q-ary cells) have multi-level states [3]."Stuck" means that the cell's charge is trapped in the cell. The trapped scenario might happened due to a defect in a cell.
- **Solution** : store new information is by increasing its trapped level. We need to find a codeword that matches the states of the partially stuck at cells.
- **The Process** : we create a new vector that will be used to mask the partially stuck at-1 memory cells. Then we add it to the information vector, so that the resulted vector will hide the stuck-at levels of the memory.
- **Previous works** : as shown in [1] and [2] , the researchers in [1] have proposed different solutions for stuck at cell. However, the redundancy was sacrificed.
  While in [2], more improvements are achieved in terms of the redundancy to be 1 and even less than 1 (lower bound speaking). However, additional error correcting besides the cyclic construction were not considered in [2].
- **The Focus** : this work will present the cyclic construction for masking partially-stuck-at-1 level. Future coming work is about improving this theorem to correct additional random errors happening while storing or retrieving information from the memory cells.

### Figure shows Partially Stuck At Least 1 Memory Scenario




### Theorem ($s_i = 1, u < q$)

If $u < q$, then there is a u-PSMC built using cyclic code with generator polynomial $g_0(x)$ and parity check polynomial $h_0(x)$ which is corresponding to the Matrix $H_0 = (1111, \dots, 1)$ such that :

- The length is $n$.
- $k_0 = n-1$, the maximum length of the information aimed to be stored. Then , the cardinality is M = $q^{n-1}$.
- $g_0(x)|(x^n-1)$, Since $h_0(x) = 1+x+x^2+\dots+x^{k_0} \Rightarrow g_0(x) = 1+x$, single parity symbol for masking the stuck-at-1.
- $s_i = 1$ "partiality stuck" , means the level at least $(1)$ , there is a scaler $\mathbf{z_0}$ such that it guarantees the masking polynomial called $\mathbf{x}(\mathbf{x}) = \mathbf{z_0}.h_0(x)$ when added to the information polynomial called $w(x)$ will not get **0** values in the partially stuck-at-1 positions $\iff \mathbf{z_0} \neq \mathbf{0}$.

Then, with this code, we can mask u partially stack-at-1
$C_0(x) = w(x) + x(x) \Rightarrow C_0(x) = m(x).g_0(x) + z_0.h_0(x)$.

## Input

- Message :
  **m(x)** $\in F_q^{\mathbf{k_0}}$ , degree $\leq \mathbf{k_0}$.
- Positions of partially stuck cells:
  $\phi_0, \phi_1, \phi_2, \dots, \phi_{u-1} \subseteq [n]$.

levels of stuck cells is 1:
$s_i = 1 \in F_q$ , at least 1 , so no need to put it as an input.

**The notation** $[a] = [0, 1, \dots, a-1]$.

## Algorithm

1. Define $H_0$ as $H_0^{(n-k_0)*n} = (1111, \dots, 1), r = n - k_0 = 1$. Then $h_0(x) = \sum_{i=0}^{k_0} h_0.X^i = 1+x+x^2+\dots+x^{n-1}$, since $k_0 = n-1$ .
2. So, $g_0(x) = 1+x$ , since $g_0(x).h_0(x) = x^n - 1$.
3. Choose $z(x) = \sum_{i=0}^{n-k_0-1} z_i.X^i \Rightarrow z(x) = \sum_{i=0}^{n-(n-1)-1} z_i.X^i$, where $k_0 = n-1 \Rightarrow z(x) = \sum_{i=0}^{0} z_i.X^i \Rightarrow z(x) = z_0$, a specific scalar(single value).
4. The partially stuck at memory level should not go below 1.
5. So, fulfill the following:
   $w_{\phi_i} + (z(x).h_0(x))_{\phi_i} = \mathbf{q} - \mathbf{s_i}, \forall i \in [u] \& s_i = 1 \iff z_0.H_{0,u} = (q-1-w_{\phi_0}, \dots, q-1-w_{\phi_{u-1}})$ , then $\rightarrow z_0, z_0, z_0, \dots z_{u-1} = (q-1-w_{\phi_0}, \dots, q-1-w_{\phi_{u-1}})$
6. $z_0 = (q-1-w_{\phi_0}, \dots, q-1-w_{\phi_{u-1}})$ ,and $z_0 \neq \mathbf{0}$.
7. $\mathbf{C_0}(\mathbf{x}) = \vec{\mathbf{w}} + \vec{\mathbf{x}} \Rightarrow \mathbf{C_0}(\mathbf{x}) = \vec{\mathbf{w}} + \mathbf{z_0}.H_0$. $\Rightarrow$ output vector **c** masks all u partially stack-at-1 cells.

### Encoding Example

Let $q = 3$ and $n = | F_{q^2} | - 1 = 3^2 - 1 = 8$ and we want to store the message $m$ that is (0210210) $\in F_q^{\mathbf{k_0}}$. The partially stuck positions named $\phi_i$ are $\phi_2$ and $\phi_5$ , $\forall i \in u \leq n$ and $u$ is the number of stuck at cells. Then according to the process we need to find the following:

1. First calculate $w(x)$.
   - $m(x) = 2x + x^2 + 2x^4 + x^5$.
   - $g_0(x) = 1 + x$.
     $w(x) = m(x).g_0(x) = (2x+x^2+2x^4+x^5).(1+x)$
     $\Rightarrow w(x) = 2x+x^2+2x^4+x^5+2x^2+x^3+2x^5+x^6$, coefficients $\mod 3$ , then we get:
     $$w(x) = 2x + x^3 + 2x^4 + x^6$$
   - Or we can write it as a vector (02012010) $\in F_q^n$.
2. Second calculate the $x(x)$.
   - $x(x) = z(x).h_0(x)$.
   - Find z(x) that should not be zero. So that we need to fulfill the following equation and since $z(x) = z_0$:
   - $[z_0, z_0] = [q-1-w_{\phi_1}, q-1-w_{\phi_5}]$ , where it is partially stuck at the positions $\phi_2$ and $\phi_5$
     $\Rightarrow [z_0, z_0] = [3-1-2, 3-1-0] \Rightarrow$
     $\Rightarrow [z_0, z_0] = [0, 2]$ , as $z_0 \neq 0$ , then $z_0 = 2$.
   - $x(x) = 2*(1+x+x^2+x^3+x^4+x^5+x^6+x^7)$
     $$\Rightarrow x(x) = 2+2x+2x^2+2x^3+2x^4+2x^5+2x^6+2x^7$$
   - Or can write it as a vector (22222222) $\in F_q^n$.
3. Now find $C_0(x) = w(x) + x(x)$.
   - $C_0(x) = 2x+x^3+2x^4+x^6+2+2x+2x^2+2x^3+2x^4+2x^5+2x^6+2x^7$
     $$\Rightarrow C_0(x) = 2+x+2x^2+x^4+2x^5+2x^7$$
   - Or can write it as a vector (21201202) $\in F_q^n$ , as it is shown the positions with partially stuck at cell are masked with at least 1.

### Decoding Steps

- Reverse the process.
- First : subtract $x(x)$ from $C_0(x)$ to get $w(x)$ that has the original information.
- Second : from $w(x)$ we get $m(x)$ that equals $w(x)/g_0(x)$.
  Note that the degree of the term $m(x).g_0(x)$ will be $(\leq n)$.

### Next Planned Improvement

- Using cyclic code in this construction for masking partially stuck at 1 did not consider correcting additional errors as the one in [1].
- We need additional symbols to correct $t$ errors and get minimum distance $d = 2*t+1$.
- This could be applied using BCH code(a sub-filed sub-code) of a GRS code. Both have cyclic properties.
- But, when choose $k_0 = n-1$ then there is no more left length in $n$ to be used for getting the required symbols for error correction. HAPPY FOR SUGGESTIONS!

### References

[1] C. Heegard, "Partitioned linear block codes for computer memory with 'stuck-at' defects," IEEE Trans. Inf. Theory, vol. 29, no. 6, pp. 831–842, Nov. 1983.

[2] A.Wachter-Zeh and E. Yaakobi, "Codes for Partially Stuck-At Memory Cells," IEEE Transactions on Information Theory, vol. 62, no. 2, pp. 639–654, February 2016.

[3] G. W. Burr et al., "Phase change memory technology," J. Vac. Sci. Technol. B, vol. 28, no. 2, pp. 223–262, 2010.



European Commission