Fault-Tolerant Multi-Level Logic Decoder for Nanoscale Crossbar Memory Arrays [*]

M. Haykel Ben Jamaa[1], Kirsten E. Moselund[2], David Atienza[1,3], Didier Bouvet[2],
Adrian M. Ionescu[2], Yusuf Leblebici[4], and Giovanni De Micheli[1]

[1]Integrated Systems Laboratory (LSI), EPFL, 1015 Lausanne, Switzerland.

[2]Electronics Laboratory (LEG), EPFL, 1015 Lausanne, Switzerland.

[3]DACYA, Complutense University (UCM), 28040 Madrid, Spain.

[4]Microelectronic Systems Laboratory (LSM), EPFL, 1015 Lausanne, Switzerland.

ABSTRACT

Several technologies with sub-lithographic features are targeting the fabrication of crossbar memories in which the nanowire decoder is playing a major role. In this paper, we suggest a way to reduce the decoder size and keep it defect tolerant by using multiple threshold voltages (V_T), which is enabled by our underlying technology. We define two types of multi-valued decoders and model the defects they undergo due to the V_T variation. Multi-valued hot decoders yield better area saving than n-ary reflexive codes (NRC), and under severe conditions, NRC enables a non-vanishing part of the code space to recover. There are many combinations of decoder type and number of V_T's yielding equal effective memory capacities. The optimal choice saves area up to 24%. We also show that the precision of the addressing voltages for decoders with unreliable V_T's is a crucial parameter for the decoder design and permits large savings in memory area.

1. INTRODUCTION

At the end of roadmap for CMOS technologies, the top-down approach still provides many technological tools for continued downscaling. The device performance is improved by boosting the mobility while choosing different materials for the channel, straining it or rotating the substrate. Undesirable coupling effects between drain and channel can be suppressed by using *Silicon on Insulator (SOI)* or *Silicon on Nothing (SON)* devices. Advanced control of etching allows getting over the photolithographic limitations; several devices are based upon this gain, for instance *Gate-All-Around Field Effect Transistors (GAA FET)* [12], Double Gate FET [31] and Fin FET [10].

Since the top-down downscaling is becoming more challenging, many efforts were put in bottom-up approaches, which proved that another category of breakthroughs will permit additional scaling of circuits. The ability to design, fabricate and manipulate molecular-scale devices below the lithographic limits was recently demonstrated [4, 14, 27]. They include one- or two-dimensional devices representing molecular switches with a large conductivity change depending on the molecular state, as well as thin nanowires, promising a very large integration density. The vanishing scale gap between these bottom-up techniques and the semiconductor industry suggests a *hybrid* approach consisting in integrating nanoscale wires and switches onto standard CMOS circuits by self assembly when the photolithography fails in achieving the targeted density.

Both top-down and bottom-up approaches support a high device integration in a regular architecture called *crossbars*; these represent a simple circuits formed by two perpendicular strips of parallel nanowires storing the information or executing the computation at the matrix nodes (*i.e.* at the nanowires crosspoints) [2, 5]. The crossbar circuits can be realized as a separate part on the CMOS chip, where non-standard CMOS processing steps may be performed in order to achieve the sub-lithographic dimensions. Then, this part has to be connected to the outer CMOS part which is defined on the lithographic scale. One of the challenges is the design of the interface between the nano- and meso-scale parts. This is the decoder circuit, which allows the addressing of each single wire in the nanoscale part by the mesocale part.

Previous approaches for designing the decoder range from a random attribution of the nanowire addresses [8, 20] to a deterministic mask-based decoder [1]. Unfortunately, the uncertainty of placement induces a large overhead in the decoder size, up to 5 times [15]. In this work, we will present a method to reduce the decoder size by implementing it in a multi-valued logic. This idea is founded on our underlying technology based on *Gate-All-Around Silicon Nanowires (GAA SiNW)* [25] which enables the fabrication of devices with multiple threshold voltages V_T. Since V_T shows a certain variability which increases when the device size shrinks, many defects are expected to happen; but the decoder robustness can be enhanced by optimizing the encoding and design parameters.

This paper is organized in the following way. In the next section we review the work related to crossbar memories ranging from processes and devices to the memory architecture, and we focus on the decoder design. In Section 3, we present the decoder model and the underlying technology. Section 4 states the formalism and semantic used in multi-valued logic encoding. Then, we construct two kinds of encoding schemes (Section 5): the multi-valued hot and the n-ary reflexive codes. The defects affecting the decoders are defined and anlyzed in Section 6. We finally present the results of simulations performed on the decoder with both encoding schemes and under various reliability assumptions (Section 7), showing the area saving and highlighting the critical decoder parameters.

2. RELATED WORK

2.1 Devices and Circuits

Fabrication of silicon nanowires usually follows one of the two different approaches; bottom-up, where wires are grown, synthe-

[*]This work is partially supported by the CCMX/MMNS project, the CSI Center, the FNS Research Grant 200021-109450/1 and the Spanish Government Research Grant TIN2005-5619.

1-4244-1382-6/07/$25.00 ©2007 IEEE

sized or self-assembled onto the silicon substrate; and top-down, where *nanowires (NWs)* are fabricated from the substrate by more traditional means such as lithography, oxidation and etching. Both approaches present their unique advantages and disadvantages. Silicon nanowires with diameters of a few nanometer can be grown from gold nanocrystals dispersed on the silicon substrate [16], and electronic devices based on n- and p-type silicon nanowires have been fabricated by laser assisted catalytic growth [32]. The in-situ doping during the growth process, results in a radial [22] or axial [14] doping profile and makes it possible to differentiate NWs. The integration density which can be obtained with these wires can be very high; however, there is little or no control of the fluidic [30] placement of each wire, and thus contacting and interfacing to the rest of the circuit is difficult, and might reduce the benefits in terms of total circuit area.

On the other hand nanowires obtained by stress-limited oxidation [18], by e-beam lithography [19] or by spacer technology [3] might not rival in integration density, even if the wire dimension is comparable to grown wires, the pitch is generally limited in one way or another by the technology. Recently, a novel top-down process called SNAP proved the ability of achieving a lithography-independent pitch by using a superlattice as an imprint pattern [24]. Placement and contacting of NWs fabricated in a top-down process is generally not problematic, since the wires are defined by lithography, so the structures are aligned to the rest of the circuit. However, since they are identical, they necessitate a mask-based differentiation technique.

In order to implement a controlled connection between the NWs, some sort of crosspoint is required. This could be implemented by a bistable NEMS switch as shown in [27] using suspended nanotubes having an energy barrier between the two switched states. In one of the states the nanotube is suspended above the crossing nanowire resulting in a very high resistance between the two conductors, and in the other state the two nanotubes are brought into contact, which leads to a low resistance current path. Recently, there has also been wide interest in the use of molecular crosspoints. In [29] a molecular monolayer sandwiched between two metal electrodes shows hysteretic switching as well as tunable resistance properties. In [23] the use of bistable molecular switches in a 2D crossbar structure is investigated. The concept of these switches is based on the molecular properties, which can be switched from a high resistance state to a low resistance state, and vice versa, by the application of an external voltage.

Logic gates based on SiNWs and switching crosspoints were demonstrated in [17] and an architectural paradigm for building logic circuits was proposed in [5, 2], which relies upon a full logic out of NOR and OR planes. An architecture with cascaded sequences of OR/NOT planes is also proposed in [9, 6]. A different paradigm based on two-terminal components, NanoFabrics, is introduced in [13].

Besides interconnection and logic, SiNW crossbar arrays are highly suitable for information storage because of their regular structure and high density. The information is stored at the crosspoints in an ideally bi-stable molecular switch. A prototype for this kind of memory was reported in [23] and the writing, reading and erasing operations on information stored in molecular switches were demonstrated. Later on, it has been claimed that the integration of molecular switches may suffer from artifacts due to the binding between the switches and the wires [21]. In reference [3], the author improved the choice of materials and suggested a better design of the organic switch, claiming the elimination of the artifacts. The organization of the memory into smaller blocks and interfacing modules was presented in [7].

2.2 Decoders

The decoder is a circuit used in both logic and memories based on crossbar architecture. A schematic principle is shown in Fig. 1, where the aspect ratios are not precise. The use of two different V_T's makes it possible, by applying suitable voltages at the vertical wires, to control which one of the horizontal wires will conduct; while the other horizontal wires will have a high resistance. The vertical wires are called control or addressing wires. Their number, thus the size of the decoder, depends on the encoding scheme and technology, which in turn follows from the fabrication approach: The encoding of NWs can be based on a random distribution of patterns (bottom-up), or deterministic mask-based patterns (top-down).

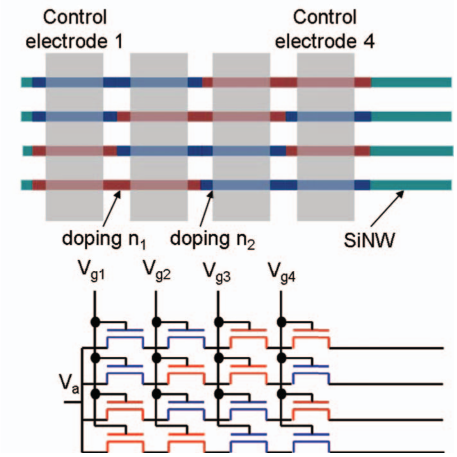


Figure 1: Decoder layout and circuit

Differentiated NWs have an axial or a radial doping profile which was defined during the NW growth process. An axial decoder was presented in [8]. In this decoder, the distribution of the V_T's is fully random. The NWs are laid out parallel to each other. They are addressable when they have different V_T patterns; this event has a certain probability that may be increased by increasing the number of addressing wires. On the other hand, the radial decoder [28] relies on NWs with several radial doping shells. After a series of etching of the NW shells, the remaining one in each region depends on the etching order. The suite of shells along the NW after all etching steps defines the NW patterns. Both axial and radial encoder give the same estimate of the number of *mesowires (MWs)* needed to address the available NWs; but the radial decoder is less sensitive to misalignment of NWs.

In order to control undifferentiated NWs a mask-based decoder has been presented in [1]. The MWs are separated from the NWs by an oxide layer which is not uniform: at some locations a high-κ is used, in the others a low-κ dielectric. The high-κ dielectric amplifies the electric field generated by the MWs relatively to the low-κ dielectric. Only NWs lying under the high-κ dielectric can be field-effect controlled by the MWs. The oxide mask is lithographically defined; making the encoder depending on the lithography limits.

There are other approaches for encoding undifferentiated NWs that are not mask-based. For instance, a random contact decoder has been presented in [20, 15], where the fully random connections between MWs and NWs through randomly deposited gold particles define the patterns of the NWs. Recently, an analog decoder called Micro to Nano Addressing Block, MNAB, was presented in [11]. It only needs two MWs, laid out parallel to the NW set. Depending on the pair of voltages applied at the MWs, the conduction in all NWs but one can be turned off.

3. CIRCUIT MODEL AND TECHNOLOGY

The memory architecture considered in this paper is depicted in Fig. 2. It consists of two parts, organized in an identical way, and laid out perpendicularly to each other. Each part is a strip of N parallel nanowires divided into a set of bundles; the nanowires within each group have a common ohmic contact to a mesowire at either extremity of the bundle. Each group is defined lithographically. We assume that the circuit has a standard binary nanowire group decoder, which is not investigated in this paper since its design is well established on the lithographic scale. M mesowires are used to address the nanowires within each group. Then, the nanowire decoder of each group has the size $N \times M$. The area sandwiched between the nanowire arrays is the actual memory, in which the information is stored in bi-stable switches grafted at the crosspoints.

Each nanowire within a single group has to be addressed; this is performed by the nanowire decoder which is formed by a set of parallel mesowires crossing the nanowire array. The part of na-

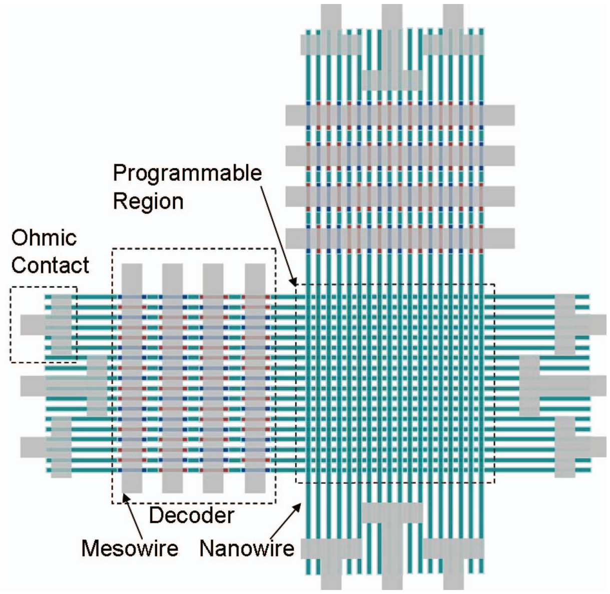


Figure 2: Crossbar memory architecture

nowires under the decoder is coated by a dielectric, thus allowing a field-effect control of the nanowires by the mesowires of the decoder. The differentiation between the FETs can be technologically achieved in a general way by any of the means described in the previous section.

The SiNWs fabricated to validate the proposed crossbar decoder architectures exploit the bulk silicon fabrication platform reported in [25]. The proposed access devices in the decoder are Gate-All-Around FETs. The nanowires are patterned into the doped Silicon substrate by a combination of isotropic etching and subsequent sacrificial oxidation. Whereupon a standard gate stack with 20 nm gate oxide is deposited, representing one of the decoder mesowires. Further details of the process can be found in [25].

The fabricated nanowires are parallel and have a length of several tens of micron, a cross-section circumference ranging from hundreds of nm to around 15 nm (smallest fabricated cross section diameter is 5nm), and a pitch depending on the lithography (optical or E-beam). Thus, in the proposed technology an increase in density, compared to CMOS, is not brought about by the lithographic scalability of the top-down process, but by using a unique cross-point for both writing and storing information. A scanning electron microscopy image of a suspended silicon wire before the gate stack is implemented as shown in Fig. 3.A, and cross-sections of the device are shown in Fig. 3.B and 3.C respectively. This device may serve as an ideal component in an addressing decoder scheme, the gate length corresponds to the width of the poly-lines connecting to the meso-world, whereas the nanoscale channel can serve a dual purpose as both access device, and as one of the nanowires in a crossbar memory.

The threshold voltage of the GAA FETs depends on the doping level of the channel, which is defined here by the substrate doping from which they are etched. Differentiation by doping is considered to reduce integration density, but since the top-down wires themselves are limited in pitch by the lithographic resolution in this technology, we do not consider this to be a major concern. In future ultra-scaled technologies, the differentiation might be possible by other means such as varying dielectric thickness. The second layer of parallel nanowires could be fabricated in a similar way, except the fact that it is patterned into polycrystalline Silicon deposited over the layer of nanowires, however this integration has not been demonstrated to date.

It is preferable to use different doping levels from the same type to complementary doping types (n and p), because p- and n-regions need to be connected by a metal wire, which may actually increase the decoder size. So, we are experimenting with more than two doping levels; i.e. more than two V_{T}'s, and the decoder of the memory will be based on a multi-valued logic. In the current technology, two different threshold voltages have been obtained by different bulk, and hence channel doping density. Multiple-V_{T} should be possible as well. The technology using doping differentiated top-down nanowires, is considered as a possible future embodiment of such a decoding scheme, but by no means the only one. The idea of multi-valued logic decoder derives from its feasibility in our technology, but it is not inherent to it. The following sections investigate the types of multi-valued logic encodings that can be implemented, their impact on the decoder area and memory density, and the related reliability aspects.

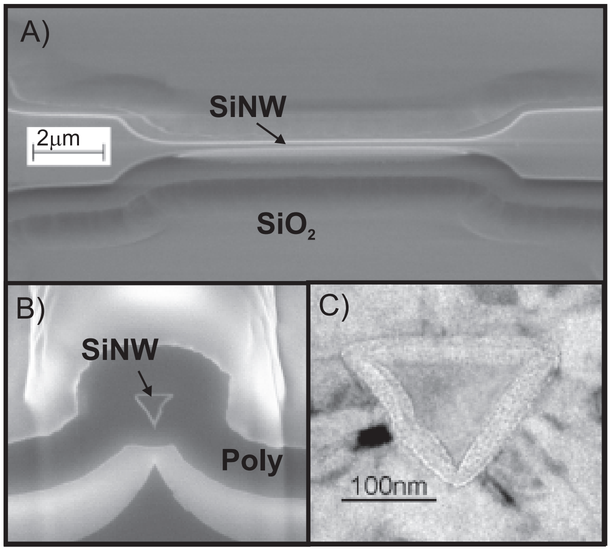


Figure 3: A) Suspended silicon nanowire, substrate is covered by LTO. B) Focused ion beam (FIB) image of a cross-secion of the final device. C) Transmission electron microscopy (TEM) image of a device channel cross-section.

4. MULTI-VALUED LOGIC ADDRESSING

In this section, we generalize the notion of encoding to multiple-valued bits by first defining some basic relations needed to identify possible codes. The kinds of multiple-valued encodings that can be built are then presented in the next section. Some basic semantic used in encoding theory is generalized from the binary definitions stated in [26] to the multiple-valued logic used in this paper. The matching between a code and its pattern corresponds here to conduction, whereas in [26] it refers to a non-conducting state. Algebraic operations are performed as defined in the ring of integers.

Definition 1. A multiple-valued pattern **a**, or simply a pattern **a**, is a suite of M digits a_i, in the n-valued base $\mathbb{B}$; *i.e.*, $\mathbf{a} = (a_0, \dots, a_{M-1}) \in \mathbb{B}^M, \mathbb{B} = \{0 \dots n-1\}$.

A pattern represents a serial connection of M GAA FETs in the SiNWs core; each digit a_i of the code word represents a threshold voltage $V_{\mathrm{T},i}$, with the convention $a_i < a_j \Leftrightarrow V_{\mathrm{T},i} < V_{\mathrm{T},j} \forall i, j = 0 \dots M-1$. An analogue equivalence holds for $a_i = a_j$ and consequently for $a_i > a_j$.

Definition 2. A multiple-valued code word **c**, or simply a code word **c**, is, similarly to a pattern, a suite of M digits c_i, in the n-valued base $\mathbb{B} = \{0, \dots, n-1\}$; *i.e.*, $\mathbf{c} = (c_0, \dots, c_{M-1}) \in \mathbb{B}^M$.

A code word represents the suite of applied voltages V_{A} at the M mesowires. These are defined such that every $V_{\mathrm{A},i}$ is slightly higher than $V_{\mathrm{T},i}$, and lower than $V_{\mathrm{T},i+1}$. Hence, a similar convention holds for the order of $V_{\mathrm{A},i}$ with respect to that of c_i.

Definition 3. A complement of digit x_i in a code word or pattern **x** is defined as: $\texttt{NOT}(x_i) = \overline{x_i} = (n-1) - x_i$. The operator NOT can be generalized to the vector $\mathbf{x}$, acting on each component as defined above. Notice that $\mathrm{NOT}(\mathrm{NOT}(\mathbf{x})) = \mathbf{x}$.

Definition 4. A pattern $\mathbf{a}$ is covered by a code word $\mathbf{c}$ if and only if the following relation holds: $\forall i = 0 \dots M-1, c_i \geq a_i$. By using the sigmoid function ($\sigma(x) = 0 \Leftrightarrow x \leq 0, \sigma(x) = 1$ elsewhere) generalized to vectors: $\sigma(\mathbf{x}) = (\sigma(x_0), \dots, \sigma(x_{M-1}))$, the definition above becomes: $\mathbf{a}$ is covered by $\mathbf{c} \Leftrightarrow \|\sigma(\mathbf{a} - \mathbf{c})\| = 0$. Alternatively, we can define the order relations on vectors $\mathbf{c}$ and $\mathbf{a}$: $\mathbf{c} < \mathbf{a} \Leftrightarrow \forall i, c_i < a_i$ and $\mathbf{c} > \mathbf{a} \Leftrightarrow \forall i, c_i > a_i$. The relation becomes an inequality if exists i such that $c_i = a_i$. Then, a pattern $\mathbf{a}$ is covered by a code word $\mathbf{c}$ if and only if $\mathbf{a} \leq \mathbf{c}$.

Under this assumption, the gate voltage applied at every GAA FETs is larger than its threshold voltages and turns it on. Therefore, the NWs having a pattern covered by the code word correponding to the applied voltages are conducting.

Definition 5. A code word $\mathbf{c}^a$ implies the code word $\mathbf{c}^b$ if and only if $\|\sigma(\mathbf{c}^b - \mathbf{c}^a)\| = 0$. We note this as follows: $\mathbf{c}^a \Rightarrow \mathbf{c}^b$

This means that if a NW with the pattern corresponding to $\mathbf{c}^a$ is covered by a code word $\mathbf{c}^*$, then the NW with the pattern corresponding to $\mathbf{c}^b$ is also covered by the same code word $\mathbf{c}^*$. Applying the voltage suite $\mathbf{c}^*$ will result in turning on the NWs with either pattern corresponding to $\mathbf{c}^a$ or $\mathbf{c}^b$.

Definition 6. The code words $\mathbf{c}^a$ and $\mathbf{c}^b$ are independently covered if and only if $\mathbf{c}^a$ does not imply $\mathbf{c}^b$ and $\mathbf{c}^b$ does not imply $\mathbf{c}^a$.

This definition means that there exists a voltage suite that turns on the NW with the pattern corresponding to $\mathbf{c}^a$ but not that with the pattern corresponding to $\mathbf{c}^b$; and reciprocally, there exists a second voltage pattern that turns on the NW with the pattern corresponding to $\mathbf{c}^b$ but not that with the pattern corresponding to $\mathbf{c}^a$.

Definition 7. The code word $\mathbf{c}^a$ belonging to the set Ω is addressable if and only if it does not imply any other code in $\Omega \backslash \{\mathbf{c}^a\}$. We define the set Ω to be addressable if and only if every code word in Ω is addressable.

If a NW has a pattern corresponding to an addressable code word, then there exists a voltage suite that activates only this NW and no other NW having its pattern in the set Ω.

PROPOSITION 1. *A set Ω of code words is addressable if and only if every code word in Ω is independently covered with respect to any other code word in Ω.*

PROOF. This follows directly from Def. 6 and 7. □

Consequently, an admissible set of applied voltages that uniquely address each NW corresponds to the set of code words Ω that independently covers every pattern in A. This set of patterns can be simply taken as Ω itself, if Ω is addressable.

5. CODE CONSTRUCTION

5.1 Hot Encoding

$\mathbf{k}$ being an n-dimensional vector $(k_0, \dots, k_{n-1})$, such that $\sum_i k_i = M$, the multi-valued $(\mathbf{k}, M)$-hot encoding is defined as the set of all code words having the length M such that each k_i represents the occurence of the digit $i, i = 0, \dots, n-1$. For instance, if $\mathbf{k} = (4, 3, 1)$, $M = 8$ and $n = 3$, then 00001112 and 00210110 belong to the considered $(\mathbf{k}, M)$-hot code space.

PROPOSITION 2. *The code space defined by a multi-valued $(\boldsymbol{k}, M)$-hot encoding is addressable.*

PROOF. Consider two code words c^a and c^b in code space defined by the $(\mathbf{k}, M)$-hot encoding. Both codes are identical except at P different digits lying at the positions $p_0, \dots, p_{P-1}$. c^b is obtained by a permutation of $\{\mathbf{c}^a_{p_0}, \dots, \mathbf{c}^a_{p_{P-1}}\}$. Hence, there is at least one position p_i for which holds $\mathbf{c}^a_{p_i} > \mathbf{c}^b_{p_i}$ and at least one position p_j for which holds $\mathbf{c}^a_{p_j} < \mathbf{c}^b_{p_j}$. This proves that $\|\sigma(\mathbf{c}^b - \mathbf{c}^a)\| \neq 0$ and $\|\sigma(\mathbf{c}^a - \mathbf{c}^b)\| \neq 0$ and that every two code words are independently covered. Then, Proposition 1 states that the whole code space is addressable. □

PROPOSITION 3. *The size of the code space defined by a multi-valued $(\boldsymbol{k}, M)$-hot encoding is maximal for $k_i = M/n \; \forall \; i = 0, \dots, (n-1)$. The size of the maximal-sized space is asymptotically proportional to $\propto n^M / M^{(n-1)/2}$ for a given n.*

PROOF. The number of code words is given by $M!/(\prod k_i!)$. This is a symmetric function of $(k_0, \dots, k_{n-1})$. Maximizing this function for a real-valued vector $\mathbf{k}$ can be easily carried out by replacing $k_i!$ by the Gamma function $\Gamma(k_i + 1)$, which gives the claimed statement. For a maximal-sized space, the Stirling formula yields for a large M: $N = n^M \cdot (2\pi \cdot M)^{1/2} / (2\pi \cdot M/n)^{n/2}$. □

In the rest of the paper, we implicitly mean the $(\mathbf{k}, M)$-hot code with the maximal-sized space when we simply talk about the $(\mathbf{k}, M)$-hot code.

5.2 N-ary Reflexive Code

It is clear that an n-ary tree code is not addressable since some code words will imply some others. For instance, if n=3, then the code word 222 implies all other code words (000 . . . 221). It is possible to prevent the inclusive character of the n-ary tree code by attaching the complement of the code word (*i.e.*, 222 becomes 222000). The as-constructed code is the *N-ary Reflexive Code (NRC)*.

PROPOSITION 4. *The code space defined by the NRC is addressable.*

PROOF. The first (non-reflected) halves of any two code words $\mathbf{c}^a$ and $\mathbf{c}^b$ having the total length M (M is even) differ by at least one digit at say position i ($i = 0 \dots (M/2-1)$). Let $\mathbf{c}^a$ be the code word such that $\mathbf{c}^a_i < \mathbf{c}^b_i$. The reflection implies that $\mathbf{c}^a_{M-1-i} > \mathbf{c}^b_{M-1-i}$. This proves that $\|\sigma(\mathbf{c}^b - \mathbf{c}^a)\| \neq 0$ and $\|\sigma(\mathbf{c}^a - \mathbf{c}^b)\| \neq 0$ and that $\mathbf{c}^a$ and $\mathbf{c}^b$ are independently covered. Then, Proposition 1 states that the whole code space is addressable. □

In a similar way, the reflection principle works for any other code (*e.g.*, Hamming code), making the whole code space addressable. However, in return it doubles the code length. Although the binary hot code is denser than the binary reflexive code, this statement holds for the multi-valued logic only if the codes are defect-free. This aspect is analyzed in the following sections.

6. DEFECT MODELS

The control of the SiNWs fabricated with our technology described in Section 3 is based on the modulation of the threshold voltage of the controlling GAA FETs. The encoding schemes proposed in Section 5 imposes a distribution of the applied control voltages between the successive threshold voltages (see Fig. 4). The main issue with the threshold voltage is its variability and process-dependency.

If $V_{\mathrm{T},i}$ varies within a small range close to its mean value, then the pattern does not change, since the NW still conducts under the same conditions. Under certain conditions, the pattern undergoes defects and can be either *i)* not covered by any valid code word, in which case the NW cannot be discovered and the pattern is useless; or *ii)* covered by at least one valid code word. In the second case, if two patterns or more are covered by the same code word, then this code word cannot be used because more than one NW would have the same address. Thus, in the second case, the pattern is only useful if at least one code word covering it covers no other pattern, insuring that the covered pattern can be addressable.

Assuming that in average every code word covers ν patterns when errors happen, let p_{I} be the probability that a pattern remains

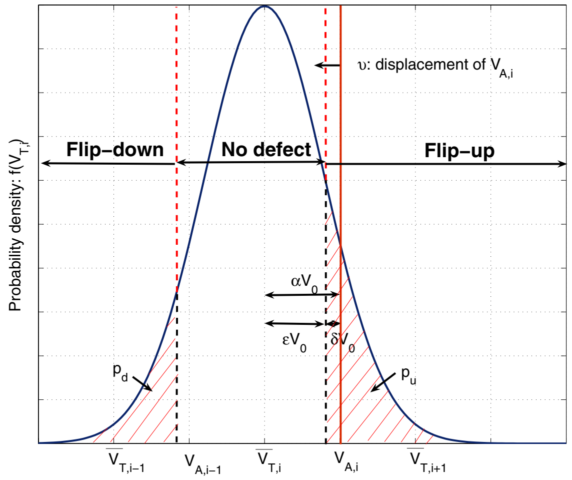


Figure 4: Coding defects induced by V_T variability

covered, and p_{U} the probability that a code word covers a unique pattern ($\overline{p}_{\mathrm{U}} = 1 - p_{\mathrm{U}}$). Let $|\Omega|$ be the original size of the code space and $|\Omega'|$ the size after errors happen. Then:

$$|\Omega'| = |\Omega| \cdot (1 - p_{\mathrm{I}})(1 - \overline{p}_{\mathrm{U}}^{\nu})$$

In order to assess $|\Omega'|$, we first model single digit errors, and then burst errors. Thus, we analytically derive p_{I} and p_{U}, and ν is estimated as a fit parameter from Monte-Carlo simulations.

6.1 Single Digit Errors

If the variability of $V_{\mathrm{T},i}$ is high or the spacing between two successive $V_{\mathrm{T},i}$'s is low due to the large number of doping levels, then $V_{\mathrm{T},i}$ may exceed $V_{\mathrm{A},i} - \delta \cdot V_0$; V_0 being a given scaling voltage. While V_{T} increases, the sensed current while a_i is applied on the digit c_i decreases ($a_i = c_i$) and the sensed current while $a_i + 1$ is applied on the same digit increases. We define a threshold $V_{\mathrm{X},i} = V_{\mathrm{A},i} - \delta \cdot V_0$ where the sensed current for a_i decreases by a factor q from its value at $\overline{V}_{\mathrm{T},i}$. Since the current in the saturation region of the GAA FET is proportional to $(V_{\mathrm{A},i} - V_{\mathrm{T}})^2$, where V_{T} is the actual GAA FET threshold voltage, then the following condition on V_{X} must hold: $(V_{\mathrm{A},i} - \overline{V}_{\mathrm{T},i})^2/(V_{\mathrm{A},i} - V_{\mathrm{X},i})^2 = q$; which gives: $\delta = (\alpha + \upsilon)/\sqrt{q}$. When $V_{\mathrm{T},i}$ exceeds $V_{\mathrm{X},i}$, the digit a_i acts as $a_i + 1$; its address becomes $c_i + 1$ and we call this case the *flip-up defect*.

Now, consider the case when $V_{\mathrm{T},i}$ falls below $V_{\mathrm{A},i-1} - \delta \cdot V_0$, then the current flowing while $a_i - 1$ is applied is not ~ 0 anymore, and always greater than q times the current flowing while $c_i - 2$ is applied. Then, a_i is implied by c_i and $c_i - 1$ but not by $c_i - 2$; its address is $c_i - 1$ which means that a_i acts as $a_i - 1$; this case is called the *flip-down defect*. When $V_{\mathrm{T},i}$ falls within the range between the threshold values for flip-up and flip-down defects, the digit is correctly interpreted. We notice that the flip-down error never happens at digits having the smallest value, 0, since the corresponding $\overline{V}_{\mathrm{T},i}$ is by definition smaller than the smallest $V_{\mathrm{A},i}$ available. For the same reason, the flip-up error never happens at the digits having the biggest value, $n - 1$. In order to study the size of the addressable code space, we consider flip-up and -down errors in the code space instead of flip-up and -down defects at the nanowires, since both considerations are equivalent.

6.2 Burst Errors in the k-Hot Code Space

Ω refers in the following to the code space of the maximal-sized multi-valued $(\mathbf{k}, M)$ hot encoding in the base $\mathbb{B} = \{0, \dots, n-1\}$ with $\mathbf{k} = (k, \dots, k)$ and $M = k \cdot n$. We consider a code word $\mathbf{c}$ in Ω undergoing a series of single digit errors. The burst error is described by the following vector $\mathbf{d} = (\mathbf{d}_0, \dots, \mathbf{d}_{n-1})$, where each $\mathbf{d}_i$ represents a pair of integers (d_i^u, d_i^d) expressing the number of flip-ups and flip-downs occurring in each digit group having the value i. Since no flip-down occurs at digits with value 0 and no flip-up occurs at digits with value $(n-1)$, then we impose $d_0^d = 0$ and $d_{(n-1)}^u = 0$. Because the number of digits having the same value is by definition k, it must hold for every d_i: $0 \leqslant d_i^u + d_i^d \leqslant k$.Then, we distinguish two types of error bursts $\mathbf{d}$:

- Burst errors of type I: $\exists i \in \{0, \dots, i-2\}/d_i^u > d_{i+1}^d$.
- Burst errors of type II: $\forall i = 0, \dots, (i-2) : d_i^u \leq d_{i+1}^d$.

Consider the case that a code word in Ω is transformed into a code word $\mathbf{c}^*$ by a burst error $\mathbf{d}$ of type I. We denote by i the smallest position in $\mathbf{d}$ at which holds $d_i^u > d_{i+1}^d$. The number of digits in $\mathbf{c}^*$ whose value is $\geq (i+1)$ becomes larger than the permitted number in any code word $\mathbf{c}^b$ in Ω, namely $k \cdot (n - i - 1)$. No code word would imply every digit of $\mathbf{c}^*$ with a value $\geq (i+1)$. Consequently, the NW with the pattern equal to the code word $\mathbf{c}^*$ is not addressed by any permitted code word. The probability of defect type I is given by Algorithm 1.

```
Algorithm 1 $p_{\mathrm{I}} = \mathtt{PA}(x^{\mathrm{u}}, x^{\mathrm{d}}, i, p)$
Construct $D$ = all possible defects in subtree
for all $\mathbf{y} = (y^{\mathrm{u}}, y^{\mathrm{d}}) \in D$ do
    $\pi_{\mathrm{sum}} \leftarrow 0$
    $p \leftarrow p \cdot k!/(y^{\mathrm{d}}! \cdot y^{\mathrm{u}}!) \cdot p_{\mathrm{u}}^{y^{\mathrm{u}}} \cdot p_{\mathrm{d}}^{y^{\mathrm{d}}}$
    if $i < n - 1$ then
        $\pi \leftarrow (y^d \geq x^u) \cdot \mathtt{PA}(y^u, y^d, i+1, p)$
    else
        $\pi \leftarrow (y^d \geq x^u) \cdot p$
    end if
    $\pi_{\mathrm{sum}} \leftarrow \pi_{\mathrm{sum}} + \pi$
end for
return $\pi_{\mathrm{sum}}$
```

Now, let the code word $\mathbf{c}^a$ in Ω be transformed into $\mathbf{c}^*$ by a burst error $\mathbf{d}$ of type II. The number of digits in $\mathbf{c}^*$ having the value i is always larger than the number of digits having the value $(i+1)$. It is possible to construct one or more code words $\mathbf{c}^b$ in Ω implying $\mathbf{c}^*$. An intuitive way consists in starting with the smallest digit value 0, and filling the digits of $\mathbf{c}^b$ by 0 with respect to the positions hold by the value 0 in the digits of $\mathbf{c}^*$. The procedure is repeated iteratively on next digit values until all digits of $\mathbf{c}^b$ are allocated (Algorithm 2). In Algorithm 2 we use the following notations: the number of digits having the value i in $\mathbf{c}^*$ is l_i. Their respective positions are $p_0^i, \dots, p_{l_i-1}^i$. The definition of the defect pattern $\mathbf{d}$ yields: $l_i = k - d_i^u + d_{i+1}^d + d_{i-1}^u \forall i$, with the convention $d_{-1}^u = 0$ for $i = 0$ and $d_{i+1}^d = 0$ for $i = n - 1$.

```
Algorithm 2 Construct all $\mathbf{c}^b$
$\nu_0 = k$
for $i = 0$ to $n - 1$ do
    Chose $\nu_i$ elements among $S_1 = \{p_{i_0}^i, \dots, p_{l_i}^i\}$ labeled $S_2 = \{q_0, \dots, q_{\nu_i - 1}\}$
    Allocate $i$ to the digits of $\mathbf{c}^b$ at positions $S_2$
    Allocate $i + 1$ to the digits of $\mathbf{c}^b$ at positions $S_1 \setminus S_2$
    $\nu_{i+1} = k - l_i + \nu_i$
    $i \leftarrow i + 1$
end for
```

Each choice of ν_i elements among l_i possible values gives many possible choices for $\mathbf{c}^b$. This proves that it is possible to find more than one single code word implying the incorrect code word $\mathbf{c}^*$.

Now, consider the event that a code word covers a unique pattern when defects happen. All patterns that can undergo a burst error and become covered by this code word are derived from it by a transformation $\mathfrak{T} = (\mathbf{t}^0, \ldots, \mathbf{t}^{n-1})$, where $\mathbf{t}^i \in \{0, \ldots, k\}^n$. t^i_{i+1} is the number of flip-ups at V_{T} level i and t^i_j $(j < i)$ is the number of order-j flip-downs from V_{T} level i to V_{T} level j. Algorithm 3 determins all valid transformations, namely, those mapping a code space onto itself; S being the set of all (p_i, μ_i) representing respectively the probabilies of all possible transformations $\mathfrak{T}$ and the number of their equivalent occurences. Then:

$$p_{\mathrm{U}} = \sum_{i=1,\ldots,|S|} \mu_i \cdot p_i/(1-p_i) \times \prod_{i=1,\ldots,|S|} (1-p_i)^{\mu_i}$$

Algorithm 3 $S = \mathtt{PB}(\mathfrak{T}, i)$

```
Construct $E$ = all possible transformations $\mathbf{t}^i$ for $V_{\mathrm{T}}$ level $i$
for all $\mathbf{t}^i \in E$ do
    Update $\mathfrak{T}$ with $\mathbf{t}^i$
    if $i < n - 1$ then
        $S \leftarrow S \cup \mathtt{PB}(\mathfrak{T}, i+1)$
    else
        if $\mathfrak{T}$ is a valid transformation then
            $u_i \leftarrow t^i_{i+1} \; \forall i$
            $d_i \leftarrow \sum_{j<i} t^i_j \; \forall i$
            $p \leftarrow \prod_{i=0}^{n-1} p_{\mathrm{d}}^{u_i} \cdot (1-p_{\mathrm{u}})^{k-u_i-d_i}$
            $\mu \leftarrow k!/u_i!/(k-u_i-d_i)!/\prod_{j<i} t^i_j!$
            $S \leftarrow \{(p, \mu)\}$
        end if
    end if
end for
return $S$
```

6.3 Burst Errors in the NRC Space

In the following Ω refers to an arbitrary NRC space with the length M (M is even) and the base size n. We define a flip-up defect at digit c_i in the code word $\mathbf{c}$ to be canceled when a flip-down defect occurs at digit c_{M-1-i}. A canceled flip-down defect is defined in a complementary way. We distinguish two types of error bursts:

- Burst errors of type I: The code word undergoes at least one non-canceled flip-up.
- Burst errors of type II: The code word only experiments flip-downs and/or canceled flip-ups.

Let the code word $\mathbf{c}^a$ in Ω be transformed into $\mathbf{c}^*$ by a burst error of type I. We denote by i one of the positions at which a non-canceled flip-up defect occurs. Then, $c^*_i = c^a_i + 1$ and $c^*_{M-1-i} \geq c^a_{M-1-i}$. Any code word $\mathbf{c}^0$ which would imply the pattern corresponding to $\mathbf{c}^*$ would verify $c^0_i \geq c^*_i = c^a_i + 1$. Then, $c^0_{M-1-i} = \mathtt{NOT}(c^0_i) = n - 1 - c^0_i \leq n - 1 - c^a_i - 1 = c_{M-1-i} - 1 < c^*_{M-1-i}$ and $\|\sigma(\mathbf{c}^* - \mathbf{c}^0)\| \neq 0$. Thus, there exists no code word in Ω that would cover the pattern corresponding to $\mathbf{c}^*$. The size of the addressable space is reduced by the number of code words undergoing the burst error of type I.

Now, let the code word $\mathbf{c}^a$ in Ω be transformed into $\mathbf{c}^*$ by a burst error d of type II. It is possible to find more than one code word $\mathbf{c}^0$ which covers the pattern correponding to $\mathbf{c}^*$. We construct first, say, the left half of $\mathbf{c}^0$, then the right half is obtained by complementing the left half. The construction rule is the following $(i = 0, \ldots, M/2 - 1)$:

- If c^a_i undergoes a non-canceled flip-down error, then c^0_i can be set to either $c^a_i - 1$ or c^a_i.
- If c^a_i undergoes a canceled flip-down error, then c^0_i is set to $c^a_i - 1$.
- If c^a_i undergoes a canceled flip-up error, then c^0_i is set to $c^a_i + 1$.
- If c^a_i has no error, then c^0_i is set to c^a_i.
- c^0_{M-1-i} is set to $n - 1 - c^0_i$

It is easy to verify that all patterns correponding to $\mathbf{c}^0$ constructed this way cover $\mathbf{c}^*$. In order to calculate the probability p_{I} of the burst error type I, we can observe the complementary event (burst error type II or no error in code word $\mathbf{c}^a$):

- If c^a_i undegoes no error, then $c^a_{M/2+i}$ must undergo a flip-down or no error with the likelyhood: $(1 - p_{\mathrm{u}} - p_{\mathrm{d}})(1 - p_{\mathrm{u}})$.
- If c^a_i unergoes a flip-down error, then $c^a_{M/2+i}$ can have any value with the likelyhood: p_{d}.
- If c^a_i undergoes a flip-up error, then $c^a_{M/2+i}$ must undergo a flip-down error with the likelyhood: $p_{\mathrm{u}} p_{\mathrm{d}}$.

If we neglegt the adjustment due to the fact that no flip-down happens at $i = 0$ and no flip-up at $i = n - 1$, we get:

$$p_{\mathrm{I}} = ((1 - p_{\mathrm{u}} - p_{\mathrm{d}})(1 - p_{\mathrm{u}}) + p_{\mathrm{d}} + p_{\mathrm{u}} p_{\mathrm{d}})^{M/2}$$

Now, we would like to calculate p_{U}. Any code word $\mathbf{c}^*$ that can be transformed through flip-up and flip-down errors to another word covered by $\mathbf{c}^0$ is situated in the neighborhood S of $\mathbf{c}^0$, defined as any code word verifying $c^*_i - c^0_i \in \{-1, 0, 1\} \; \forall i$. Let α be the number of digits in a half-word such that $c^*_i - c^0_i = 0$. Neglecting the adjustment due to V_{T} range borders, there are $\mu_\alpha = \binom{M/2}{\alpha} \cdot 2^{M/2-\alpha}$ possible code words fulfilling this condition. Everyone of them can be tranformed into $\mathbf{c}^0$ if the α digits in each half undergo no flip-up error, and each one of the $M/2 - \alpha$ digits for which holds $c^*_i - c^0_i = 1$ undergoes a flip-down error. The likelyhood of each valid transformation is $p_\alpha = (1 - p_{\mathrm{u}})^{2\alpha} \cdot p_{\mathrm{d}}^{M/2-\alpha}$. The likelyhood that only one code word in S is transformed into another word covered by $\mathbf{c}^0$ is given by:

$$p_{\mathrm{U}} = \sum_{i=0}^{M/2} \mu_i \cdot p_i/(1-p_i) \times \prod_{i=0}^{M/2} (1-p_i)^{\mu_i}$$

7. SIMULATION RESULTS

The defect models defined in Section 6 enable the assessment of the number of NWs that the decoder can address uniquely even with variable threshold voltages. Based on this estimate, we can evaluate the effective capacity of the crossbar memory, *i.e.*, the number of addressable bits. An analysis of the defects affecting the switches are beyond the scope of this paper, and the fabricated SiNWs scarcely suffer from breakages [25]. Thus, we can estimate the effective capacity C_{eff} in analog way to the memory yield in [8]: $C_{\mathrm{eff}} = (P^2_{\mathrm{contact}} \cdot |\Omega'|)^2$, with P_{contact} the probability that the NW ohmic contact is bad ($P_{\mathrm{contact}} = 0.95$ from [8]). The area can be estimated from the geometry as sketched in Fig. 2, and the process as explained in [25]. Furthermore, we used the following simulation parameters: q=100, $V_0 = V_{\mathbf{DD}} = 1$ V, $\alpha \sim 1/n$ since the difference between the highest and lowest $V_{\mathrm{T},i}$'s is limited by the constant V_{DD}.

In order to assess the variation of the addressable code space under variable V_{T}, we plotted seperately the uncovered part $|\Omega|_{\mathrm{un}} = p_{\mathrm{I}} \cdot |\Omega|$, the addressble part $|\Omega'|$ and the immune part in which no defects occur: $|\Omega|_{\mathrm{im}} = (1 - p_{\mathrm{u}} - p_{\mathrm{d}})^M \cdot |\Omega|$. The fit parameter ν was estimated with Monte-Carlo simulations. Fig. 5 shows the sizes of these subspaces for a ternary (3,14)-reflexive code depending on the 3σ-value of V_{T}. The monte-Carlo simulation confirms in the same figure the analytical results and gives the value 2.8 for the fit parameter ν. $|\Omega'|$ drops quickly when 3σ reaches 0.4V. At the

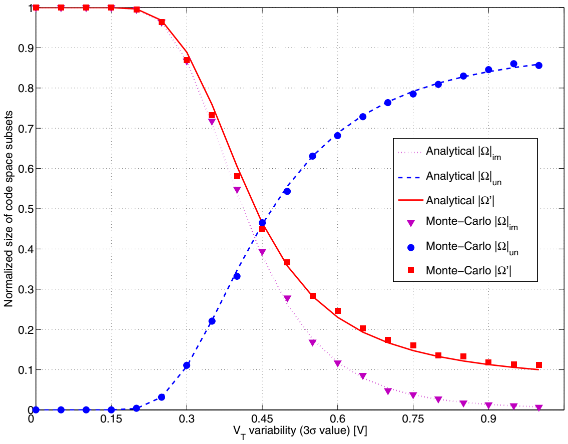


Figure 5: Dependency of different code space subsets on V_{T} variability

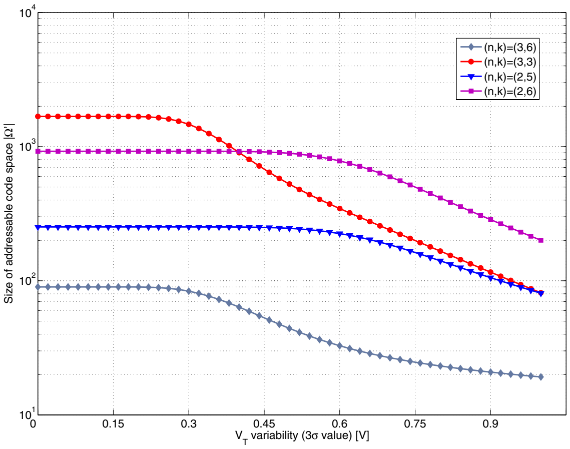


Figure 6: Number of addressable NWs for different hot codes

same time, more patterns become uncovered. Interestingly, there are more addressable than immune patterns, because some defective patterns can be randomly addressed. This tendency increases for unreliable technologies, and around 10% of the original code space size can be randomly addressed under extreme conditions. The behavior of hot codes is similar, except for large defect probabilities: under these conditions the size of the addressable space goes towards 0 because the construction of hot codes imposes more constraints than the NRC.

The sizing of the memory blocks (*i.e.*, the ohmic regions in Fig. 2) and the number of V_{T} are dependant. As a matter of fact, Fig. 6 shows that increasing the number of V_{T} has two opposite effects: On one hand, it enables the addressing of more wires with the same code length; on the other hand, it makes the transistors more vulnerable to defects and increases the number of lost codes. A typical trade-off situation is illustrated in Fig. 6 with the ternary (3,9) and binary (2,12) hot codes (with (n,k)=(3,3) and (2,6) respectively) yielding almost the same number of addressable NWs for 3σ around 0.4V. The first one saves area because it has shorter codes, whereas the second one is technologically easier to realize (only 2 different V_{T}). The use of the ternary decoder is recommended for reliable technologies (insuring less area and more codes), but when the technology becomes more unreliable, there is a trade-off between the area-saving and the easier fabrication process.

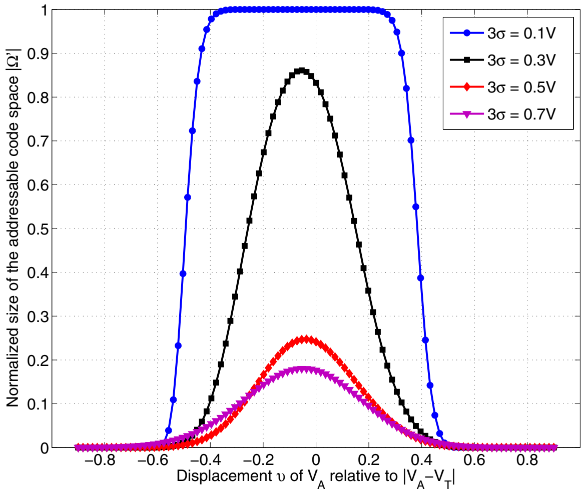
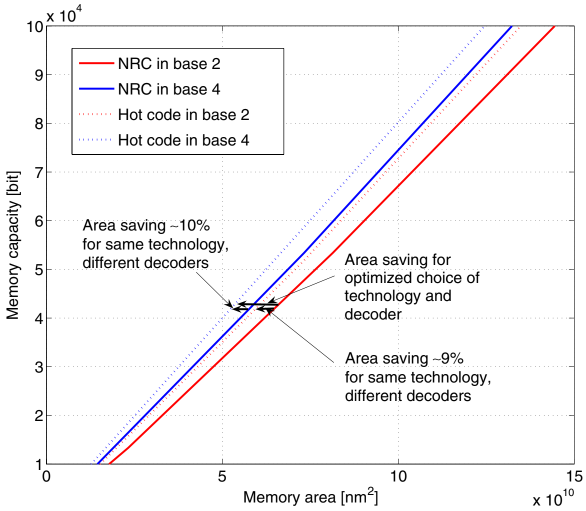


Figure 7: Impact of the value of V_{A} on the number of addressable NWs

We also investigated the impact of the placement of V_{A} between two successive $V_{\mathrm{T,i}}$. We postulated in an intuitive way that V_{A} has to be the median of $V_{\mathrm{T},i}$ and $V_{\mathrm{T},i+1}$. While modeling the defects (Section 6), we allowed V_{A} to translate by a small value υ. When υ increases (*i.e.*, V_{A} moves towards $V_{\mathrm{T},i}$), then the probability of a flip-up increases and that of a flip-down decreases. The opposite happens when V_{A} moves towards $V_{\mathrm{T},i+1}$. The normalized number of addressable nanowires has been plotted in Fig. 7. For unreliable devices with 3σ >0.1V, the optimal position of V_{A} is slightly shifted from the middle of $V_{\mathrm{T},i}$ and $V_{\mathrm{T},i+1}$ towards $V_{\mathrm{T},i+1}$ by a few tens of mV. While reliable devices show a plateau around the optimal value of υ and necessitate no accurate calibration of V_{A}, the circuit designer has to calibrate the applied voltages V_{A} in a precise way when the transistors are not reliable; otherwise a certain loss in the number of addressable nanowires has to be taken into account. This circuit level issue has to be considered for either hot or n-ary reflected codes (both codes showed a similar behavior). In our technology characterized by a 3σ-value below 100 mV, no calibration will be needed.

We explored the memory effective capacity/area design space and we performed a simulation of the design space of our technology (Fig. 8). The processes based on 2 and 4 V_{T} were considered for both hot and n-ary reflected codes. As Fig. 8 shows, the hot code generally reduces the decoder area, and consequently the memory area, because with the same code length, it is possible to address a larger code space. For instance, by using the same technology with 4 V_{T} to fabricate a 4 kb-memory , the NRC has an area overhead of ∼ 10% compared to the hot code. The uses of a simpler technology with 2 V_{T} implies an area overhead of ∼ 24% for the same memory size. The area saving can be performed at either the technology or system level. It is worth to notice that the area saving is more significant for small memory sizes (typically less than 0.1 Mb), because the memory area for large memories is dominated by the area of the programmable array. The programmable array can be split into smaller blocks defined by the size of the ohmic regions (Fig. 2) in order to reach the optimal size.

Since the placement of V_{A} between the successive V_{T}'s is a critical design aspect, we investigated its impact on the overall memory area/capacity design space. In Fig. 9 we assumed a less reliable technology with 3σ-value around 1.2V. If V_{A} is simply placed at the median of each two successive $V_{\mathrm{T},i}$ and no care is taken to place it at the optimal position (*i.e.* υ = -20 mV), then the memory has an area overhead of around 11%. On the other hand, even at the suboptimal position ($\upsilon = 0$), the area saving compared to the position corresponding to υ = 20 mV is 17% for a hot decoder and 27% for a n-ary reflected decoder. These remarks confirm the importance of accurately controlling V_{A} in crossbar memories.



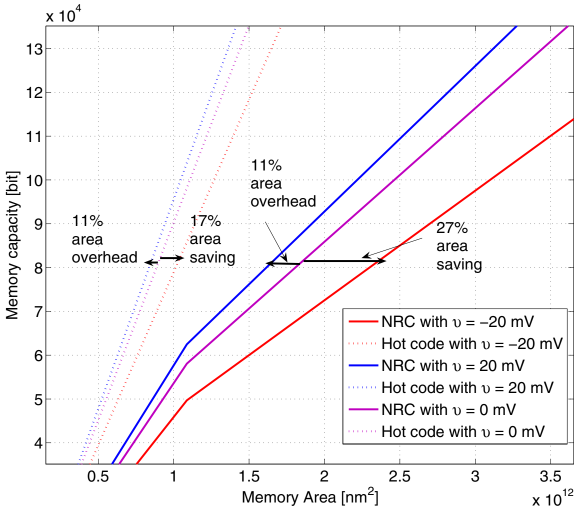


Figure 8: Memory area/capacity design space for different encoders

Figure 9: Memory area/capacity design space for different placements of V_A

8. CONCLUSIONS

In this work we studied multi-valued decoders for crossbar memories. We presented the memory architecture enabled by our technology, and the model of the decoder utilizing the available access devices. Through the formalism of the multi-valued addressing, we extended the hot and binary reflexive codes to multi-valued logic. We modeled the impact of defects driven by the V_T variability on the number of addressable NWs. The performed simulations showed that hot decoders with multiple V_T allow an area saving up to 24%. For highly unreliable technologies, NRC guarantees that at least 10% of the NWs recover. We also demonstrated that the accurate control of the addressing voltages increases the number of addressable wires and saves area up to 27%. In future work, we will extend the decoder modeling to other defects including the mask misalignment and we will investigate the impact of additional transitions of multi-valued logic on power consumption. On the technology side, we are currently processing a prototype of the memory array and the decoder.

9. REFERENCES

[1] Robert Beckman, Ezekiel Johnston-Halperin, Yi Luo, Jonathan E. Green, and James R. Heath. Bridging dimensions: Demultiplexing ultrahigh density nanowire circuits. *Science*, 310(5747):465–468, 2005.

[2] Michael Butts, A. DeHon, and Seth Copen Goldstein. Molecular electronics: Devices, systems and tools for gigagate, gigabit chips. In *Proceedings of the International Conference on Computer-Aided Design*, pages 433–440, 2002.

[3] G. F. Cerofolini. Realistic limits to computation. ii. the technological side. *Applied Physics A*, 86(1):31–42, 2007.

[4] Y. Cui, Z. Zhong, D. Wang, W. U. Wang, and C. M. Lieber. High performance silicon nanowire field effect transistor. *Nano Letters*, 3(2):149–152, 2003.

[5] A. DeHon. Array-based architecture for FET-based, nanoscale electronics. *IEEE Transactions on Nanotechnology*, 2(1):23–32, 2003.

[6] A. DeHon. Design of programmable interconnect for sublithographic programmable logic arrays. In *Proceedings of the International Symposium on Field-Programmable Gate Arrays (FPGA)*, pages 127–137, 2005.

[7] A. DeHon, S.C. Goldstein, P.J. Kuekes, and P. Lincoln. Nonphotolithographic nanoscale memory density prospects. *IEEE transactions on Nanotechnology*, 4(2):215–228, 2005.

[8] A. DeHon, P. Lincoln, and J.E. Savage. Stochastic assembly of sublithographic nanoscale interfaces. *IEEE Trans. on Nanotechnology*, 2(3):165–174, 2003.

[9] A. DeHon and Michael J. Wilson. Nanowire-based sublithographic programmable logic arrays. In *Proceedings of the International Symposium on Field-Programmable Gate Arrays (FPGA)*, pages 123–132, 2004.

[10] J. Kedzierski et al. Metal-gate FINFET and fully-depleted SOI devices using total gate silicidation. In *IEEE Electron Devices Meeting*, pages 247–250, 2002.

[11] K. Gopalakrishnan et al. The micro to nano addressing block. In *IEEE Electron Devices Meeting*, page 19.4, 2005.

[12] S. Monfray et al. 50 nm-gate all around (GAA)-silicon on nothing (SON)-devices: a simple way to co-integration of GAA transistors within bulk MOSFET process. In *Symposium on VLSI Technology*, pages 108–109, 2002.

[13] Seth Copen Goldstein and Mihai Budiu. Nanofabrics: spatial computing using molecular electronics. In *Proceedings of the 28th Annual International Symposium on Computer Architecture*, pages 178–189, 2001.

[14] M. S. Gudiksen, L. J. Lauhon, J. Wang, D. C. Smith, and C. M. Lieber. Growth of nanowire superlattice structures for nanoscale photonics and electronics. *Nature*, 415:617–620, 2002.

[15] T. Hogg, Yong Chen, and P.J. Kuekes. Assembling nanoscale circuits with randomized connections. *IEEE Transactions on Nanotechnology*, 5(2):110–122, 2006.

[16] J. D. Holmes, K. P. Johnston, R. C. Doty, and B. A. Korgel. Control of thickness and orientation of solution-grown Silicon nanowires. *Science*, 287(5457):1471–1473, 2000.

[17] Yu Huang, Xiangfeng Duan, Yi Cui, Lincoln J. Lauhon, Kyoung-Ha Kim, and Charles M. Lieber. Logic gates and computation from assembled nanowire building blocks. *Science*, 249:1313–1317, 2001.

[18] J. Kedzierski and J. Bokor. Fabrication of planar Silicon nanowires on Silicon-on-insulator using stress limited oxidation. *Journal of Vacuum Science and Technology B*, 15(6):2825–2828, 1997.

[19] S.-M. Koo, A. Fujiwara, J.-P. Han, E. M. Vogel, C. A. Richter, and J. E. Bonevich. Inversion current in Silicon nanowire field effect transistors. *Nano Letters*, 4(11):2197–2201, 2004.

[20] P. J. Kuekes and R. S. Williams. Demultiplexer for a molecular wire crossbar network (MWCN DEMUX), 2001.

[21] Chun Ning Lau, Duncan R. Stewart, R. Stanley Williams, and Marc Bockrath. Direct observation of nanoscale switching centers in metal/molecule/metal structures. *Nano Letters*, 4(4):569–572, 2004.

[22] L. J. Lauhon, M. S. Gudiksen, D. Wang, and C. M. Lieber. Epitaxial core-shell and core-multishell nanowire heterostructures. *Nature*, 420:57–61, 2002.

[23] Y. Luo, C. P. Collier, J. O. Jeppesen, K. A. Nielsen, E. DeIonno, G. Ho, J. Perkins, H.-R. Tseng, T. Yamamoto, J. F. Stoddart, and J. R. Heath. Two-dimensional molecular electronics circuits. *Journal of Chemical Physics and Physical Chemistry (ChemPhysChem)*, 3:519–525, 2002.

[24] N. A. Melosh, A. Boukai, F. Diana, B. Gerardot, A. Badolato, P. M. Petroff, and J. R. Heath. Ultrahigh-density nanowire lattices and circuits. *Science*, 300(5616):112–115, 2003.

[25] K. E. Moselund, D. Bouvet, L. Tschuor, V. Pot, P. Dainesi, C. Eggimann, N. Le Thomas, R. Houdré, and A. M. Ionescu. Cointegration of gate-all-around MOSFETs and local Silicon-on-insulator optical waveguides on bulk silicon. *IEEE Transactions on Nanotechnology*, 6(1):118–125, 2007.

[26] Eric Rachlin. Robust nanowire decoding, www.cs.brown.edu/publications/theses/masters/2006/eerac.pdf, 2006.

[27] T. Rueckes, K. Kim, E. Joselevich, G. Y. Tseng, C.-L. Cheung, and C. M. Lieber. Carbon nanotube based nonvolatile random access memory for molecular computing. *Science*, 289:94–97, 2000.

[28] J. E. Savage, E. Rachlin, A. DeHon, C. M. Lieber, and Y. Wu. Radial addressing of nanowires. *ACM Journal on Emerging Technologies in Computing Systems*, 2(2):129–154, 2006.

[29] D. R. Stewart, D. A. A. Ohlberg, P. A. Beck, Y. Chen, R. Stanley Williams, J. O. Jeppesen, K. A. Nielsen, and J. Fraser Stoddart. Molecule-independent electrical switching in Pt/organic monolayer/Ti devices. *Nano Letters*, 4(1):133–136, 2004.

[30] D. Whang, S. Jin, Y. Wu, and C. M. Lieber. Large-scale hierarchical organization of nanowire arrays for integrated nanosystems. *Nano Letters*, 3(9):1255–1259, 2003.

[31] H.-S.P. Wong, K.K. Chan, and Y. Taur. Self-aligned (top and bottom) double-gate MOSFET with a 25 nm thick silicon channel. In *IEEE Electron Devices Meeting*, pages 427–430, 1997.

[32] C. M. Lieber Y. Cui. Functional nanoscale electronic devices assembled using silicon nanowire building blocks. *Science*, 291(5505):851–853, 2001.