

CMOL: Devices, Circuits, and Architectures

Konstantin K. Likharev and Dmitri B. Strukov

Stony Brook University, Stony Brook, NY 11794, USA

Summary. This chapter is a brief review of the recent work on various aspects of the prospective hybrid semiconductor/nanowire/molecular (“CMOL”) integrated circuits. The basic idea of such circuits is to combine the advantages of the currently dominating CMOS technology (including its flexibility and high fabrication yield) with those of molecular devices with nanometer-scale footprint. Two-terminal molecular devices would be self-assembled on a pre-fabricated nanowire crossbar fabric, enabling very high function density at acceptable fabrication costs. Preliminary estimates show that the density of active devices in CMOL circuits may be as high as 10^{12} cm^{-2} and that they may provide an unparalleled information processing performance, up to 10^{20} operations per cm^2 per second, at manageable power consumption. However, CMOL technology imposes substantial requirements (most importantly, that of high defect tolerance) on circuit architectures. In the view of these restrictions, the most straightforward application of CMOL circuits is terabit-scale memories, in which powerful bad-bit-exclusion and error-correction techniques may be used to boost the defect tolerance. The implementation of Boolean logic circuits is more problematic, though our preliminary results for reconfigurable, uniform FPGA-like CMOL circuits look very encouraging. Finally, CMOL technology seems to be uniquely suitable for the implementation of the “CrossNet” family of neuromorphic networks for advanced information processing including, at least, pattern recognition and classification, and quite possibly much more intelligent tasks. We believe that these application prospects justify a large-scale research and development effort focused on the main challenge of the field, the high-yield self-assembly of molecular devices.

1 Introduction

The recent spectacular advances in molecular electronics (for reviews see, e.g., Refs. 1-3 and other chapters of this collection), and especially the experimental demonstration of molecular single-electron transistor by several groups [4]-[8], give hope for the practical introduction, within the next 10 to 20 years, of the first integrated circuits with active single- or few-molecule devices.

This long-expected breakthrough could not arrive more timely. Indeed, the recent results [9, 10] indicate that the current VLSI paradigm, based on a combination of lithographic patterning, CMOS circuits, and Boolean logic, can hardly be extended into a-few-nm region. The main reason is that at gate

length below 10 nm, the sensitivity of parameters (most importantly, the gate voltage threshold) of silicon field-effect transistors (MOSFETs) to inevitable fabrication spreads grows exponentially. As a result, the gate length should be controlled with a few-angstrom accuracy, far beyond even the long-term projections of the semiconductor industry [11]. Even if such accuracy could be technically implemented using sophisticated patterning technologies, this would send the fabrication facilities costs (growing exponentially even now) skyrocketing, and lead to the end of the Moore’s Law some time during the next decade.

The main alternative nanodevice concept, single-electronics [10, 12], offers some potential advantages over CMOS, including a broader choice of possible materials. Unfortunately, for room-temperature operation the minimum features of these devices (single-electron islands) should be below ~ 1 nm [12]. Since the relative accuracy of their definition has to be between 10 and 20%, the absolute fabrication accuracy should be of the order of 0.1 nm, again far too small for the current and realistically envisioned lithographic techniques.

This is why there is a rapidly growing consensus that the impending crisis of the microelectronics progress may be resolved only by a radical paradigm shift from the lithography-based fabrication to the “bottom-up” approach. In the latter approach, the smallest active devices should be formed in a special way ensuring their fundamental reproducibility. The most straightforward example of such device is a specially designed and chemically synthesized molecule comprising of a few hundreds of atoms, including the functional parts (e.g., acceptor groups working as single-electron islands and short fragments of non-conducting groups as tunnel junctions [4]-[8]), the groups enabling chemically-directed self-assembly of the molecule on prefabricated electrodes (e.g., thiol or isocyanide groups [1]-[8]), and very probably some additional groups ensuring sufficient rigidity and stability of the molecule at room temperature.

Unfortunately, integrated circuits consisting of molecular devices alone are hardly viable, because of limited device functionality. For example, the voltage gain of a 1-nm-scale transistor, based on any known physical effect (e.g., the field effect, quantum interference, or single-electron charging), can hardly exceed one, i.e. the level necessary for sustaining the operation of virtually any active analog or digital circuit.¹ This is why we believe that the

¹The very recent suggestion [13] to replace transistors with the so-called Goto pairs of two-terminal latching switches in crossbar circuits runs into several problems, most importantly the relation between the retention time and switching speed. In order to be useful for most electronics applications, the latches should be switched very fast (in a few picoseconds in order to compete with advanced MOSFETs), but retain their internal state for the time necessary to complete the calculation (ideally, for a few years, though several hours may be acceptable in some cases). This means that the change of the applied voltage by the factor of two (the difference between the fully selected and semi-selected crosspoints of a crossbar) should change the switching rate by at least 16 orders of magnitude. However, even the most favor-

only plausible way toward high-performance nanoelectronic circuits is to integrate molecular devices, and the connecting nanowires, with CMOS circuits whose (relatively large) field-effect transistors would provide the necessary additional functionality, in particular high voltage gain.

Recently, several specific proposals of such circuits were published and several groups made initial steps toward the experimental implementation of semiconductor-molecular hybrids [15]-[17]. (Detailed reviews of this, and some other previous work on molecular electronics circuitry may be found in Refs. 18, 19.) The goal of this chapter is to review the recent work in one promising direction toward hybrid semiconductor-molecular electronics, the so-called CMOL approach. We will start from a discussion (in Sec. 2 and 3) of the hardware aspects of this concept. The remainder of the chapter is devoted to a discussion of possible architectures and applications of the CMOL circuits. Section 4 describes the results of our recent analysis of their most straightforward application, digital memories. In Sec. 5 we discuss the situation with possible CMOL logic circuits. One more promising direction of CMOL work, toward mixed-signal neuromorphic networks, is reviewed in Sec. 6. Finally, in the Conclusion (Sec. 7) we briefly summarize the results of our discussion.

2 Devices

The first critical issue in the development of semiconductor/molecular hybrids is making a proper choice in the trade-off between molecule simplicity and functionality. On one hand, simple molecules (like the octanedithiols [20]), which may provide nonlinear but monotonic $I - V$ curves with no hysteresis (i.e. no internal memory), are hardly sufficient for highly functional integrated circuits, because a semiconductor memory subsystem would hardly be able to store enough data for processing by more numerous molecular devices. On the other hand, a very complex molecule (like a long DNA strand [21]) may have numerous configurations that can be, as a matter of principle, used for information storage. However, such molecules are typically very “soft”, so that thermal fluctuations at room temperature (that is probably the only option for broad electronics applications) may lead to uncontrollable switches between their internal states, making reliable information storage and usage difficult, if not totally impossible.

This is why we believe that relatively short and rigid molecules (with the number of atoms of the order of one hundred), having two (or a few) metastable internal states, are probably the best choice for the initial development of molecular electronics. Our own best choice is the binary “latching

able physical process we are aware of (the quantum-mechanical tunneling through high-quality dielectric layers like the thermally-grown SiO_2) may only produce, at these conditions, the rate changes below 10 orders of magnitude, even if uncomfortably high voltages of the order of 12 V are used [14].

switch”, i.e. a two-terminal, bistable device with $I - V$ curves of the type shown in Fig. 1a.² Such switch may be readily implemented, for example, as a combination of two single-electron devices: a “transistor” and a “trap” (Fig. 1b).³ If the applied drain-to-source voltage $V = V_d - V_s$ is low, the trap island in equilibrium has no extra electrons ($n = 0$), and its net electric charge $Q = -ne$ is zero. As a result, the transistor is in the virtually closed (OFF) state, and source and drain are essentially disconnected. If V is increased beyond a certain threshold value V_+ , its electrostatic effect on the trap island potential (via capacitance C_s) leads to tunneling of an additional electron into the trap island: $n \rightarrow 1$. This change of trap charge affects, through the coupling capacitance C_c , the potential of the transistor island, and suppresses the Coulomb blockade threshold to a value well below V_+ . As a result, the transistor, whose tunnel barriers should be thinner than that of the trap, is turned into ON state in which the device connects the source and drain with a finite resistance R_0 . (Thus, the trap island plays the role similar to that of the floating gate in the usual nonvolatile semiconductor memories [14].) If the applied voltage stays above V_+ , this connected state is sustained indefinitely; however, if V remains low for a long time, the thermal fluctuations will eventually kick the trapped electron out, and the transistor will get closed, disconnecting the electrodes. This ON \rightarrow OFF switching may be forced to happen much faster by making the applied voltage V sufficiently negative, $V \approx V_-$.⁴

Figure 1c shows a possible molecular implementation of the device shown in Fig. 1b. Here two different diimide acceptor groups play the role of single-electron islands, while short oligo-ethynylene-phenylene (OPE) chains are used as tunnel barriers. The chains are terminated by isocyanide-group “clamps” (“alligator clips”) that should enable self-assembly of the molecule across a gap between two metallic electrodes.

This immediately brings us to possibly the most important challenge faced by the development of VLSI molecular electronics, the reproducible self-assembly of the molecules on prefabricated electrodes. To the best of our knowledge, no group has yet succeeded to achieve acceptable yield of such process even for single devices. Moreover, even successful (conducting) samples differ by the current scale and sometimes the general shape of their $I - V$ curves. This is not entirely surprising, because the used clamp groups (like those shown in Fig. 1c) can hardly ensure a unique position of the molecule relative to the electrodes, and hence a unique structure and transport prop-

²Multi-terminal devices would be immeasurably more complex for the chemically-directed self-assembly.

³Low-temperature prototypes of this device have been implemented and successfully tested experimentally, with electron trapping times beyond 12 hours [22].

⁴A virtually similar functionality may be achieved using configurational changes of specially selected molecules [13, 23, 24], however, such molecules are rather complex, and their switching may be too slow for most applications.

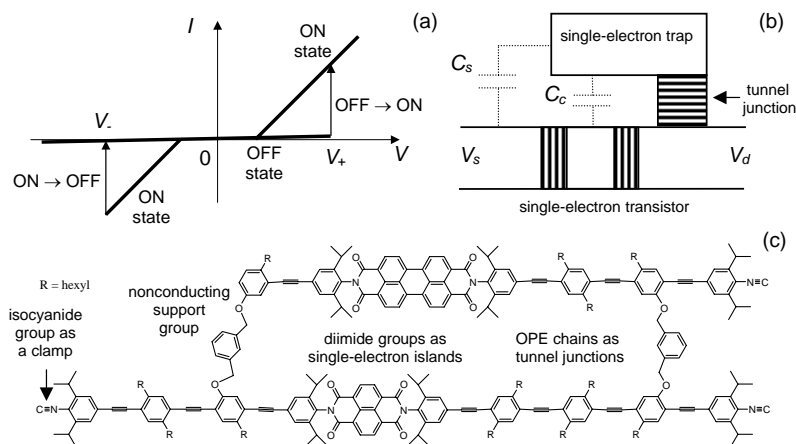


Fig. 1. Two-terminal latching switch: (a) $I - V$ curve (schematically), (b) single-electron device schematics [25], and (c) a possible molecular implementation of the device (courtesy A. Mayr).

erties of molecular-to-electrode interfaces.

One possible way toward high self-assembly yield is the chemical synthesis of molecules including relatively large “floating electrodes” (large acceptor groups or metallic clusters - see Fig. 2). If the characteristic internal resistance R_0 of such a molecule is much higher than the range of possible values of molecule/electrode resistances R_i , and the floating electrode capacitances are much higher than those of the internal single-electron islands, then the transport through the system will be determined by R_0 and hence be reproducible.⁵

Another possible way toward high yield is to form a self-assembled monolayer (SAM) on the surface of the lower nanowire level, and only then deposit and pattern the top layer. Such approach has already given rather reproducible results (in the nanopore geometry) for simple, short molecules [20]. The apparent problem here is that each crosspoint would have several parallel devices even if the nanowire width is scaled down to a few nanometers, and this number may not be somewhat different from crosspoint to crosspoint. However, all circuits discussed below can function properly even in this case.

The potentially enormous density of molecular devices can hardly be used

⁵Actually, this approach to interfaces is very much parallel to that accepted de facto in semiconductor electronics. Indeed, despite decades of research, properties of silicon-to-metal interfaces (in particular, the Fermi level pinning due to surface traps) are still neither completely understood nor fully predictable. This is why in most semiconductor circuit technologies, metal-semiconductor junctions are used only as passive Ohmic contacts, while active devices are built around much better explored $p - n$ junctions formed inside the semiconductor.

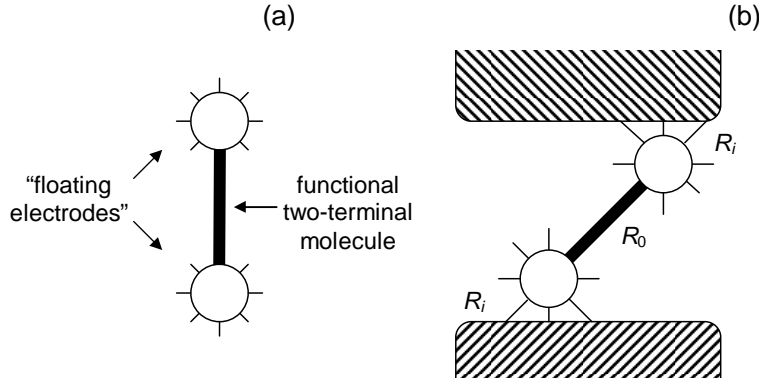


Fig. 2. A molecule with “floating electrodes” (a) before and (b) after its self-assembly on “real electrodes”, e.g., metallic nanowires (schematically).

without individual contacts to each of them. This is why the fabrication of wires with nanometer-scale cross-section is another central problem of molecular microelectronics. The currently available photolithography methods, and even their rationally envisioned extensions, will hardly be able to provide such resolution. Several alternative techniques, like the direct e-beam writing and scanning-probe manipulation can provide a nm-scale resolution, but their throughput is forbiddingly low for VLSI fabrication. Self-growing nanometer-scale-wide structures like carbon nanotubes or semiconductor nanowires can hardly be used to solve the wiring problem, mostly because these structures (in contrast with the specially synthesized molecules that have been discussed above) do not have means for reliable placement on the lower integrated circuit layers with the necessary (a-few-nm) accuracy. Fortunately, there are several new patterning methods, notably nanoimprint [26] and interference lithography [27], which may provide very high resolution (in future, down to a few nanometers) than the standard photolithography.

3 Circuits

These novel patterning technologies cannot be used, however, for the fabrication of arbitrary integrated circuits, in particular because they lack adequate layer alignment accuracy. This means that the nanowire layers should not require precise alignment with each other and with the CMOS subsystem. While the former requirement may be readily satisfied by using the “crossbar” nanowire structure (i.e., two layers of similar wires perpendicular to those of the other layer), the solution of the latter problem (CMOS-to-nanowire interface) is much harder. In fact, the interface should enable the CMOS subsystem, with a relatively crude device pitch $2\beta F_{\text{CMOS}}$ (where $\beta \sim 1$ is the ratio of the CMOS cell size to the wiring period), to address each wire

separated from the next neighbors by a much smaller distance F_{nano} .

Several solutions to this problem, which had been suggested earlier, seem either unrealistic, or inefficient, or both. In particular, the interface based on statistical formation of semiconductor-nanowire field-effect transistors gated by CMOS wires [28, 29] can only provide a limited (address-decoding-type) connectivity. In addition, the resistivity of semiconductor nanowires would be too high for high-performance hybrid circuits. Even more importantly, the technology of ordering chemically synthesized semiconductor nanowires into highly ordered parallel arrays has not been developed, and the authors of this review are not aware of any promising idea that may allow such assembly.

A more interesting approach was discussed in Ref. 30 (see also Ref. 18). It is based on a cut of the ends of nanowires of a parallel-wire array, along a line that forms a small angle $\alpha = \arctan(F_{\text{nano}}/F_{\text{CMOS}})$ with the wire direction. As a result of the cut, the ends of adjacent nanowires stick out by distances (along the wire direction) differing by $2F_{\text{CMOS}}$, and may be contacted individually by the similarly cut CMOS wires. Unfortunately, the latter (CMOS) cut has to be precisely aligned with the former (nanowire) one, and it is not clear from Ref. 30 how exactly such a feat might be accomplished using available patterning techniques.

Figure 3 shows our approach to the interface problem. (We call such circuits CMOL, standing for CMOS/nanowire/MOLecular hybrids.) The difference between the CMOL approach (based on earlier work on the so-called ‘‘InBar’’ networks [31], [32]), and the suggestion discussed above [30] is that in CMOL the CMOS-to-nanowire interface is provided by pins distributed all over the circuit area.⁶ In the generic CMOL circuit (Fig. 3), pins of each type (contacting the bottom and top nanowire levels) are located on a square lattice of period $2\beta F_{\text{CMOS}}$. Relative to these arrays, the nanowire crossbar is turned by a (typically, small) angle α which satisfies two conditions (Fig. 3b):

$$\sin \alpha = F_{\text{nano}}/\beta F_{\text{CMOS}}, \quad (1)$$

$$\cos \alpha = r F_{\text{nano}}/\beta F_{\text{CMOS}}, \quad (2)$$

where r is a (typically, large) integer. Such tilt ensures that a shift by one nanowire (e.g., from the second wire from the left to the third one in Fig. 3c) corresponds to the shift from one interface pin to the next one (in the next row of similar pins), while a shift by r nanowires leads to the next pin in the same row. This trick enables individual addressing of each nanowire even at $F_{\text{nano}} \ll \beta F_{\text{CMOS}}$. For example, the selection of CMOS cells 1 and 2 (Fig. 3c) enables contacts to the nanowires leading to the left one of the two nanodevices shown on that panel. Now, if we keep selecting cell 1, and instead of cell 2 select cell 2’ (using the next CMOS wiring row), we contact the nanowires going to the right nanodevice instead.

⁶Such sharp-pointed pins may be fabricated similarly to the tips used in field-emission arrays - see, e.g., Ref. 33.

It is also clear that if all the nanowires and molecular devices are similar to each other (the assumption that will be accepted in all the following discussion), a shift of the nanowire/molecular subsystem by one nanowiring pitch with respect to the CMOS base does not affect the circuit properties. Moreover, a straightforward analysis of Fig. 3c shows that at an optimal shape of the interface pins, even a complete lack of alignment of these two subsystems leads to a circuit yield loss about 75%. Such loss may be acceptable, taking into account that the cost of the nanosystem fabrication, including the chemically-directed assembly of molecular devices (e.g., from solution [1]-[3], [34]) may be rather low, especially in the context of an unparalleled density of active devices in CMOL circuits. In fact, the only evident physical limitation of the density is the quantum-mechanical tunneling between parallel nanowires. Simple estimates show that the tunneling current becomes substantial at the distance between the wires $F_{\text{nano}} \approx 1.5$ nm. Even by accepting a more conservative value of 3 nm, we get the device density $n = 1/(2F_{\text{nano}})^2$ above 10^{12}cm^{-2} , i.e. at least three orders of magnitude higher than any purely CMOS circuit ever tested.

4 CMOL Memories

The similarity of all molecular devices, that seems necessary for the simplicity of CMOL circuit fabrication, imposes substantial restrictions on architectures and hence possible applications of the circuits. An even more essential restriction comes from the anticipated finite yield of chemically-directed self-assembly of molecular devices, that will hardly ever reach 100%. As a result, all practical CMOL architectures should be substantially defect-tolerant. This tolerance may be most simply implemented in embedded memories and stand-alone memory chips, with their simple matrix structure. In such memories, each molecular device (for example the single-electron latching switch - see Fig. 1) would play the role of a single-bit memory cell, while the CMOS subsystem may be used for coding, decoding, line driving, sensing, and input/output functions.⁷

⁷ It may seem that a large problem in such memories is the necessity for the latching switches to combine a sufficient retention time and write/erase speed (see Footnote 1 in the Introduction). However, in memories the speed requirements may be substantially relaxed: a-few-microsecond write/erase time may be acceptable for some, and a-few-nanosecond time, for most applications. Moreover, the periodic memory refresh (similar to that used in the present-day DRAM) may allow to use cells with retention time as low as a few seconds. Hence, the switching speed ratio (at the doubling of applied voltage) should be from about 5 to 9 orders of magnitude. The former requirement may be easy to satisfy, while the latter challenge may possibly be met using single-electron trap barriers with an appropriate structure [35].

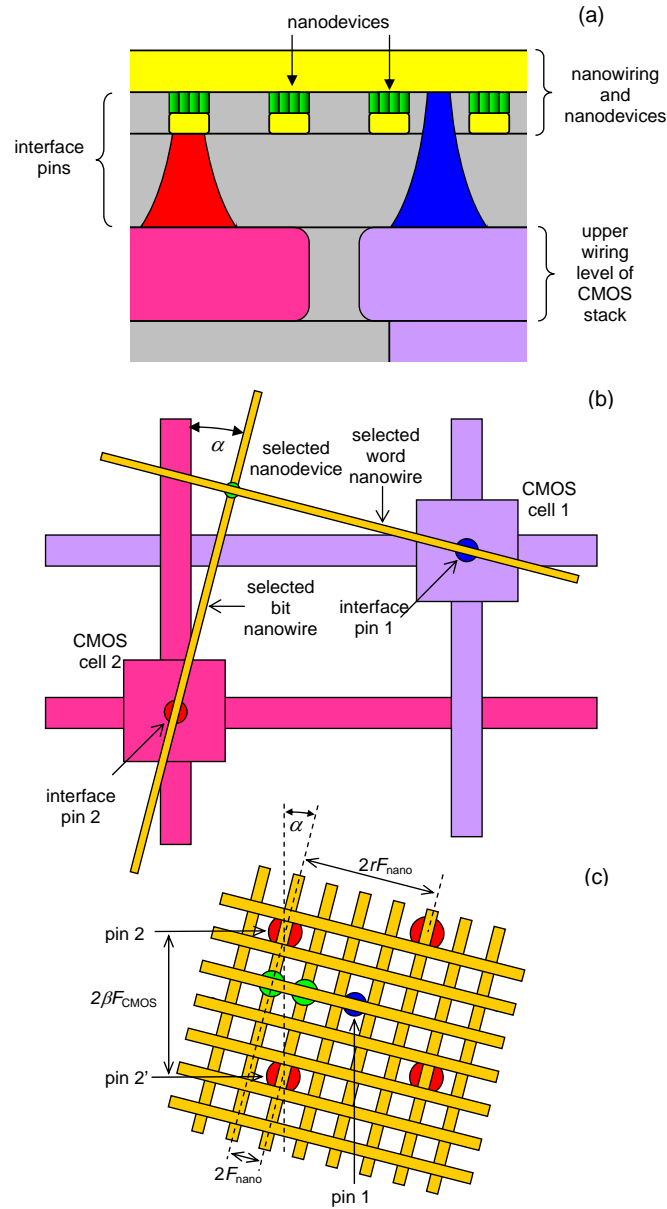


Fig. 3. The generic CMOL circuit: (a) a schematic side view; (b) a schematic top view showing the idea of addressing a particular nanodevice via a pair of CMOS cells and interface pins, and (c) a zoom-in top view on the circuit near several adjacent interface pins. On panel (b), only the activated CMOS lines and nanowires are shown, while panel (c) shows only two devices. (In reality, similar nanodevices are formed at all nanowire crosspoints.) Also disguised on panel (c) are CMOS cells and wiring.

We have carried out [36] a detailed analysis of such memories, including the application of two major techniques for increasing their defect tolerance: the memory matrix reconfiguration (the replacement of several rows and columns, with the largest number of bad memory cells, for spare lines), and error correction (based on the Hamming codes). Figure 4 shows the top structure of the CMOL memory, accepted at that analysis. It is essentially a matrix of L memory blocks, each block in turn being a rectangular array of $(n + a) \times (m + b)$ memory cells. Here a and b are the numbers of spare rows and columns, respectively, while $n \times m$ is the final block size after the reconfiguration. (With the account of error correction, the total number of useful bits in the memory is slightly below the product $n \times m \times L$.) A p -bit word addressed at each particular time step is distributed over p blocks. Each of these bits has the same external word and bit addresses in its block, though due to the internal line re-routing during the initial reconfiguration process (see below), the real physical location of the used memory cell may be different in each block.

Each block is a CMOL matrix, so that at each elementary operation, the block decoders address two vertical and two horizontal lines implemented in the CMOS layers of the circuit, thus selecting a pair of CMOS cells (Fig. 3b). Each cell has a simple “relay” structure (using either one or two pass transistors [36]) and connects one of CMOS-level wires leading to the cell to the corresponding nanowire. As has been explained in Sec. 3 above, this allows the four cell address decoders of each block to reach each memory cell, even if the cell density is much higher than $1/(F_{\text{CMOS}})^2$.

We have started our analysis with the calculation of the block yield y and the full memory yield Y for several combinations of various reconfiguration techniques with Hamming code error correction (assuming so far only one type of defects: the absence of molecular devices at certain crosspoints, formally equivalent to the “stuck-on-open” faults). Figure 5 shows typical results of such calculation for the following cases:

- (i) no reconfiguration, no error correction;
- (ii) simple “Repair Most” reconfiguration algorithm, in which a worst rows of the array (with the largest number of bad bits) are excluded first, and b worst columns of the remaining matrix next; and
- (iii) upper bound for the best possible, but exponentially complex “Exhaustive Search” reconfiguration.

The figure shows that the array reconfiguration (“repair”) may improve the yield rather dramatically, while the difference between the two repair methods is not too large, especially if the number of redundant lines is not too high - below, or of the order of the final memory size. (The difference is somewhat larger if the array reconfiguration is used together with the error correction.)

Our next step was to use the yield calculation results to evaluate the additional memory area necessary to achieve a certain fixed yield, as a function

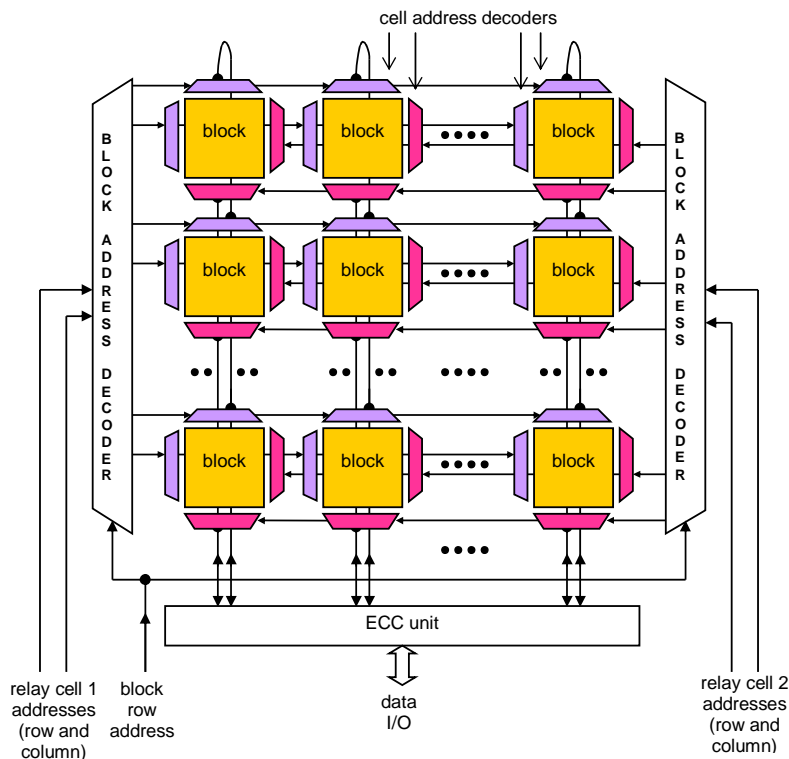


Fig. 4. The top structure of CMOL memory analyzed in Ref. 36. At each instance, block address decoders allow to send the cell row and column addresses to a single row of blocks. The cell addresses are then processed by decoders of each block.

of the memory parameters, in particular the block size (at fixed total memory size). The area is contributed by spare lines necessary for the array configuration, additional parity bits necessary for the Hamming-code error correction, and CMOS components including the decoders, drivers, sense amplifiers and a relatively small CMOS-based memory storing the reconfiguration results.

Figure 6 shows a typical result for the total chip area (per useful bit) as a function of the linear size n of the block. At small n , the area per bit grows because of the contribution of the peripheral CMOS circuits (mostly, the cell address decoders), while at large n it grows because the necessary number of redundant array lines becomes too large. As a result, there always exist a certain block size (and hence the number of blocks in the full memory) that minimizes the area.

Figure 7 shows this optimized area per bit as a function of the molecular device yield, for two values of the $F_{\text{CMOS}}/F_{\text{nano}}$ ratio and two defect tolerance boost techniques. (Results for purely CMOS memories are also shown for comparison.) The results show that the array reconfiguration, especially

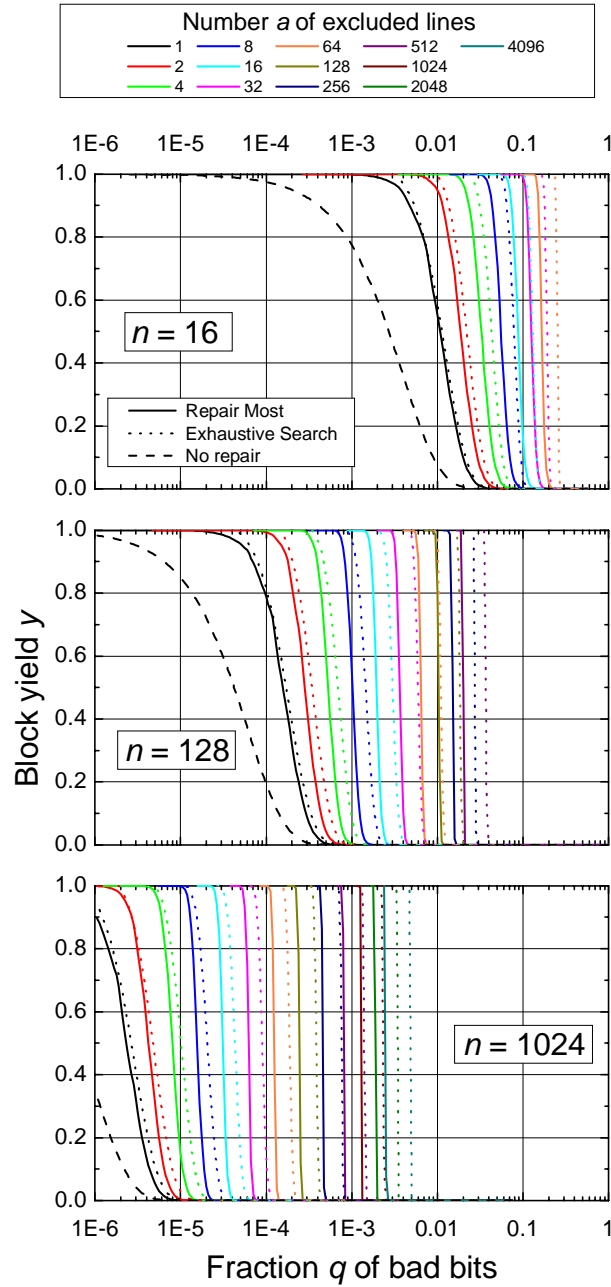


Fig. 5. Comparison of the defect tolerance provided by the two reconfiguration (bad line exclusion) techniques: “Repair Most” (solid lines) and “Exhaustive Search” (dotted lines), without additional error correction, for a square block matrix ($n = m, a = b$). As a reference, dashed lines show the results without the reconfiguration.

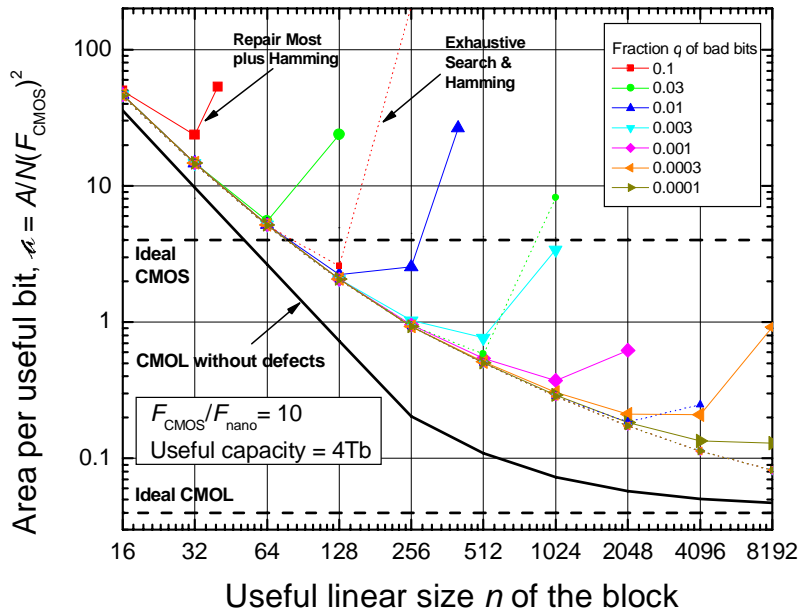


Fig. 6. The reciprocal memory density (area per useful bit) as a function of the block size, for $F_{\text{CMOS}}/F_{\text{nano}} = 10$, and several values of the single device yield. The dashed lines show the single bit footprint, i.e., the reciprocal memory density in the ideal case (no bad devices, no peripheral circuits), for CMOS and nanodevice implementations.

applied in synergy with error correction, can increase the memory defect tolerance very substantially, however, the single bit yield still has to be close to 100%. For example, in a realistic case $F_{\text{CMOS}}/F_{\text{nano}} = 10$, the hybrid memories can overcome a perfect CMOS memory only if the fraction of bad bits is below $\sim 15\%$, even using the Exhaustive Search algorithm of bad bits exclusion, which may require an impracticably long time. For the simple and fast Repair Most algorithm, the bad bit fraction should be reduced to $\sim 2\%$. If one wants to obtain an order-of-magnitude density advantage from the transfer to hybrid memories (such a goal seems natural for the introduction of a novel technology), the numbers given above should be reduced to approximately 2% and 0.1%, respectively.

These results for the required single device yield do not look overly optimistic,⁸ but this should not obscure the fact that when this threshold has been achieved, extremely impressive memories will become available. For example, the normalized cell area $a \equiv A/N(F_{\text{CMOS}})^2 = 0.4$ (Fig. 7) at $F_{\text{CMOS}} = 32 \text{ nm}$ means that a memory chip of a reasonable size ($2 \times 2 \text{ cm}^2$) can store

⁸Our plans are to look for different CMOL memory architectures with a comparable density, but better fault tolerance.

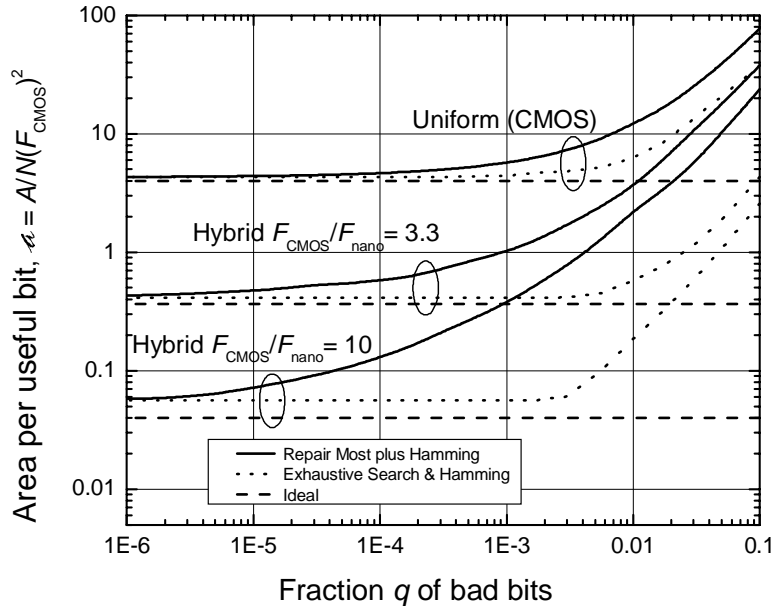


Fig. 7. The area per useful bit after the block size optimization, as a function of single bit yield, for hybrid and purely semiconductor memories.

about 1 terabit of data - crudely, one hundred Encyclopedia Britannica's.⁹

5 CMOL FPGA: Boolean Logic Circuits

The situation with digital (Boolean) logic is even more complex. In the usual custom logic circuits the location of a defective gate from outside is hardly possible, while spreading around additional logic gates (e.g., providing von Neumann's majority multiplexing [37]) for error detection and correction becomes very inefficient for fairly low fraction q of defective devices. For example, even the recently improved von Neumann's scheme requires a 10-fold redundancy for q as low as $\sim 10^{-5}$ and a 100-fold redundancy for $q \approx 3 \times 10^{-3}$ [38].

This is why the most significant previously published proposals for the implementation of logic circuits using CMOL-like hybrid structures had been based on reconfigurable regular structures like the field-programmable gate arrays (FPGA).¹⁰ Before our recent work, two FPGA varieties had

⁹Comparable densities may be achieved in prospective magnetic and electrostatic data storage systems [10], however, in contrast with random access memories they do not allow a virtually instant (nanosecond-scale) access to every data bit.

¹⁰See Refs. 18, 40 for their detailed reviews.

been analyzed, one based on look-up tables (LUT) and another one using programmable-logic arrays (PLA).

In the former case, all possible values of an m -bit Boolean function of n binary operands are kept in m memory arrays, of size $2^n \times 1$ each. (For $m = 1$, and some representative applications the best resource utilization is achieved with n close to 4 [39], while the famous reconfigurable computer Teramac [40] is using LUT blocks with $n = 6$ and $m = 2$.) The main problem with this approach is that the memory arrays of the LUTs based on realistic molecular devices cannot provide address decoding and output signal sensing (recovery). This means that those functions should be implemented in the CMOS subsystem, and the corresponding overhead may be estimated using our results discussed in the previous section. In particular, Fig. 6 shows that for a memory with $2^6 \times 2$ bits, performing the function of a Teramac's LUT block, and for a realistic ratio $F_{\text{CMOS}}/F_{\text{nano}} = 10$ the area overhead would be above four orders of magnitude (!), and would even lose the density (and hence performance) competition to a purely-CMOS circuit performing the same function.¹¹

The PLA approach is based on the fact that an arbitrary Boolean function can be re-written in the canonical form, i.e. in the two-level logical representation. As a result, it may be implemented as a connection of two crossbar arrays, for example one performing the AND, and another the OR function [18]. The first problem with the application of this approach to the CMOS/molecular hybrids is the same as in the case of LUT's: the optimum size of the PLA crossbars is finite, and typically small [42], so that the CMOS overhead is extremely large. Moreover, any PLA logic built with diode-like molecular devices faces an additional problem of high power consumption. In contrast with LUT arrays, where it is possible to have current only through one molecular device at a time, in PLA arrays the fraction of open devices is of the order of one half [18]. Let us estimate the static power dissipated by such an array. The specific capacitance of a wire in an integrated circuit is always of the order of 2×10^{-10} F/m.¹² With $F_{\text{nano}} = 3$ nm, this number shows that in order to make the RC time constant of the nanowire below than, or of the order of the logic delay in modern CMOS circuits ($\sim 10^{-10}$ s), the ON resistance R_0 of a molecular device has to be below $\sim 7 \times 10^7$ ohms. For reliable operation of single-electron transistor (and apparently any other active electronic nanodevice) at temperature T , the scale V_0 of voltage $V = V_s - V_d$ across it has to be at least $10k_B T$ [10]. For room temperature this gives $V_0 > 0.25$

¹¹Increasing the memory array size to the optimum shown in Fig. 6 is not an option, because the LUT performance scales (approximately) only as a log of its capacity [41].

¹²For example, for a simple geometric model of the nanowire crossbar, in which both the width and the thickness of the wire, and both the vertical and the horizontal distances between the nanowires are all equal to F_{nano} , the specific capacitance is close to $C \approx 0.48 \times 10^{-10} \epsilon$ [F/m], where ϵ is the relative dielectric constant of the insulating environment (3.9 for SiO₂) [36].

Volt, so that static power dissipation per one open device, $P_0 = V_0^2/R_0$ is close to 10 nW. With the open device density of $0.5/(2F_{\text{nano}})^2 \approx 10^{12} \text{ cm}^{-2}$, this creates a power dissipation density of at least 10 kW/cm², much higher than the current and prospective technologies allow to manage [11].

As a matter of principle, power consumption may be reduced by using dynamic logic, but this approach requires more complex nanodevices. For example, Refs. 42, 43 describe a dynamic-mode PLA-like structure using several types of molecular-scale devices, most importantly including field-effect transistors formed at crosspoints of two nanowires. In such transistor, one (semiconductor) nanowire would serve as a drain/channel/source structure, while the perpendicular nanowire would play the role of the gate. Unfortunately, such circuits would fail because of the same fundamental physical reason that provides the fundamental limitation the Moore’s Law (see the Introduction): any semiconductor MOSFET with a-few-nm-long channel is irreproducible because of exponential dependence of the threshold voltage on the transistor dimensions [45].

Recently, we suggested [46, 47] an alternative approach to Boolean logic circuits based on CMOL concept, that is close to the so-called cell-based FPGA [48]. In this approach (Fig. 8a, b), an elementary CMOS cell includes two pass transistors and an inverter, and is connected to the nanowire/ molecular subsystem via two pins.¹³ During the configuration process the inverters are turned off, and the pass transistors may be used for setting the binary state of each molecular device, just like described above for CMOL memory. Each pin of a CMOS cell can be connected through a nanowire-nanodevice-nanowire link to each of $M \equiv 2r^2 - 2r - 1$ other cells within a square-shaped “connectivity domain” around the pin (painted light-gray in Fig. 8a). Figure 8c shows how such fabric may be configured for the implementation of a fan-in-two NOR gate. This is already sufficient to implement any logic function (see, e.g., Fig. 9), though gates with larger fan-in and fan-out are clearly possible.

Note that during the circuit operation the switching latches should not change their state, working just either as diodes if they are in the ON state or open circuits with some (parasitic) high resistance if they are turned OFF (Fig. 1a). This is why the switching speed to retention time requirement (see Footnote 1) is relaxed even more than in CMOL memories: while the retention time should be long (at least a few hours, better a few years), the programming time as long as a few seconds may be acceptable, because the programming of the whole circuit requires just $\sim M$ sequential steps.

Generally, there may be many different algorithms to reconfigure the

¹³For convenience of signal input and output, the nanowire crossbar is turned by additional 45° in comparison with the generic CMOL (Fig. 3), so that Eqs. (1), (2) now take the form $\sin \alpha = (r - 1)F_{\text{nano}}/\beta F_{\text{CMOS}}$, $\cos \alpha = rF_{\text{nano}}/\beta F_{\text{CMOS}}$. Also note the breaks in each nanowire in the middle of its contacts with the interface pins.

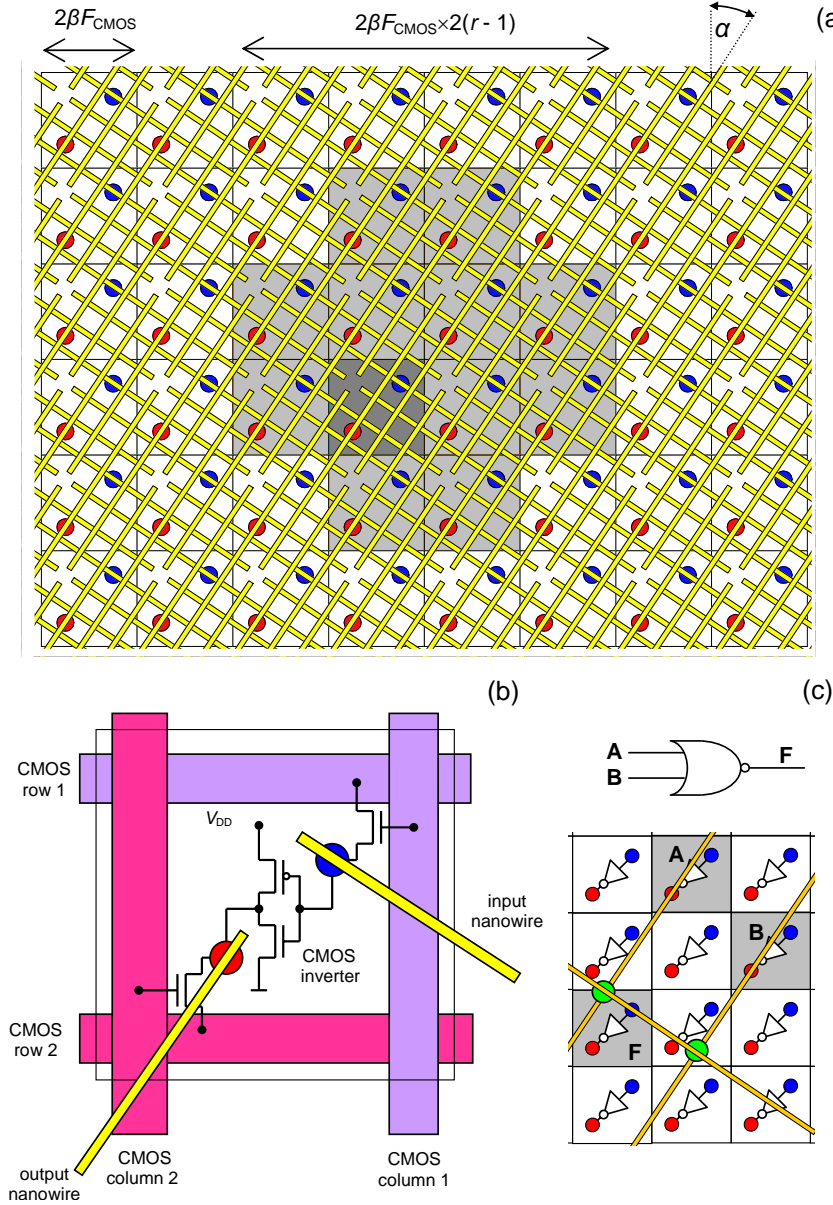


Fig. 8. CMOL FPGA: (a) the general structure of the circuit and (b) a single CMOS cell, and (c) NOR gate implementation. In panel (a), the cells painted light-gray may be connected to the input pin of a specific cell (shown dark-gray). For the sake of clarity, panel (b) shows only two nanowires (that contact the given cell), while panel (c) shows only the three nanowires used inside the NOR gate.

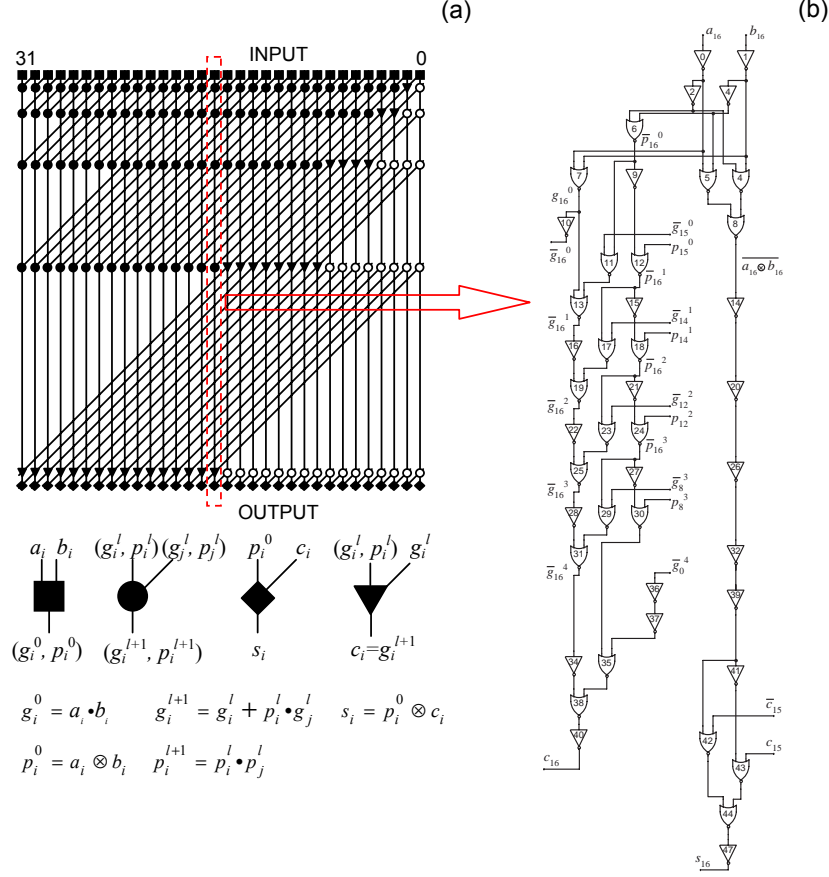


Fig. 9. (a) The 32-bit Kogge-Stone adder and (b) its single (16th) bit slice implemented with NOR gates only.

CMOL FPGA structure around known defects, including quasi-optimal, exhaustive-search options which are impracticable, because the resources required for their implementation are exponential in circuit size. We have developed a simple approach, linear in M , in which the CMOL FPGA configuration is carried out in two stages. First, the desired circuit is mapped on the apparently perfect (defect-free) CMOL fabric.¹⁴ At the second stage, the circuit is reconfigured around defective components using a simple algorithm [46, 47].

Our Monte Carlo simulation (again, so far only for the “no-assembly”-

¹⁴We have found it highly beneficial, from the view of defect tolerance, to confine the cell connections to a smaller square shaped domain of $M' \equiv 2r'^2 - 2r' - 1$ cells, with r' slightly below the maximum connectivity radius r .

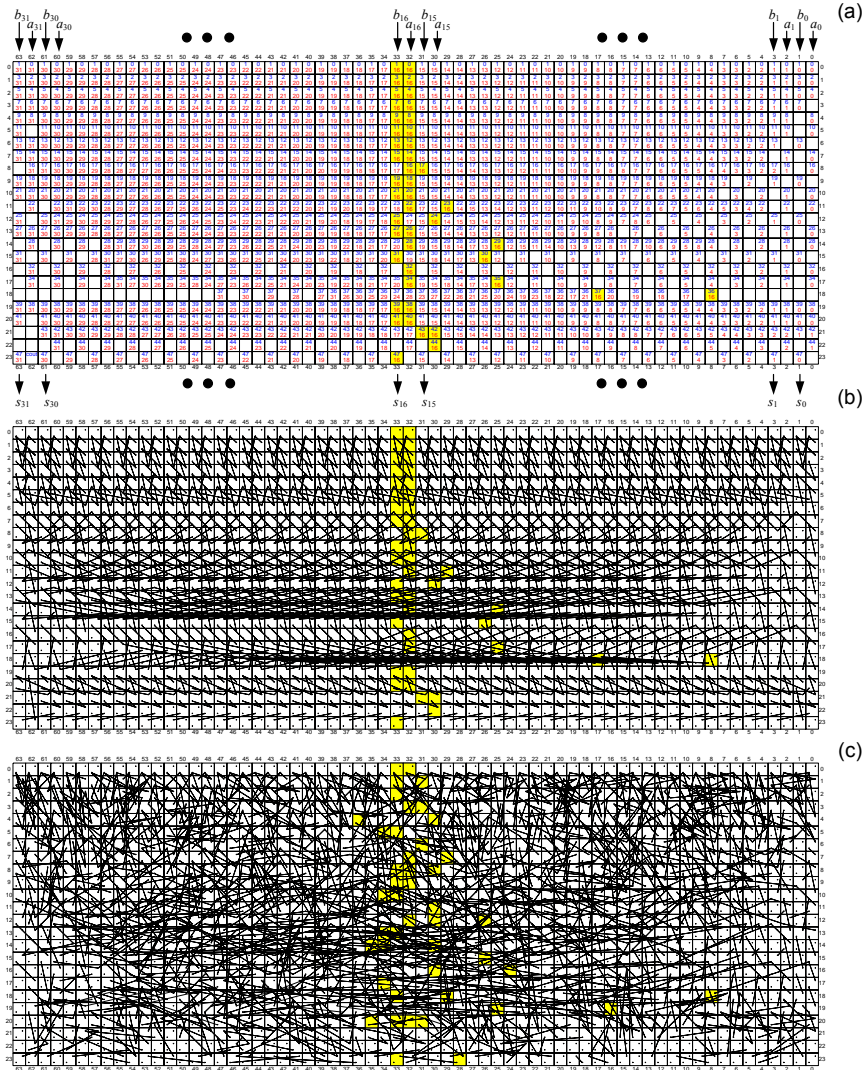


Fig. 10. Mapping of the 32-bit Kogge-Stone adder on CMOL FPGA fabric: (a) the initial cell map, (b) the corresponding initial map of cell connections, and (c) a typical connection map after a successful reconfiguration of the circuit around as many as 50% of randomly located bad nanodevices. Gates of the 16th bit slice (Fig. 9) are painted yellow.

type defects) has shown that even this simple configuration procedure may ensure very high defect tolerance. For example, Fig. 10c shows that the reconfiguration of a simple logic circuit, the 32-bit Kogge-Stone adder [49], mapped on the CMOL fabric with realistic values of parameters $r = 12$ and $r' = 10$, may allow to make fully functional a system with as many as 50% of missing nanodevices. Under a more strict requirement of the 99% circuit yield (sufficient for a 90% yield of properly organized VLSI chips), the defect tolerance of this circuit is about 22%, while that of another key circuit, a fully-connected 64-bit crossbar switch, is about 25%. These impressive results may be explained by the fact that each CMOS cell is served by $M \gg 1$ nanodevices used mostly for reconfiguration.

It is especially important that CMOL FPGA circuits may combine such high defect tolerance with high density and performance, at acceptable power consumption. Indeed, approximate estimates have shown [46, 47] that for the power of 200 W/cm^2 (planned by the ITRS for the long-term CMOS technology nodes [11]), an optimization of the power supply voltage V_{DD} may bring the logic delay of the 32-bit Kogge-Stone adder down to just 1.9 ns, at the total area of $110 \mu\text{m}^2$, i.e. provide an area-delay product of $150 \text{ ns}\cdot\mu\text{m}^2$, for realistic values $F_{\text{CMOS}} = 32 \text{ nm}$ and $F_{\text{nano}} = 8 \text{ nm}$ (Fig. 11c). This result should be compared with the estimated $70,000 \text{ ns}\cdot\mu\text{m}^2$ (with 1.7 ns delay and $39,000 \mu\text{m}^2$ area) for a fully CMOS FPGA implementation of the same circuit (with the same $F_{\text{CMOS}} = 32 \text{ nm}$).

A more full evaluation of the CMOL FPGA concept would require the simulation of a substantial number of various functional units and other circuits necessary for digital signal processing and/or general-purpose computing. (This work will probably require, in turn, the development of new, or a modification of existing CAD tools.) Eventually, CMOL FPGA systems should be evaluated on the generally accepted computing benchmarks. However, we believe that even the preliminary estimates described above give a strong evidence that this approach may far outperform CMOS FPGAs in virtually all areas of their application.

The comparison between CMOL FPGA and custom CMOS chips is a more complex issue.¹⁵ Indeed, in the sample circuits explored so far, each CMOS cell is using just a few latching switches for actual operation. As we have seen, this gives a spectacular defect tolerance, but provides only a limited increase in the function density. However, nothing in our CMOL design prevents using gates with much higher fan-in, for which the function density will be substantially improved, hopefully with only a modest sacrifice of the defect tolerance. A quantitative study of this opportunity is one of our immediate goals.

¹⁵If we leave alone the fact that the FPGA approach allows to bypass the current bottleneck of VLSI chip design, i.e. design productivity, which is one of the major problems of microelectronics [11].

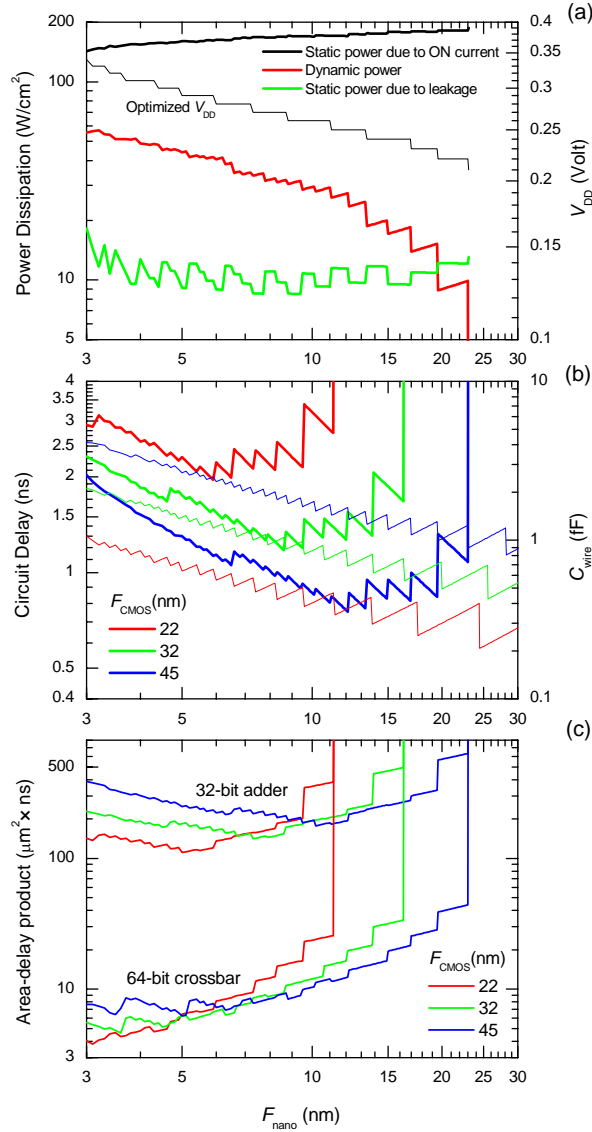


Fig. 11. CMOL FPGA optimization results as functions of nanowire half-pitch: (a) three components of the total power (fixed at $200 \text{ W}/\text{cm}^2$), and the optimum value of the power supply voltage V_{DD} , for the 32-bit Kogge-Stone adder with $F_{\text{CMOS}} = 45 \text{ nm}$; (b) nanowire segment capacitance (thin lines) and the total logic delay of the circuit (bold lines); and (c) area-delay product $A\tau$ of the two CMOL FPGA circuits for three ITRS long-term CMOS technology nodes. The (formal) jump of the $A\tau$ product to infinity at some $(F_{\text{nano}})_{\text{max}}$ reflects the fact that circuit mapping on the CMOL fabric may only be implemented for F_{nano} below this value. The finite sharp jumps of the curves are due to the discrete changes of angle α .

6 CMOL CrossNets: Neuromorphic Networks

The requirement of high defect tolerance gives an incentive to consider CMOL implementation of alternative information processing architectures, in particular analog or mixed-signal neuromorphic networks (see, e.g., Ref. 50), because such networks are by their structure deeply parallel and hence inherently defect-tolerant. An additional motivation for using neuromorphic networks comes from the following comparison of the performance of the biological neural systems and present-day Boolean-logic computers in one of the basic advanced information processing tasks: image recognition (more strictly speaking, classification [50]). A mammal’s brain recognizes a complex visual image, with high fidelity, in approximately 100 milliseconds. Since the elementary process of neural cell-to-cell communication in the brain takes approximately 10 milliseconds, this means that the recognition is completed in just a few “clock ticks”. In contrast, the fastest modern microprocessors performing digital number crunching at a clock frequency of a few GHz and running the best commercially available code, would require many minutes (i.e., of the order of 10^{12} clock periods) for an inferior classification of a similar image. The contrast is very striking indeed, and serves as a motivation for the whole field of artificial “neural networks”.

Presently, these networks are mostly just a concept for writing software codes that are implemented on usual digital computers. Unfortunately, the high expectations typical for the neural network’s “heroic period” (from the late 1980s to the early 1990s) have not fully materialized, in particular because the computer resources limit the number of neural cells to a few hundreds, insufficient for performing really advanced, intelligent information processing tasks. The advent of hybrid CMOL circuits may change the situation.

Recently, our group suggested [31], [32] a new family of neuromorphic network architectures, Distributed Crosspoint Networks (“CrossNets” for short) that map uniquely on the CMOL topology. Each such network consists of the following components:

(i) Neural cell bodies (“somas”) that are relatively sparse and hence may be implemented in the CMOS subsystem. Most of our results so far have been received within the simplest Firing Rate approach [50], in which somas operate just as differential amplifiers, with a nonlinear saturation (“activation”) function, which are fed by the incoming (dendritic) nanowires and apply their output signal to outgoing (axonic) wires.

(ii) “Axons” and “dendrites” that are implemented as mutually perpendicular nanowires of the CMOL crossbar.

(iii) “Synapses” that control coupling between the axons and dendrites (and hence between neural cells) based on the molecular latching switches (see Fig. 1 and its discussion).

CrossNet species differ by the number and direction of intercell couplings (Fig. 12) and by the location of somatic cells on the axon/dendrite/synapse field (Fig. 13). The cell distribution pattern determines the character of cell

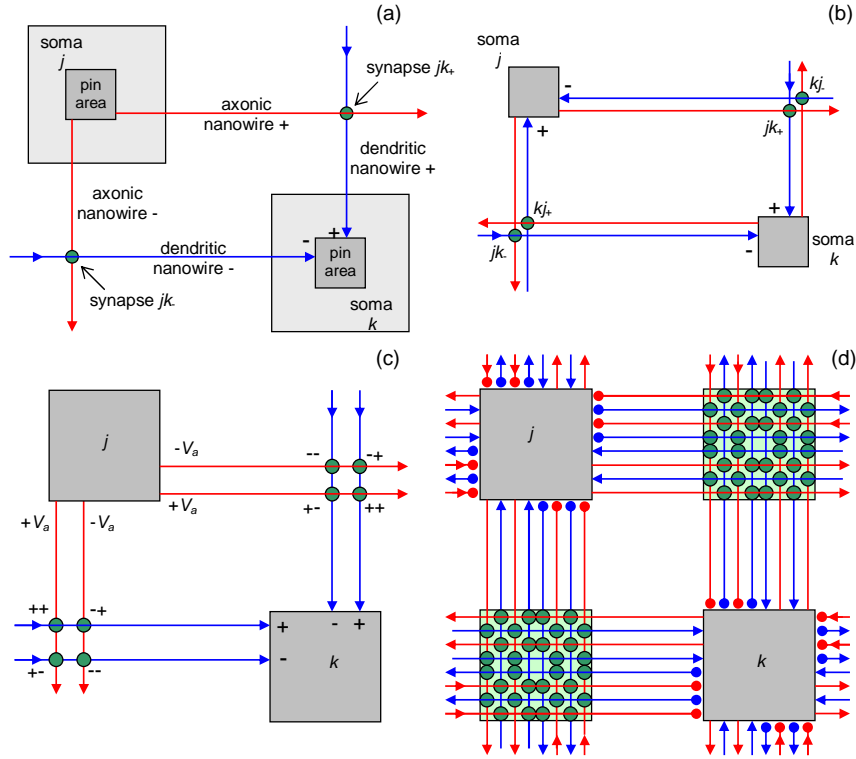


Fig. 12. Schemes of cell connections in CrossNets: (a) simple (non-Hebbian) feed-forward network, (b) simple recurrent network, (c) Hebbian feedforward CrossNet and (d) Hebbian recurrent CrossNet[51]. Red lines show “axonic”, and blue lines “dendritic” nanowires. Dark-gray squares are interfaces between nanowires and CMOS-based cell bodies (somas), while light-gray squares in panel (a) show the somatic cells as a whole. (For the sake of clarity, the latter areas are not shown in the following panels and figures.) Signs show the somatic amplifier input polarities. Green circles denote nanodevices (latching switches) forming elementary synapses. For clarity the panels (a)-(c) show only the synapses and nanowires connecting one couple of cells (j and k). In contrast, panel (d) shows not only those synapses, but also all other functioning synapses located in the same “synaptic plaquettes” (painted light-green) and the corresponding nanowires, even if they connect other cells. (In CMOL circuits, molecular latching switches are also located at all axon/axon and dendrite/dendrite crosspoints; however, they do not affect the network dynamics, resulting only in approximately 50% increase of power dissipation.) The solid dots on panel (d) show open-circuit terminations of synaptic and axonic nanowires, that do not allow direct connections of the somas, in bypass of synapses.

coupling. For example, the “FlossBar” (Fig. 13a) has a layered structure typical for the so-called multilayered perceptrons [50], while the “InBar” (in which somas sit on a square lattice inclined by a small angle relatively the axonic/dendritic lattice, Fig. 13b) implements a non-layered “interleaved” network. Also important is the average distance M between the somas, that determines connectivity of the networks, i.e. the average number of other cells coupled directly (i.e., via one synapse) to a given soma. The most remarkable property of CMOL CrossNets is that the connectivity of these (quasi-)2D structures may be very large. This property is very important for advanced information processing, and distinguishes CrossNets favorably from the so-called cellular automata with small (next-neighbor) connectivity which severely limits their functionality.

In contrast to the usual computers, neuromorphic networks do not need

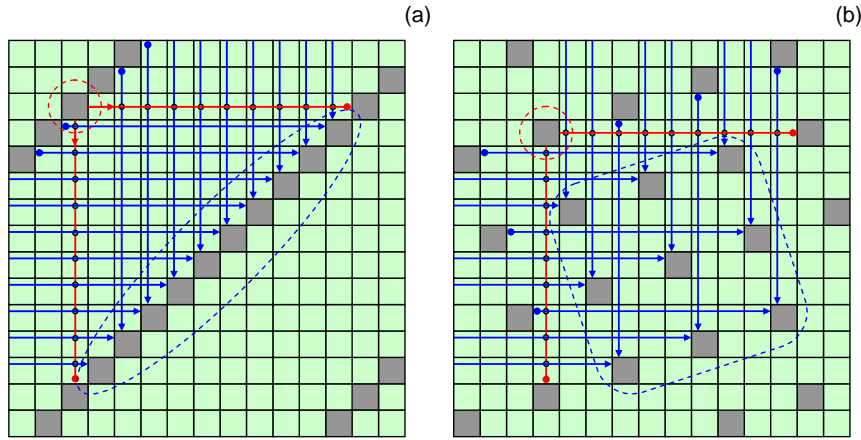


Fig. 13. Two particular CrossNet species: (a) FlossBar and (b) InBar. For clarity, the figures show only the axons, dendrites, and synapses providing connections between one soma (indicated by the dashed red circle) and its recipients (inside the dashed blue lines), for the simple (non-Hebbian) feedforward network.

an external software code, but need to be “trained” to perform certain tasks. For that, the synaptic connections between the cells should be set to certain values. The neural network science has developed several effective training methods [50]. The application of these methods to CMOL CrossNets faces several hardware-imposed challenges:

- (i) CrossNets use continuous (analog) signals, but the synaptic weights are discrete (binary, if only one latching switch per synapse is used).
- (ii) The only way to reach for any particular synapse in order to turn it on or off is through the voltage V applied to the device through the two corresponding nanowires. Since each of these wires is also connected to many

other switches, special caution is necessary to avoid undesirable “disturb” effects.

(iii) Processes of turning single-electron latches on and off are statistical rather than dynamical [12], so that the applied voltage V can only control probability rates Γ of these random events.

In our recent work [51] we have proved that, despite these limitations,

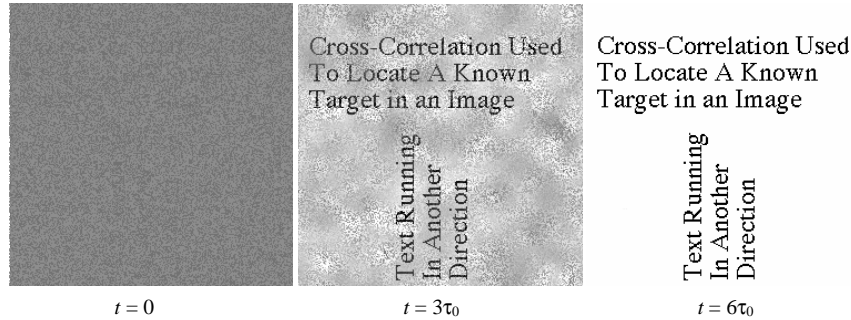


Fig. 14. The recall of one of three trained black-and-white images by a recurrent InBar-type CrossNet with 256×256 neural cells, binary synapses, and connectivity $4M = 64$, operating in the local quasi-Hopfield mode. The initial image (left panel) was obtained from the trained image (identical to the one shown in the right panel) by flipping as many as 40% of randomly selected pixels.

CrossNets can be taught, by at least two different methods, to perform virtually all the major functions demonstrated earlier with usual neural networks, including the corrupted pattern restoration in the recurrent quasi-Hopfield mode (Fig. 14) and pattern classification in the feedforward multilayered perceptron mode [51, 52].¹⁶ Moreover, at least in the former mode the CrossNets can be spectacularly resilient. For example, operating at network capacity just a half of its maximum, a quasi-Hopfield CrossNet may provide a 99% result fidelity with as many as 85% (!) of bad molecular devices - see Fig. 15. This defect tolerance is much higher than that of CMOL memories (see Sec. 4 above) and even that of CMOL FPGA circuits (Sec. 5).

The fact that CrossNets may perform the tasks that had been demonstrated with artificial neural networks earlier may seem not very impressive until the possible performance of this hardware is quantified. Estimates [31, 32, 51] show that for realistic parameters as have been used in Sec. 4 above ($F_{\text{nano}} = 4 \text{ nm}$, $V = 0.25 \text{ Volt}$), and a very respectable connectivity parameter $M \sim 10^3$, the areal density of CrossNets may be at least as high

¹⁶In order to operate as perceptron-type classifiers, CrossNets require multi-latch synapses. This increase can be achieved by using small (e.g., 4×4) square fragments of CrossNet arrays for each synapse [51]. This increase is taken into account in the density estimates given below.

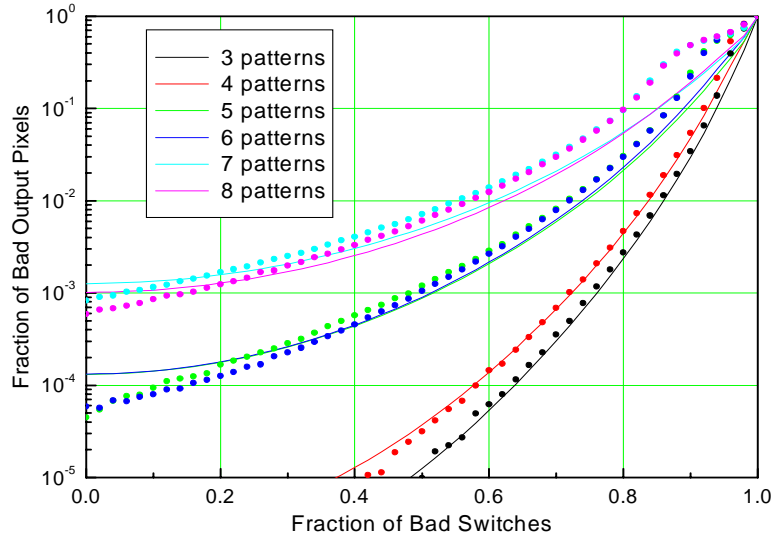


Fig. 15. Defect tolerance of a recurrent InBar with connectivity parameter $M = 25$, operating in the Hopfield mode. Lines show the results of an approximate analytical theory, while dots those of a numerical experiment.

as that of the cerebral cortex (above 10^7 cells per cm^2), while the average cell-to-cell communication delay τ_0 may be as low as ~ 10 ns (i.e., about six orders of magnitude lower than in the brain), at power dissipation below $100 \text{ W}/\text{cm}^2$.¹⁷ This implies, for example, that a 1-cm^2 CMOL CrossNet chip would be able to recognize a face in a high-resolution image of a crowd faster than in 100 microseconds [52]. We believe that such applications alone may form not just a narrow market niche, but a substantial market for the hybrid CMOS/molecular electronics.

Moreover, there is a hope that CrossNets will be able to perform even more complex intelligent tasks if trained using more general methods such as global reinforcement [50, 53]. (A successful result of a very preliminary attempt at such training is presented in Fig. 16.) If these hopes materialize, there will be a chance that such pre-training of a properly organized, hierarchical CrossNet-based system,¹⁸ may help it to reach a functionality comparable with that of a newborn child brain, in some sense replacing the DNA-based genetic

¹⁷The reason for such a large difference with power estimates for Boolean logic circuits (Sec. 5) is that in neuromorphic networks we can afford to increase the open molecular latch resistance to $\sim 10^9$ ohms, and thus increase the logic delay from ~ 100 ps to ~ 10 ns, still providing an extremely high integrated circuit performance ($\sim 10^{12}/10^{-8} \approx 10^{20}$ of a-few-bit operations per cm^2 per second) due to the natural parallelism of the neuromorphic network operation.

¹⁸It is curious that the use of the so-called “X layout”, frequently employed in VLSI circuits, for the CMOS-based subsystem of fast, long-range communications

inheritance. It seems possible that a connection of such pre-trained system to a proper informational environment via a high-speed communication network may trigger a self-development process that may be several orders of magnitude faster than that of the biological cerebral cortex. The reader is invited to imagine possible consequences of such self-development liberated from the dead weight artifacts of the biological evolution.

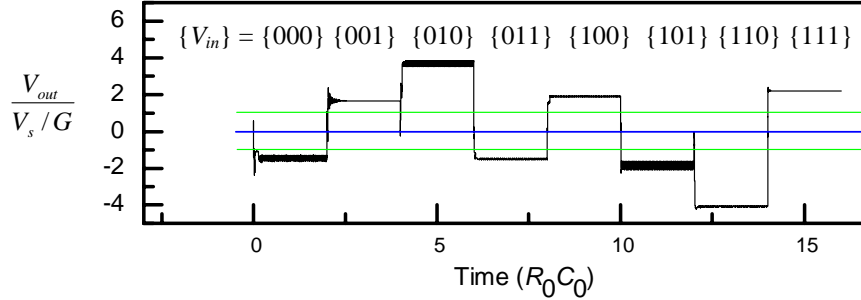


Fig. 16. A result of the global-reinforcement training of a small CrossNet: the dendritic signal of the output stage of a recurrent InBar with quasi-continuous Hebbian synapses, trained to calculate the parity of three binary inputs. All the values of V_{out} above the upper green line correspond to binary 1, while those below the bottom green line, to binary 0.

7 Conclusions

There is a chance for the development, within the next 10 to 20 years, of hybrid “CMOL” integrated circuits that will allow to extend Moore’s Law to the few-nm range. Preliminary estimates show that such circuits could be used for several important applications, notably including terabit-scale memories, reconfigurable digital circuits with multi-teraflops-scale performance, and mixed-signal neuromorphic networks that may, for the first time, compete with biological neural systems in areal density, far exceeding them in speed, at acceptable power dissipation. We believe that these prospects more than justify large-scale research and development efforts in the synthesis of functional molecular devices, their chemically-directed self-assembly, nanowire patterning, and CMOL circuit architectures.

Acknowledgements

between CrossNet blocks automatically leads to a circuit geometry (Fig. 17) that reminds the mammal brain structure.

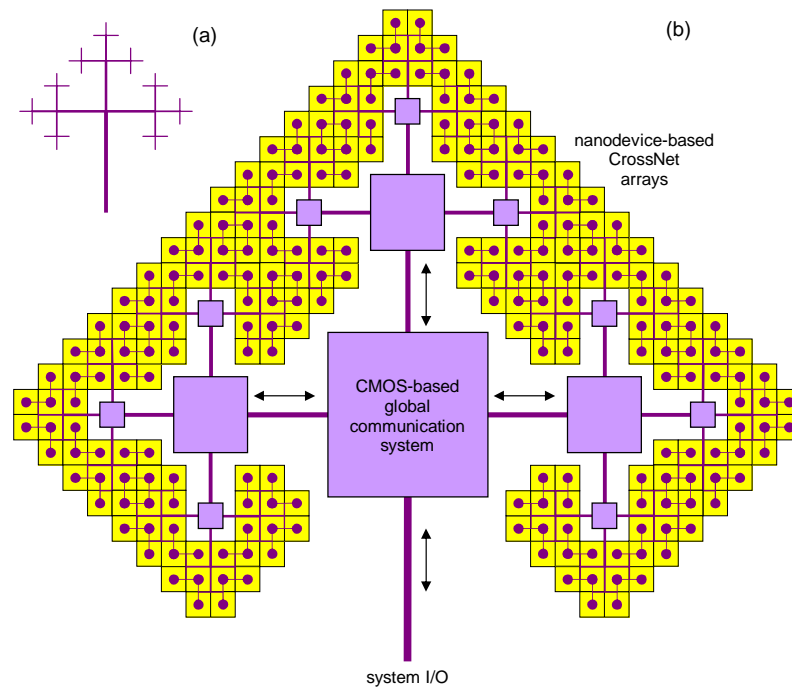


Fig. 17. X layout for the high-speed global communication network: (a) general structure and (b) possible 2D geometry of a hierarchical CMOL CrossNet system using such layout. For the parameters cited above ($F_{\text{nano}} = 3 \text{ nm}$ and $4M = 10^4$), a $30 \times 30\text{-cm}^2$ system may have as many neural cells ($\sim 2 \times 10^{10}$) and synapses ($\sim 10^{15}$) as the human cerebral cortex, while operating at much higher speed, at manageable power.

Useful discussions of the issues considered in this chapter with P. Adams, J. Barhen, V. Beiu, W. Chen, E. Cimpoiasu, S. Das, J. Ellenbogen, J. H. Lee, X. Liu, J. Lukens, X. Ma, A. Mayr, V. Protopopescu, M. Reed, M. Stan, and Ö. Türel are gratefully acknowledged. Figure 1c is a courtesy by A. Mayr. The work on this topic at Stony Brook was supported in part by AFOSR, NSF, and MARCO via FENA Center.

References

1. J. R. Heath and M. A. Ratner: Molecular electronics, *Physics Today* **56**, 43 (2003)
2. J. R. Reimers, C. A. Picconatto, J. C. Ellenbogen, and R. Shashidhar (eds.): *Molecular Electronics III*, *Ann. New York Acad. Sci.* **1006** (2003)
3. J. Tour: *Molecular Electronics* (World Scientific, Singapore 2003)

4. H. Park, J. Park, A. K. L. Lim, E. H. Anderson, A. P. Alivisatos, and P. L. McEuen: Nanomechanical oscillations in a single-C-60 transistor, *Nature* **407**, 57 (2000)
5. S. P. Gubin, Y. V. Gulyaev, G. B. Khomutov, V. V. Kislov, V. V. Kolesov, E. S. Soldatov, K. S. Sulaimankulov, and A. S. Trifonov: Molecular clusters as building blocks for nanoelectronics: The first demonstration of a cluster single-electron tunnelling transistor at room temperature, *Nanotechnology* **31**, 185 (2002)
6. N. B. Zhitenev, H. Meng, and Z. Bao: Conductance of small molecular junctions, *Phys. Rev. Lett.* **88**, 226801 (2002)
7. J. Park, A. N. Pasupathy, J. I. Goldsmith, C. Chang, Y. Yaish, J. R. Petta, M. Rinkoski, J. P. Sethna, H. D. Abruña, P. L. McEuen, and D. C. Ralph: Coulomb blockade and the Kondo effect in single-atom transistors, *Nature* **417**, 722 (2002)
8. S. Kubatkin, A. Danilov, M. Hjort, J. Cornil, J. L. Bredas, N. Stuhr-Hansen, P. Hedegard, and T. Bjornholm: Single-electron transistor of a single organic molecule with access to several redox states, *Nature* **425**, 698 (2003)
9. D. J. Frank, R. H. Dennard, E. Nowak, P. M. Solomon, Y. Taur, and H. S. P. Wong: Device scaling limits of Si MOSFETs and their application dependencies, *Proc. IEEE* **89**, 259 (2001)
10. K. K. Likharev: Electronics below 10 nm, in *Nano and Giga Challenges in Microelectronics* (Elsevier, Amsterdam 2003), pp. 27-68
11. *International Technology Roadmap for Semiconductors. 2003 Edition, 2004 Update*, available online at <http://public.itrs.net/>
12. K. K. Likharev: Single-electron devices and their applications, *Proc. IEEE* **87**, 606 (1999)
13. P. J. Kuekes, D. R. Stewart, and R. S. Williams: The crossbar latch: Logic value storage, restoration, and inversion in crossbar circuits, *J. Appl. Phys.* **97**, 034301 (2005)
14. W. D. Brown and J. E. Brewer (eds.): *Nonvolatile Semiconductor Memory Technology* (IEEE Press, Piscataway, NJ 1998)
15. Y. Chen, G. Y. Jung, D. A. A. Ohlberg, X. M. Li, D. R. Stewart, J. O. Jeppesen, K. A. Nielsen, J. F. Stoddart, and R. S. Williams: Nanoscale molecular-switch crossbar circuits, *Nanotechnology* **14**, 462 (2003)
16. Z. H. Zhong, D. L. Wang, Y. Cui, M. W. Bockrath, and C. M. Lieber: Nanowire crossbar arrays as address decoders for integrated nanosystems, *Science* **302**, 1377 (2003)
17. C. Li, W. D. Fan, B. Lei, D. H. Zhang, S. Han, T. Tang, X. L. Liu, Z. Q. Liu, S. Asano, M. Meyyappan, J. Han, and C. W. Zhou: Multilevel memory based on molecular devices, *Appl. Phys. Lett.* **84**, 1949 (2004)
18. M. R. Stan, P. D. Franzon, S. C. Goldstein, J. C. Lach, and M. M. Ziegler: Molecular electronics: From devices and interconnect to circuits and architecture, *Proc. IEEE* **91**, 1940 (2003)
19. S. Das, G. Rose, M. M. Ziegler, C. A. Picconatto, and J. C. Ellenbogen: Architectures and simulations for nanoprocessor systems integrated on the molecular scale, Chapter 17 of this collection (2005)
20. W. Wang, T. Lee, and M. Reed: Intrinsic electronic conduction mechanisms in self-assembled monolayers, Chapter 10 of this collection (2005)
21. D. Porath: DNA-based devices, Chapter 15 of this collection (2005)
22. L. Ji, P. D. Dresselhaus, S. Y. Han, K. Lin, W. Zheng, and J. E. Lukens: Fabrication and characterization of single-electron transistors and traps, *J. Vac. Sci. Technol. B* **12**, 3619 (1994)

23. C. P. Collier, E. W. Wong, M. Belohradsky, F. M. Raymo, J. F. Stoddart, P. J. Kuekes, R. S. Williams, and J. R. Heath: Electronically configurable molecular-based logic gates, *Science* **285**, 391 (1999)
24. C. P. Collier, G. Mattersteig, E. W. Wong, Y. Luo, K. Beverly, J. Sampaio, F. M. Raymo, J. F. Stoddart, and J. R. Heath: A [2]catenane-based solid state electronically reconfigurable switch, *Science* **289**, 1172 (2000)
25. S. Fölling, Ö. Türel, and K. K. Likharev: Single-electron latching switches as nanoscale synapses, in *Proceedings of the 2001 International Joint Conference on Neural Networks* (Int. Neural Network Soc., Mount Royal, NY 2001), pp. 216-221
26. S. Zankovych, T. Hoffmann, J. Seekamp, J. U. Bruch, and C. M. S. Torres: Nanoimprint lithography: Challenges and prospects, *Nanotechnology* **12**, 91 (2001)
27. S. R. J. Brueck: There are no fundamental limits to optical lithography, in *International Trends in Applied Optics* (SPIE Press, Bellingham, WA 2002), pp. 85-109
28. P. J. Kuekes and R. S. Williams: Demultiplexer for a molecular wire crossbar network (MWCN DEMUX), US Patent No. 6,256,767 (July 3, 2001)
29. A. DeHon, P. Lincoln, and J. E. Savage: Stochastic assembly of sublithographic nanoscale interfaces, *IEEE Trans. on Nanotechnology* **2**, 165 (2003)
30. M. M. Ziegler and M. R. Stan: CMOS/nano co-design for crossbar-based molecular electronic systems, *IEEE Trans. on Nanotechnology* **2**, 217 (2003)
31. Ö. Türel and K. K. Likharev: CrossNets: Possible neuromorphic networks based on nanoscale components, *Int. J. of Circuit Theory and Appl.* **31**, 37 (2003)
32. K. K. Likharev, A. Mayr, I. Muckra, and O. Türel: CrossNets: High-performance neuromorphic architectures for CMOL circuits, *Ann. New York Acad. Sci.* **1006**, 146 (2003)
33. K. L. Jensen: Field emitter arrays for plasma and microwave source applications, *Physics of Plasmas* **6**, 2241 (1999)
34. J. H. Fendler: Chemical self-assembly for electronic applications, *Chemistry of Materials* **13**, 3196 (2001)
35. K. K. Likharev: Riding the crest of a new wave in memory [NOVORAM], *IEEE Circuits and Devices* **16**, 16 (July 2000)
36. D. B. Strukov and K. K. Likharev: Prospects for terabit-scale nanoelectronic memories, *Nanotechnology* **16**, 137 (2005)
37. J. von Neumann: Probabilistic logics and the synthesis of reliable organisms from unreliable components, in *Automata Studies* (Princeton University Press, Princeton, NJ 1956), pp. 329-78
38. S. Roy and V. Beiu: Multiplexing schemes for cost-effective fault-tolerance, Report at IEEE-NANO'04 (Münich, Germany, Aug. 2004); accepted for publication in *IEEE Trans. on Nanotechnology* (2005)
39. J. Rose, R. J. Francis, D. Lewis, and P. Chow: Architecture of field-programmable gate arrays - the effect of logic block functionality on area efficiency, *IEEE J. of Solid-State Circuits* **25**, 1217 (1990)
40. J. R. Heath, P. J. Kuekes, G. S. Snider, and R. S. Williams: A defect-tolerant computer architecture: Opportunities for nanotechnology, *Science* **280**, 1716 (1998)
41. E. Ahmed and J. Rose: The effect of LUT and cluster size on deep-submicron FPGA performance and density, *IEEE Trans. on VLSI Syst.* **12**, 288 (2004)

42. J. Kouloheris and A. E. Gamal: PLA-based FPGA versus cell granularity, in *Proceedings of the Custom Integrated Circuits Conference* (IEEE Press, Piscataway, NJ 1992), pp. 4.3.1-4
43. A. DeHon: Law of large numbers system design, in *Nano, Quantum and Molecular Computing* (Kluwer Academic Publishers, Boston, MA 2004)
44. A. DeHon and M. J. Wilson: Nanowire-based sublithographic programmable logic arrays, in: *Proc. of FPGA'04* (Monterey, CA 2004), pp. 123-132.
45. V. A. Sverdlov, T. J. Walls, and K. K. Likharev: Nanoscale silicon MOSFETs: A theoretical study, *IEEE Trans. on Electron Devices* **50**, 1926 (2003)
46. D. B. Strukov and K. K. Likharev: A reconfigurable architecture for hybrid CMOS/nanodevice circuits, submitted for presentation at FCCM'05 (Napa Valley, CA, April 2005); preprint available online at <http://rsfq1.physics.sunysb.edu/likharev/nano/FCCM2005.pdf>.
47. D. B. Strukov and K. K. Likharev: CMOL FPGA: A cell-based, reconfigurable architecture for hybrid digital circuits using two-terminal nanodevices, submitted to *Nanotechnology*; preprint available online at <http://rsfq1.physics.sunysb.edu/likharev/nano/FPGA05.pdf>.
48. J. Rabaey, A. Chandrakasan, and B. Nikolic: *Digital Integrated Circuits, A Design Perspective* (Pearson Education, Singapore 2003)
49. P. M. Kogge, and H. S. Stone: Parallel algorithm for efficient solution of a general class of recurrence equations, *IEEE Trans. on Computers* **22**, 783 (1973)
50. J. Hertz, A. Krogh, and R. G. Palmer: *Introduction to the Theory of Neural Computation* (Perseus, Cambridge, MA 1991)
51. Ö. Türel, J. H. Lee, X. Ma, and K. K. Likharev: Neuromorphic architectures for nanoelectronic circuits, *Int. J. of Circuit Theory and Appl.* **32**, 277 (2004)
52. J. H. Lee and K. K. Likharev: CMOL CrossNets as pattern classifiers, submitted for presentation at the 8th Int. Work-Conference on Artificial Neural Networks (Barcelona, Spain, June 2005); preprint available at <http://rsfq1.physics.sunysb.edu/likharev/nano/IWANN05.pdf>
53. R. S. Sutton and A. G. Barto: *Reinforcement Learning* (MIT Press, Cambridge, MA 1998)

Index

- Boolean logic, 1, 14, 26
- CMOL circuits, 1, 6
- CMOL technology, 1, 3, 6
- CMOS technology, 1
- CrossNet, 1, 22
- defect tolerance, 1, 13, 18, 20, 25, 26
- fabrication yield, 1, 5, 8, 13, 20
- floating electrodes, 5
- floating gate, 4
- interface, 6
- isocyanide groups, 2, 4
- latching switch, 2, 4, 8, 16, 20, 22–24
- memories, 1, 8, 27
- Moore's Law, 2, 16, 27
- MOSFET, 2, 16
- nanowire crossbar, 1, 6, 7, 15, 16, 22
- neuromorphic networks, 1, 22, 27
- OPE chains, 4
- pattern classification, 1, 25
- pattern recognition, 1, 25
- power consumption, 1, 15, 20, 26, 28
- retention time, 2, 8, 16
- self-assembly, 1, 2, 4, 8, 27
- single-electron transistor, 1, 4, 15
- single-electron trap, 4, 8
- single-electronics, 2, 4
- switching speed, 2, 8, 16
- tunneling, 3, 4, 8
- two-terminal devices, 1, 2, 4
- voltage gain, 2, 3

