Hoang Do
Daniel Benjamin
Scribe Report

CMOL Circuit Architectures

Outline:

- Crisis of microelectronics.
- Basic idea behind CMOL
- Structure of hybrid circuits
- Advantages of CMOL
- Applications

Current VLSI Paradigm:

- Lithography
- CMOS circuits → Cannot be extended into a few nm-region
- Boolean logic

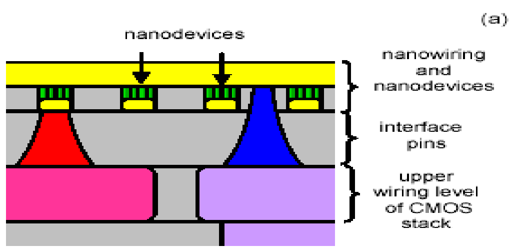
Solution: Shift to bottom-up approach

Application areas: digital memories, reconfigurable Boolean logic, etc.

Idea behind CMOL:

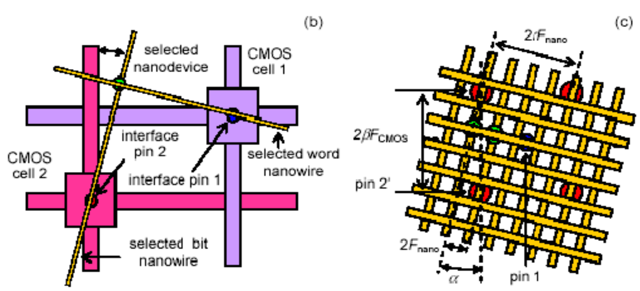
Combines the advantages of CMOS (flexible, high fabrication yield, lower cost) with the advantages of nanodevices (small, high potential density).

Dubbed CMOL:



Difference from interfaces that we have seen: Using pins instead of contacts. These pins are smaller than contacts (in terms of contact points – tips). Original application of these pins is field emission. There are 2 types of pins (red and blue) reaching different layers of nanowires.

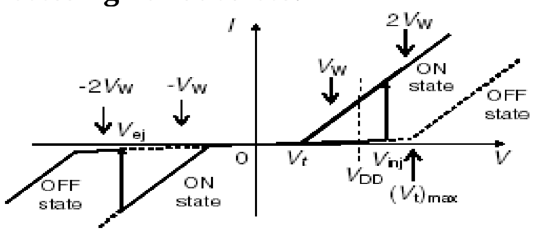
Pin placement and addressing:



+ F_{CMOS} is half pitch of CMOS sub-system.
+ F_{nano} is half pitch of nanowires.
+ β is a dimensionless factor.
+ Pin arrangement as square array with each side $2\beta F_{CMOS}$.
+ α is the nanowire-crossbar angle = arcsin (F_{nano}/F_{CMOS}).

There was a small discussion regarding the purpose of turning the crossbar to an angle. There were two answers to this question. First, by turning to an angle, it creates more flexibility in alignment, which is very important in a nanoscale system. Second, according to Mickey's opinion, these angled-contacts would create a higher density of cross-point contacts between nanowires and CMOS cells.
Also, how did they fabricate nanowires on top of CMOS sub-system?
This work shown here was just done on simulation, not yet practically implemented. Mickey thought it could be done by means of nano-imprinting.

Accessing nanodevices:



Applying +2Vw or -2Vw will exceed corresponding switching threshold, while half selected devices are not disturbed.
Advantages of CMOL:
Nanowire formation with lack of precise alignment is allowed.
Fabrication and self-assembly is less challenging.
Disadvantages of CMOL:
Defect tolerant circuits are required. Since nanowires are so small, and fabrication is not yet feasible, there will be huge amount of defects.
Semiconductor transistors' fabrication is almost perfect, with very high yield, so it is more competitive than CMOLs.
Application of CMOL:
CMOL memories:

- Nanodevice: single bit memory cell.
- CMOS sub-system: coding, decoding, sense amplifier, line driving.

Dealing with defect tolerance:
Two techniques:

- Memory reconfiguration
- Error correction: Hamming code.

Optimized total chip area per useful bit:
Hybrid CMOL memories' yield is not as high as CMOS memories' yet.
CMOL has higher density → higher error rate.
CMOL FPGA:
Reconfiguration is very important technique for dealing with defective nanodevices.
FPGA based CMOL structure is important.

FPGA (PLA and LUT) structure has the CMOS overhead problem.

Architecture:

For FPGA application: $\alpha = 45$ degrees.

Input-output of a CMOS cell connection: Pin-nanowire-nanodevice-nanowire-pin.

CMOL Cells:

Two pass transistors, an inverter, which is connected to nanowire/nanodevice subsystem via two pins.

Reconfiguration:

Needed to increase the fabrication yield.

Reconfigurable computer architectures allow one to locate bad components first and then perform optimum configuration (i.e. Teramac computer)

Reconfiguration Algorithm:

- Quasi-optimal
- Exhaustive search

These are not practical approaches.

Simple linear time algorithm:

- First stage: Map circuit on defect-free CMOL fabric.
- Second stage: reconfigure around defective components.

After mapping:

Assume one defect type: absence of nanodevices at certain cross-points, and defects are randomly distributed.

Algorithm: Each gate from a cell with bad input/output connection is moved to a new cell while keeping input/output gates in fixed positions.

Implementing the algorithm:

There was a discussion regarding the illogical sounding implementation of this reconfiguration algorithm. Prof. Bahar pointed out that waiting until the second stage to reconfigure did not make sense. The reconfiguration might as well be done before the mapping. How can we locate defective region to do the swapping? This algorithm seems to be illogical in that sense.

What if there are several positions for gate swaps?

Prioritize each position based on providing smaller interconnect length.

Kogge-Stone Adder:

Integer parallel-prefix adder.

Most regular adder structure → easily mapped into CMOL FPGA fabric.

Logic depth: number of logic levels in the critical path.

Smaller r' (connectivity domain radius) → larger logic depth → larger number of CMOS cells.

Before reconfiguration: Poor defect tolerance, yield goes down rapidly, q-bad nanodevice fraction is low.

After reconfiguration: q increased dramatically ($q=0.5$).

Full Crossbar:

General configuration of CMOL circuit.

Assigning each pair by using greedy algorithm.

Path creation for the I/O pairs.

Routing algorithm: Select vertical size (m) of array, smallest value for m, calculate the minimum value of m and the logic depth.

Results:
Crossbar is less defect-tolerant than adder.
As (r-r’) increases, defect tolerance of crossbar becomes better because the repair region is larger.
Performance: leaking static power always happens.

Comparison between CMOS FPGA and CMOL FPGA:

CMOS FPGA	CMOL FPGA
Fnano= 8mm	Fcmos=32nm
Area = 70000μ^2	110 μ^2
Delay = 5.1ns	1.3ns
500 times larger delay product.	Power acceptable.

CMOL Crossnets: Neuromorphic networks
In each Crossnet somas are implemented in CMOS subsystem.
Mutually perpendicular nanowires of CMOL crossbar implement axons and dentries allowing one cell to be connected to unlimited number M of other cells. This parallelism gives flexibility to Crossnets.
Two Crossnet species:

- FlossBars: layered topology.
- InBars: Interleaved structure.

Conclusion:
Elif thought the conclusion is weak “There is a chance for development hybrid CMOL ICs”.
Several application areas: Terabit scale memories, Reconfigurable digital circuits, Mixed signal neuromorphic networks.
Challenges: Development of high yield techniques.