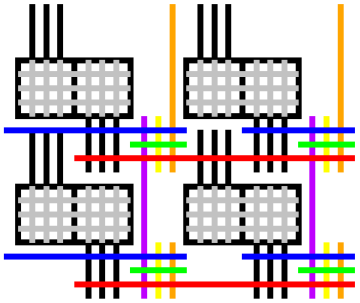
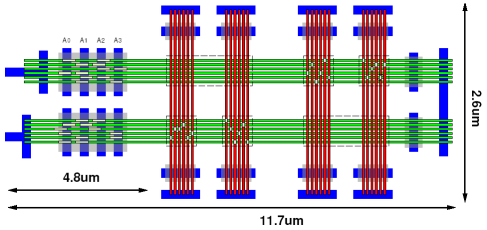


Nanowire-Based Computing Systems



André DeHon

andre@cs.caltech.edu

In collaboration with

Helia Naeimi, Michael Wilson, Charles Lieber, Patrick Lincoln, and John Savage

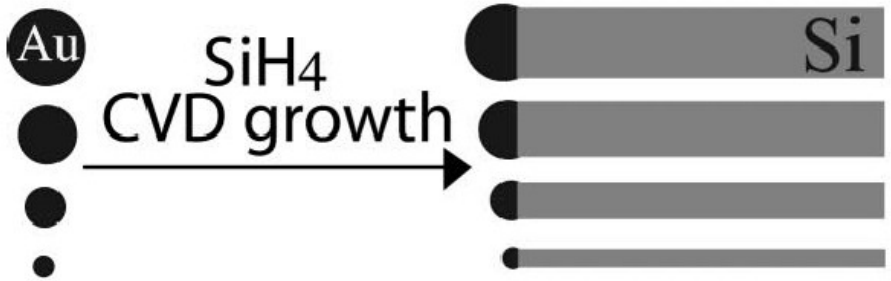
Focus Challenge

- How build programmable logic from nanowires and molecular-scale switches?
  - With regular self-assembly
    - Only have statistical differentiation
  - With high defect rates

Building Blocks

Semiconducting Nanowires

- Few nm's in diameter (*e.g.* 3nm)
  - Diameter controlled by seed catalyst
- Can be microns long
- Control electrical properties via doping
  - Materials in environment during growth
  - Control thresholds for conduction

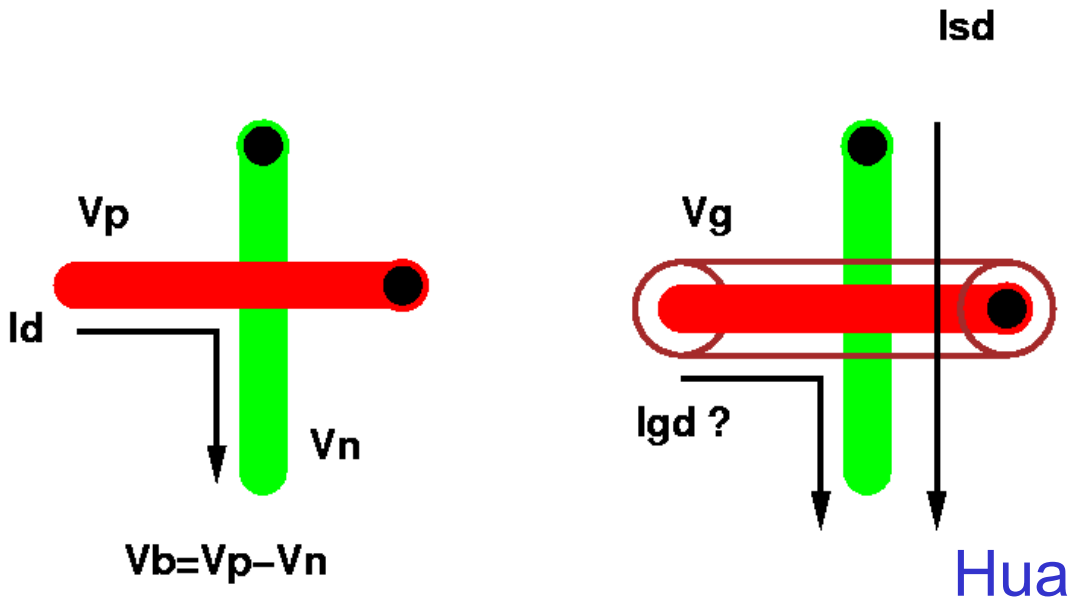


From:
Cui…Lieber
APL v78n15p2214

Devices

Doped nanowires give:

Diode and FET Junctions

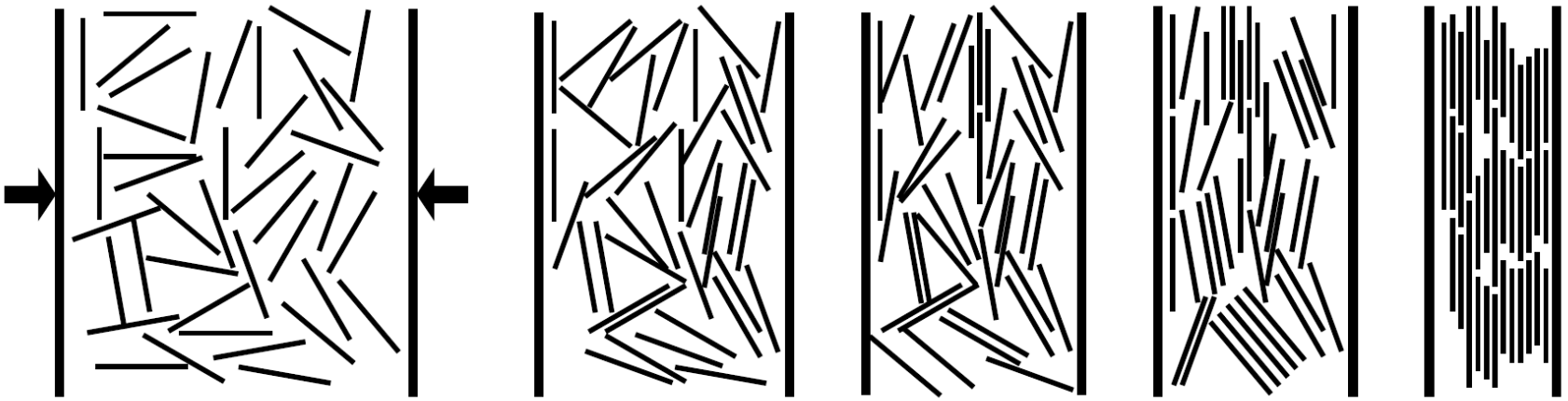


Cui…Lieber
Science 291 p851

Huang…Lieber
Science 294 p1313

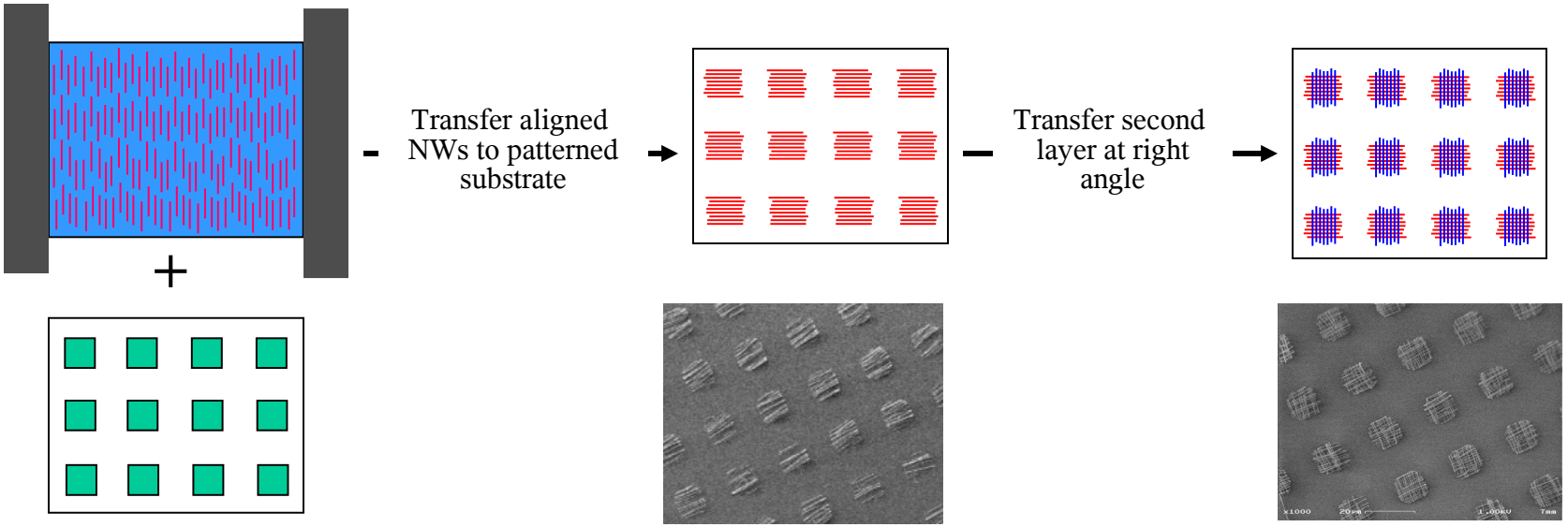
Langmuir-Blodgett (LB) transfer

- Align Nanowires



Langmuir-Blodgett (LB) transfer

- Can transfer tight-packed, aligned SiNWs onto surface
  - Maybe grow sacrificial outer radius, close pack, and etch away to control spacing

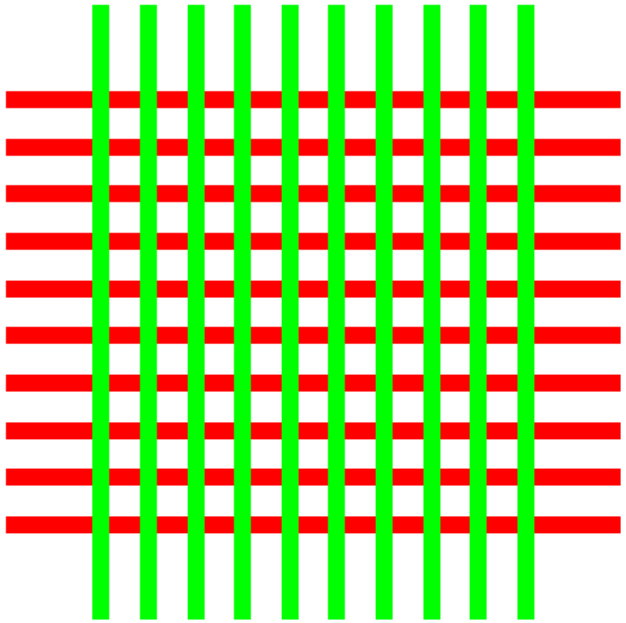


Whang, Nano Letters 2003 v7n3p951

Homogeneous Crossbar

- Gives us homogeneous NW crossbar
  - Undifferentiated wires
  - All do the same thing

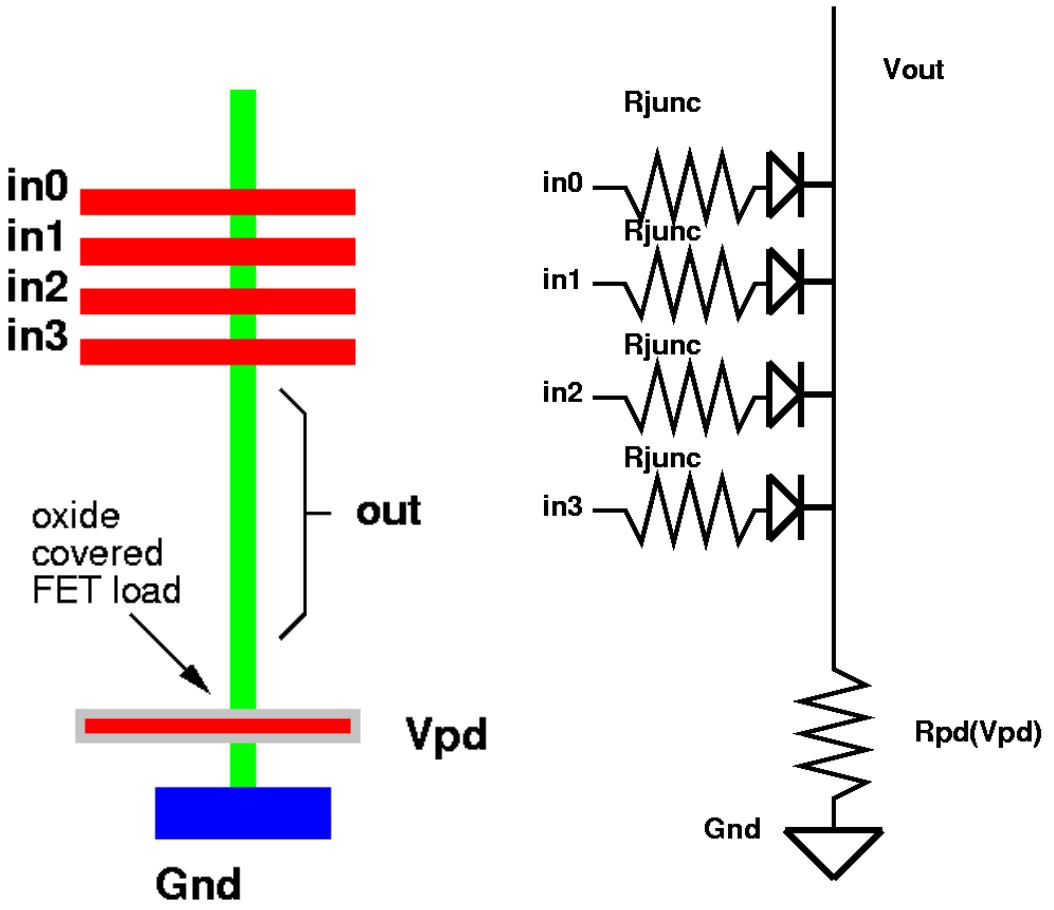
- Can we build arbitrary logic starting with regular assembly?



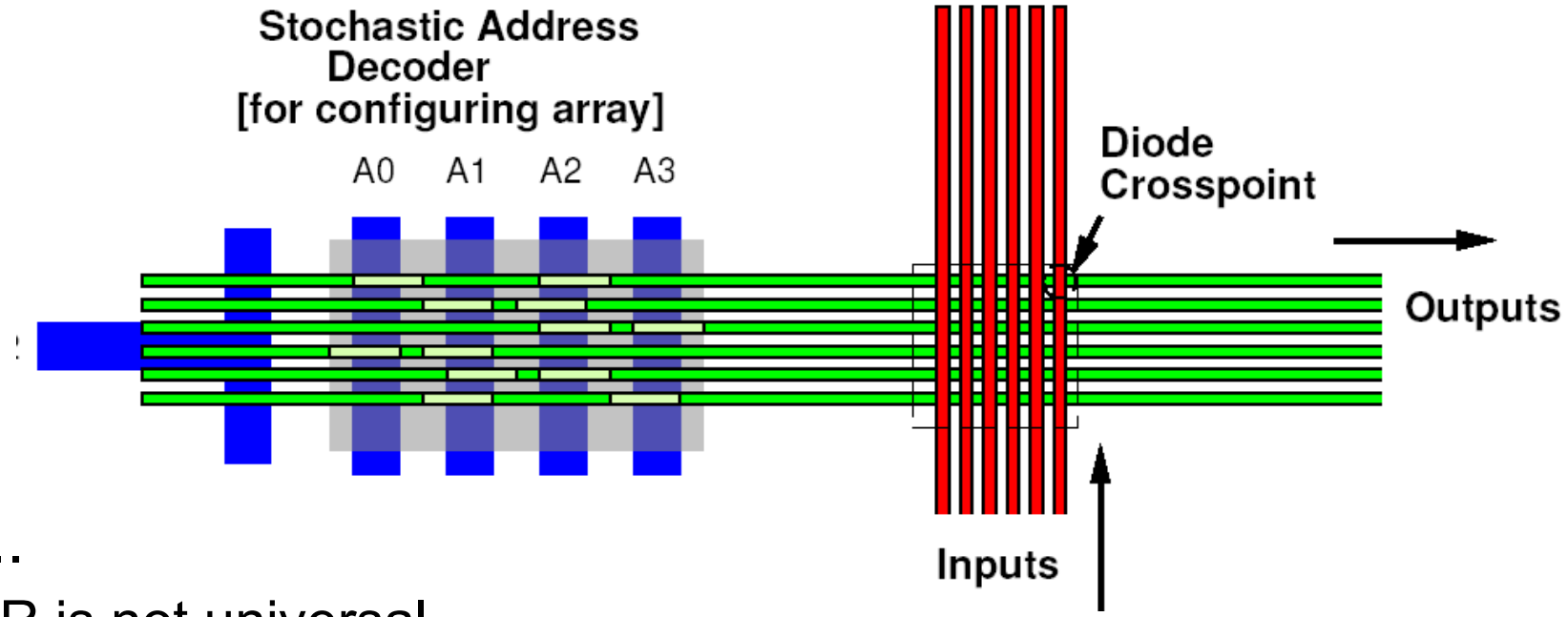
…on to Logic…

Diode Logic → Wired OR

- Arise directly from touching NW/NTs
- Passive logic
- Non-restoring
- Non-volatile Programmable crosspoints



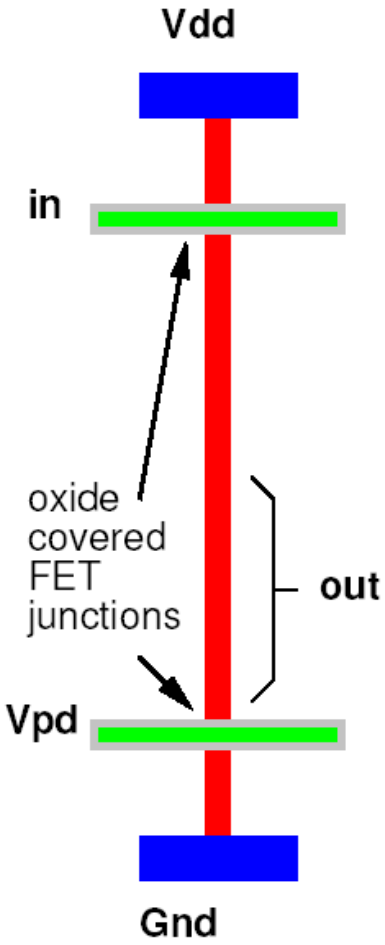
Use to build Programmable OR-plane

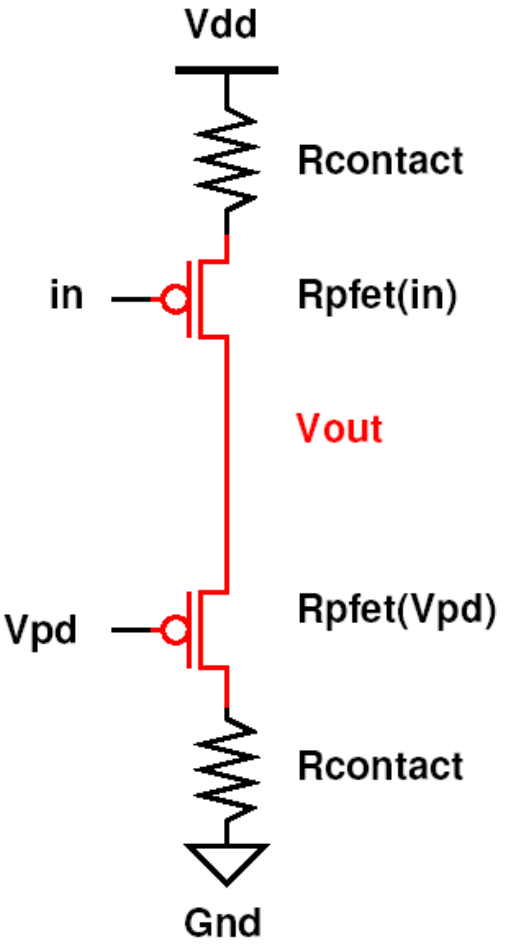


- But..
  - OR is not universal
  - Diode logic is non-restoring → no gain, cannot cascade

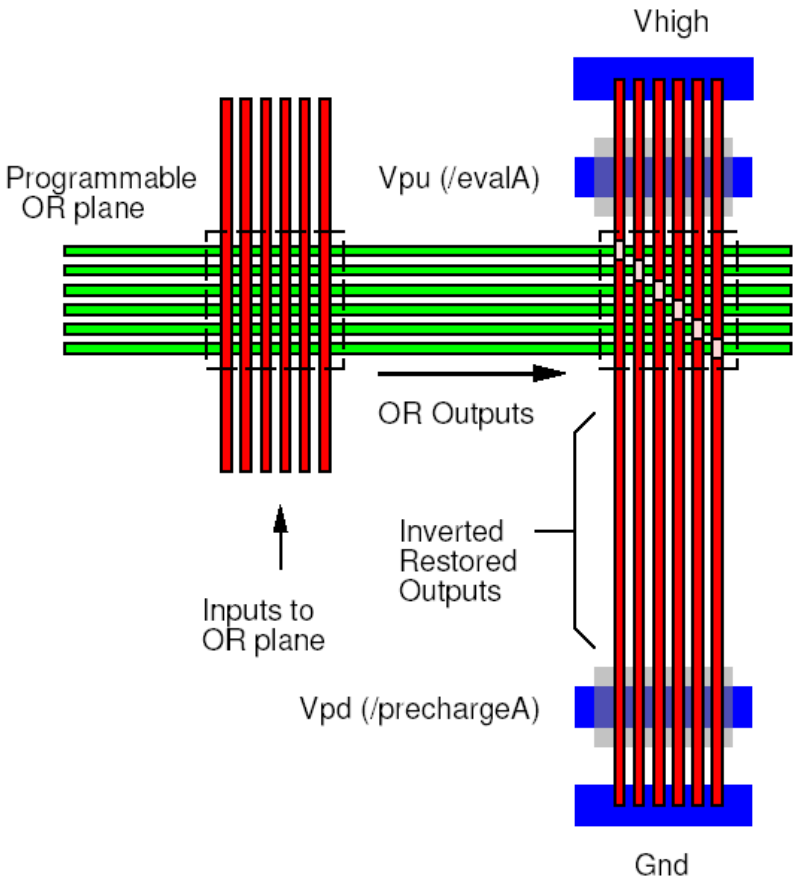
PMOS-like Restoring FET Logic

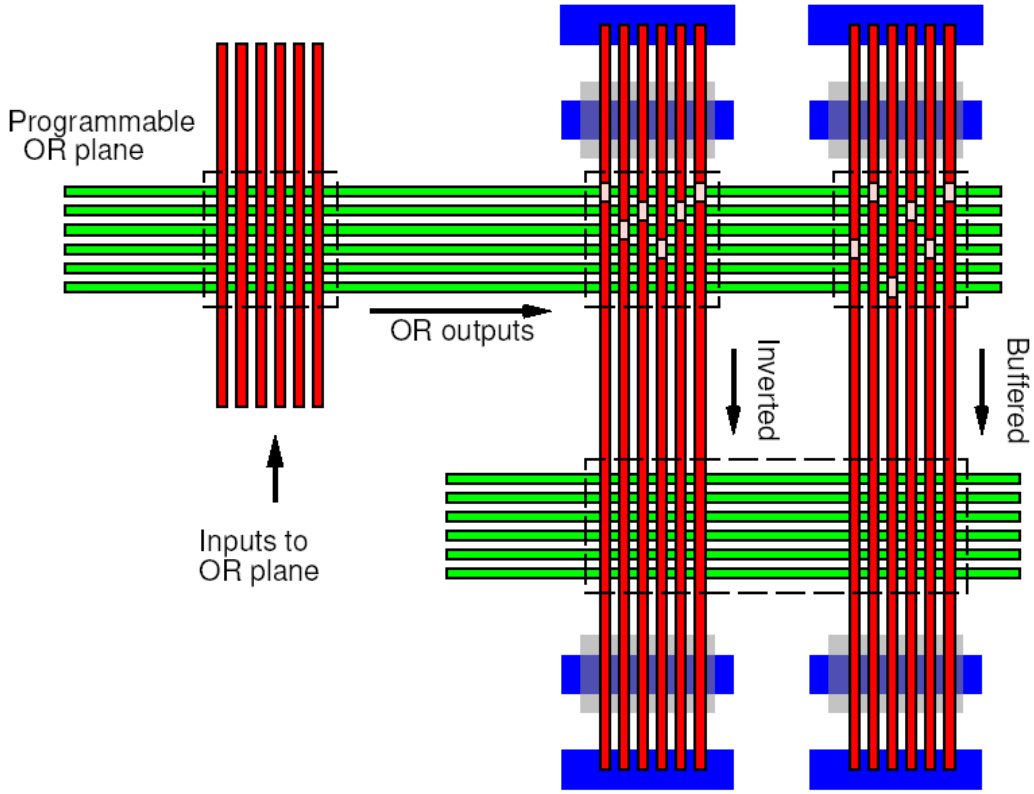
- Use FET connections to build restoring gates
- Static load
  - Like NMOS (PMOS)
- Maybe precharge



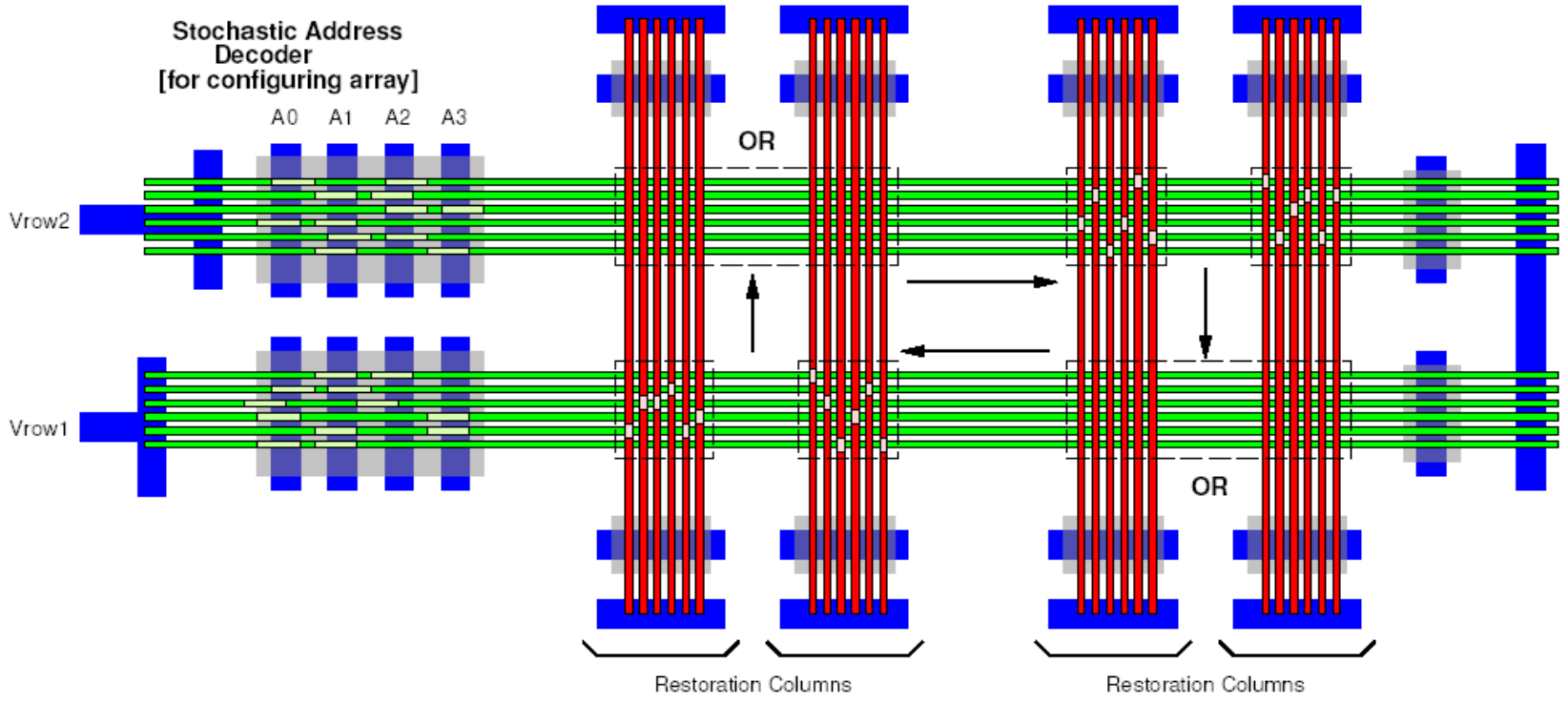


Restoration Array





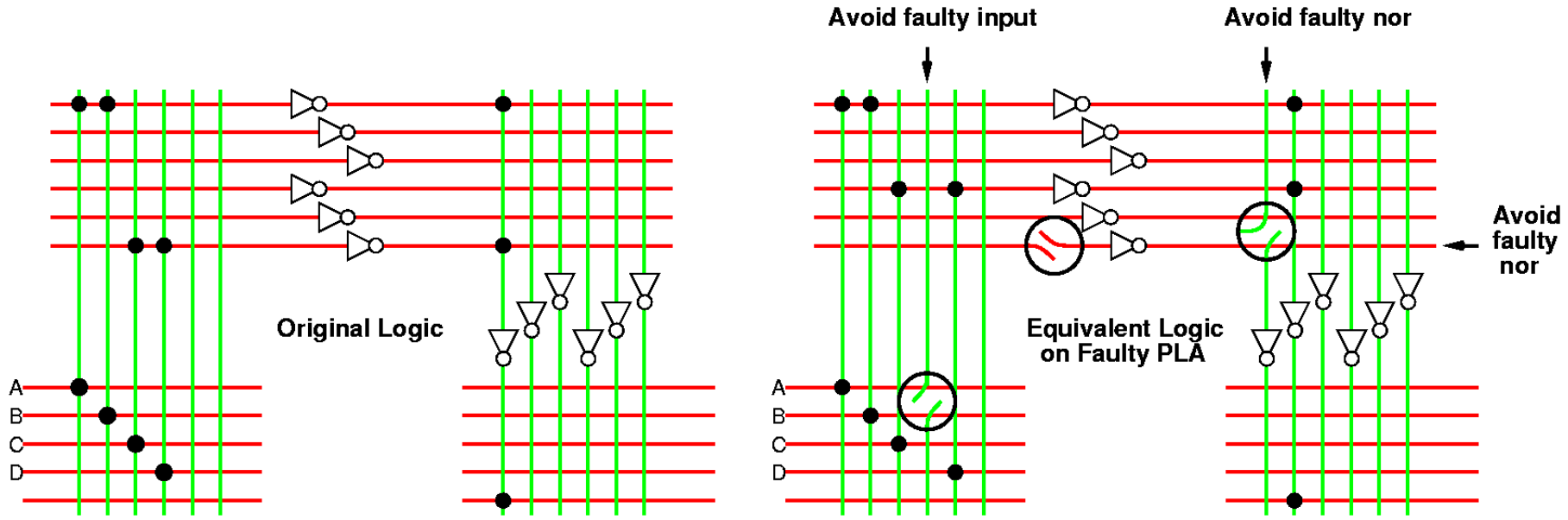
Simple Nanowire-Based PLA



NOR-NOR = AND-OR PLA Logic

FPGA 2004

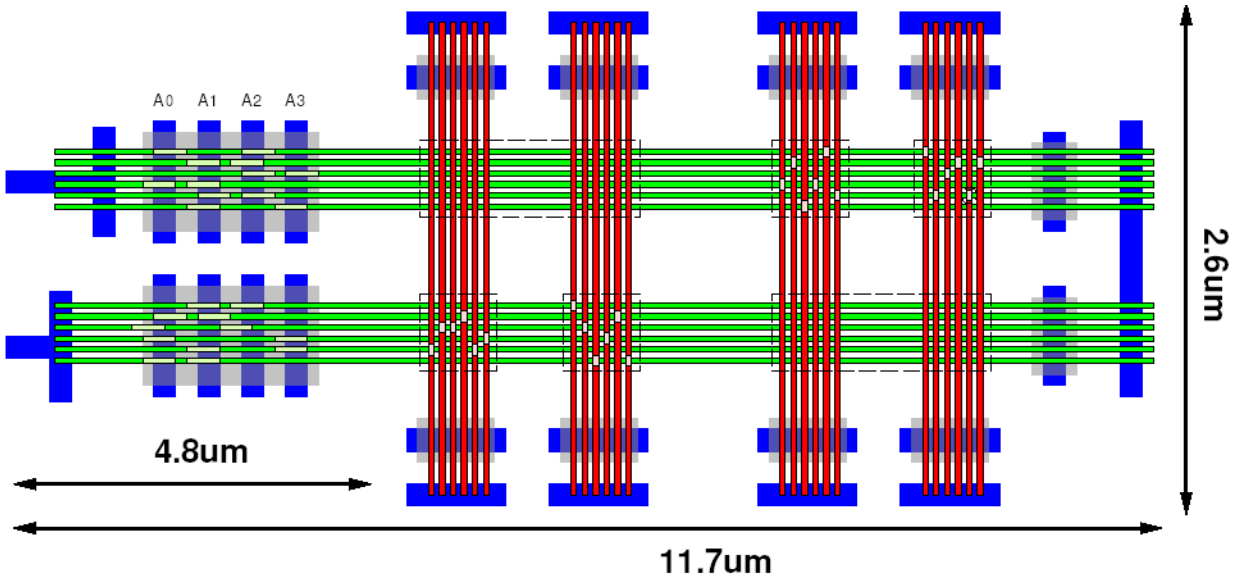
Defect Tolerant



All components (PLA, routing, memory) interchangeable;
Have M-choose-N property
Allows local programming around faults

Simple PLA Area

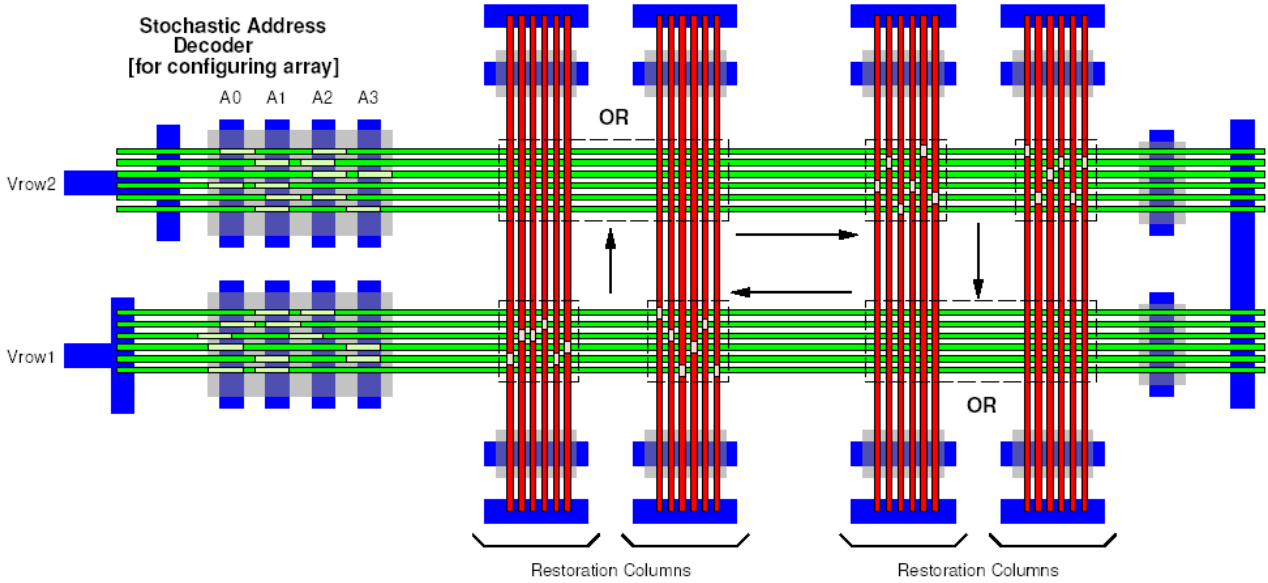
- 60 OR-term PLA
  - Useable
- 131 raw row wires
  - Defects
  - Misalign
- 171 raw inverting wires
  - Defects
  - Statistical population
- 60M sq. nm.
  - (2 planes)



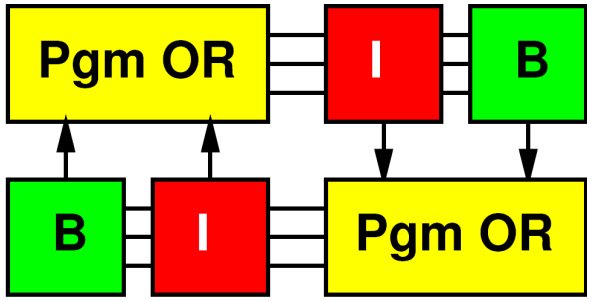
90nm support lithography;
10nm nanowire pitch

Scaling Up

Scaling Up



- Large arrays are not viable
  - Not exploit structure of logic
  - Long Nanowires tend to break
  - Long Nanowires will be slow



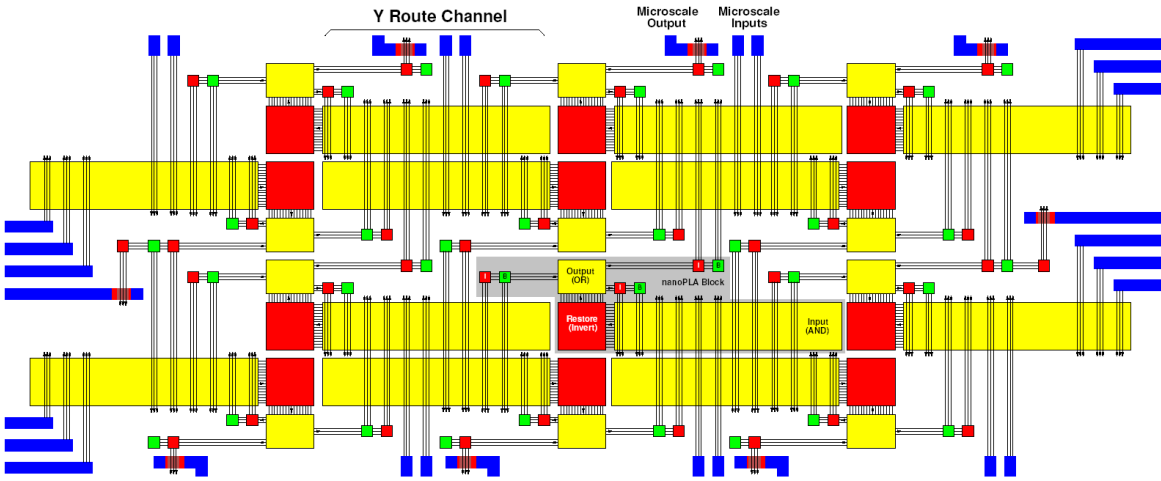
Complete Substrate for Computing

- Know NOR gates are universal
- Selective inversion
- Interconnect structure for arbitrary routing

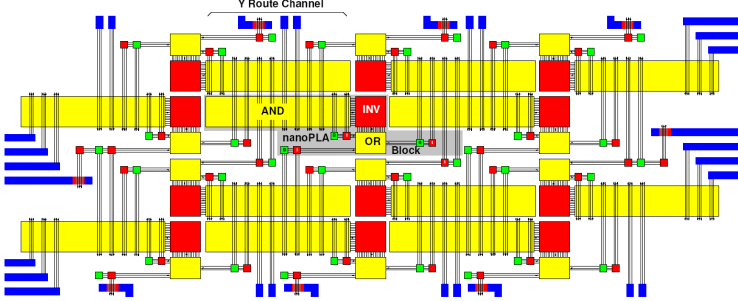
➔Can compute any logic function

- Programmable structure similar to today's FPGAs

- Can combine with nanomemories



Interconnected nanoPLA Tile



inputs

inputs

Input Wired–OR Region (AND Plane)

Internal Inversion and Restoration

Feedback Column

Buffer

Invert

Output Wired–OR Region (OR Plane)

Up Output

Overlapping Segment

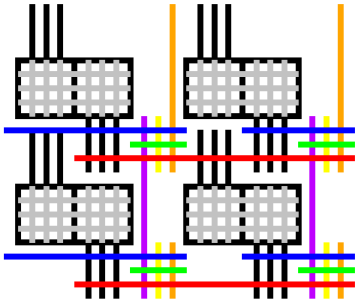
Down Output

Buffer

Invert

Buffer

Invert

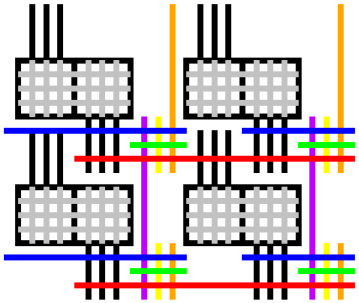


Summary

- Can engineer designer structures at atomic scale without lithographic patterning
- Must build regular structure
  - Amenable to self-assembly
- Can differentiate
  - Stochastically
  - Post-fabrication programming
- Sufficient building blocks to define universal computing systems
- Reach or exceed extreme DSM lithography densities
  - With modest lithographic support

Additional Information

- <http://www.cs.caltech.edu/research/ic/>
- <http://www.cmliris.harvard.edu/>



CMOL vs NASICs

T. Wang
University of Massachusetts, Amherst

September 29, 2005

Agenda

- Problems of pure nanoelectronics
- Problems of Nano-CMOS interface
- Advantages
- Problems
- Conclusions on CMOL

Why not Pure Nanoelectronics?

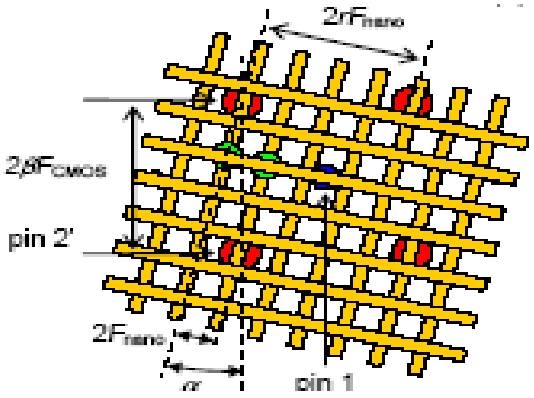
- Problems of Nanodevices
  - Simple devices: Limited functionality
  - Complex devices: Vulnerable to temperature
  - Voltage gain<1, signal attenuation
- Hybrid Architecture
  - Nanodevices + CMOS circuits
  - Interface between CMOS and nanodevices

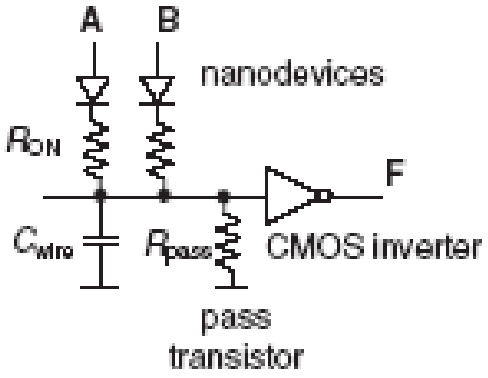
CMOS/Nano Interface

- Stochastic assembling:
  - Limited connectivity
    - Long time to program
  - Resistance of interface affects performance
  - Fabrication issue

CMOL overview

- **C**MOS / nanowire / **MOL**ecular hybrids
- Uses combination of Micro – Nano
  - Nano implements regular blocks (ie memory)
  - CMOS used for logic, decoder and driver circuits
- Best suited for memories and cell-based FPGA d

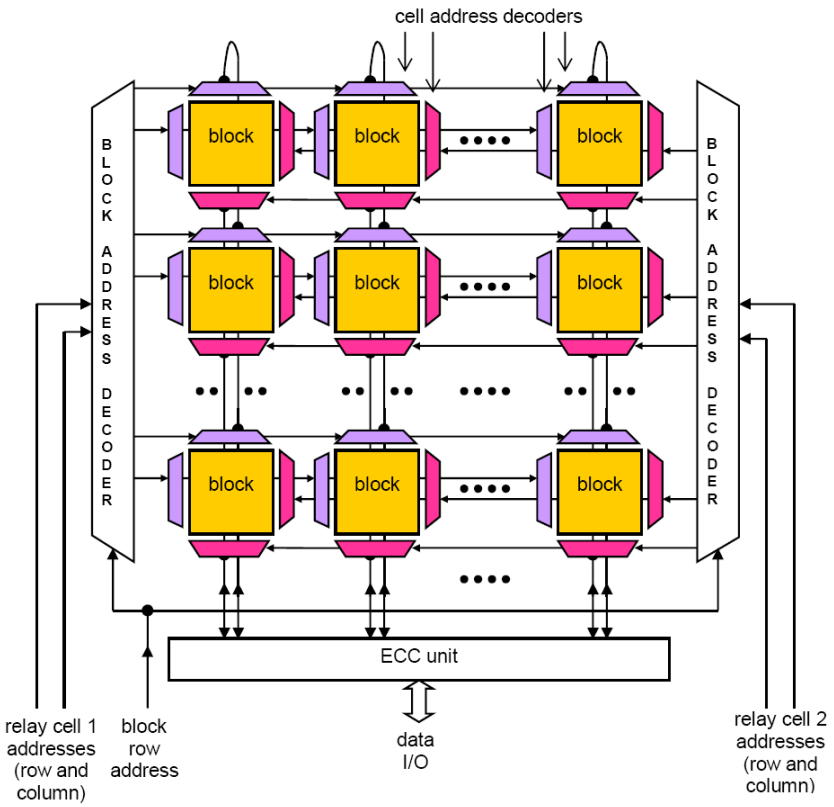




CMOS/Nano Interface in CMOL

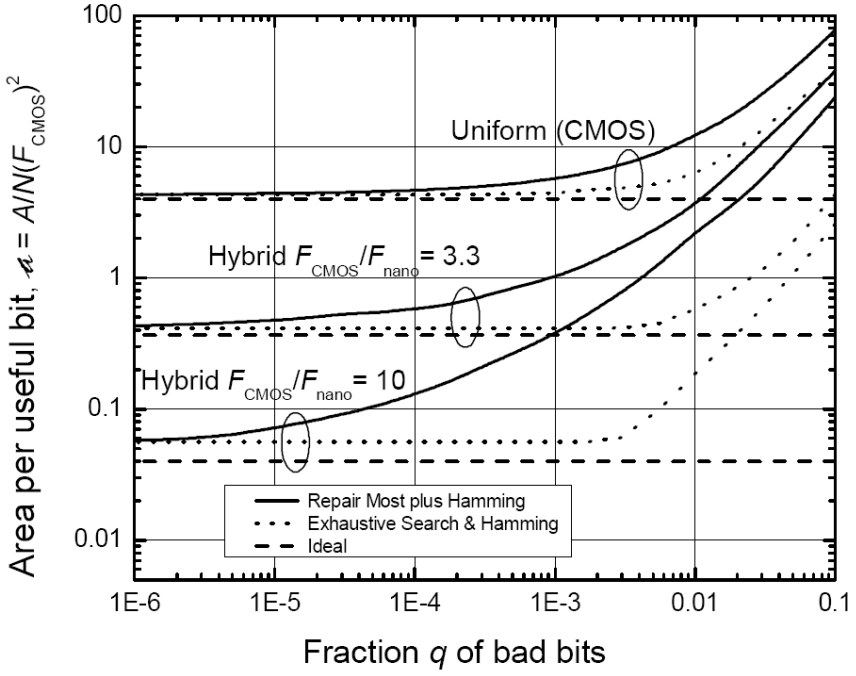
- The key idea is to tilt nanoarray at a certain angle **α**
- Each nanowire can fall on a certain CMOS pin (for addressing)

CMOL Memory



- Nanodevices for memory cell
- CMOS for coding, decoding, sensing, I/O
- Fault tolerance:
  - Reconfiguration
  - Hamming codes

Area Efficiency of CMOL Memory



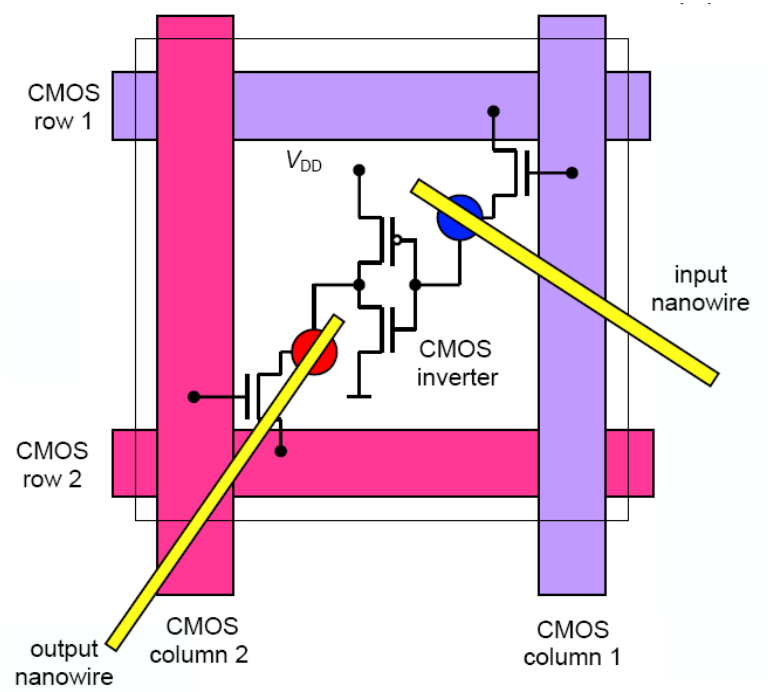
The area per useful bit as a function of single bit yield, for hybrid and purely memories.

- CMOL memory is denser than CMOS memory
- Reconfiguration improves fault-tolerance
- High yield of single bit is still required.

CMOL Logic Circuits

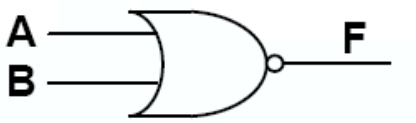
- LUT based FPGA: address decoding and sensing require CMOS circuits. For small LUT, CMOS overhead is great.
- PLA based FPGA:
  - The same problem as LUT-based FPGA
  - Power dissipation (10KW/cm^2)
- CMOL FPGA: reconfigurable regular structure

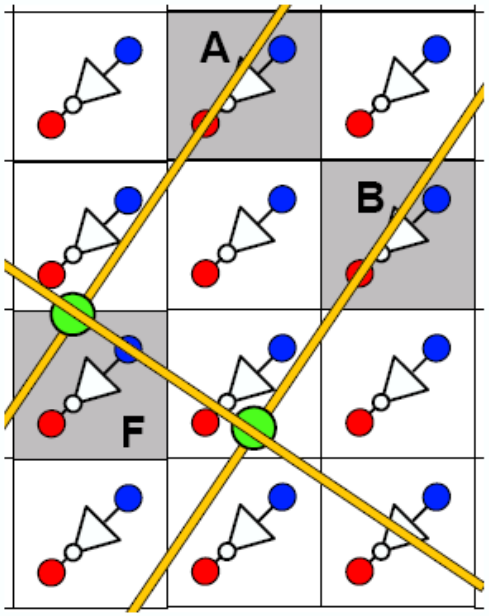
CMOL FPGA – A Single Cell



- Logics in CMOS
- Interconnections in Nanowires

CMOL FPGA – A NOR Gate



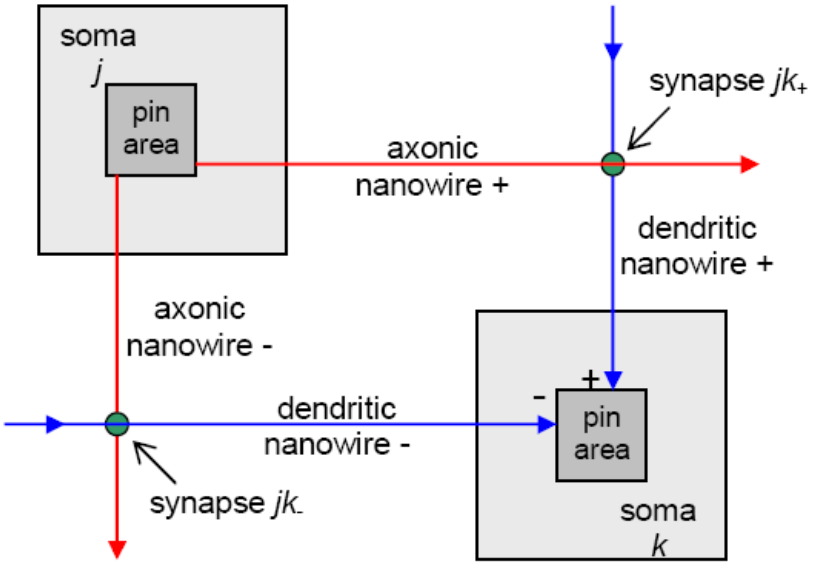


- Wire-OR logic on nanowires?
- NOR is enough for arbitrary logic functions

CMOL CrossNets: Neuromorphic Network

- CrossNet is an architecture of neuromorphic network
  - Somas (in CMOS): Neural cell bodies
  - Axons and dendrites (mutually perpendicular nanowires)
  - Synapses (switches between nanowires): control coupling between axons and dendrites

Schematic of CrossNet



- Light-gray squares in panel (a) show the somatic cells as a whole.
- Signs show the somatic amplier input polarities.
- Green circles denote nanodevices forming elementary synapses.

Conclusions

- CMOL for:
  - Memories: reconfiguration, overhead of CMOS
  - FPGAs: really denser than CMOS?
  - Neuromophic networks: no comments
- Density, delay, power dissipation
  - Delay should be better than NASIC, since it use a new layer for CMOS.
  - Power dissipation is hard to compare with NASIC.