On the Use of Nonvolatile Programmable Links for Restructurable VLSI*

J.I. Raffel
MIT Lincoln Laboratory, Lexington, Massachusetts

VLSI — Objectives, Problems and History

There seems to be general agreement that VLSI implies the fabrication of digital logic circuits having minimum feature definition at least as small as one micron and levels of integration of greater than one hundred thousand gates per package.

Associated with VLSI are processing and device performance limitations which have been identified and extensively studied as well as a variety of architectural and digital design strategies which have received only cursory treatment. Much attention has been given to the use of x-ray, electron-beam and other lithographic techniques for achieving submicron device geometries. Ion implantation and the scaling laws for minimizing short channel effects in FET's have been widely discussed, but relatively little has been proposed in the way of solutions to the yield problems raised by the combination of very small devices and such high levels of integration.

Specifically, past improvements in circuit integration have almost universally been predicated on brute force improvements in lowering defect density to achieve economically viable yields of perfect devices. There are reasons to believe that with VLSI this is neither a possible nor necessarily desirable objective. The structures we propose to fabricate are now so complex and catastrophic defect size so small that, on the one hand, there is a great likelihood that we can never make them perfect and, on the other, the possibility that, for the first time, we can afford the degree of redundancy necessary to provide fault-tolerant operation on a significant scale. The effective use of redundancy to achieve acceptable yield levels introduces difficult problems in device implementation, layout, logic-design and testing.

We choose at the outset to define as an appropriate goal not simply the utilization of higher gate density to proceed from LSI to VLSI but the

---

*This work was sponsored by the Department of the Air Force.

The views and conclusions contained in this document are those of the contractor and should not be interpreted as necessarily representing the official policies either expressed or implied, of the United States Government.

simultaneous expansion of chip size to whole wafer technology to provide a much greater increase in overall chip complexity than could be achieved by gate shrinkage alone. The decision to compound these individual contributions to chip capacity is based on the belief that once a level of complexity is reached which requires restructurability, old thresholds are erased and the tradeoffs between chip size and yield are fundamentally altered.

Restructurable Logic

The ability to reconfigure or restructure the logic of a monolithic integrated circuit may be used to accomplish three different sets of objectives which are sometimes confused.

It is proposed that defect avoidance, the first of these, is probably essential for the levels of integration and resolution required for VLSI. This category may be further subdivided depending on whether restructuring is accomplished only at wafer probe or whether faults which develop in the field are also correctible.

The ability to restructure for functional specialization is a way of achieving customized structures with standard modules thus reducing the long turn-around times due to design, layout, fabrication, and testing associated with all-custom designs.

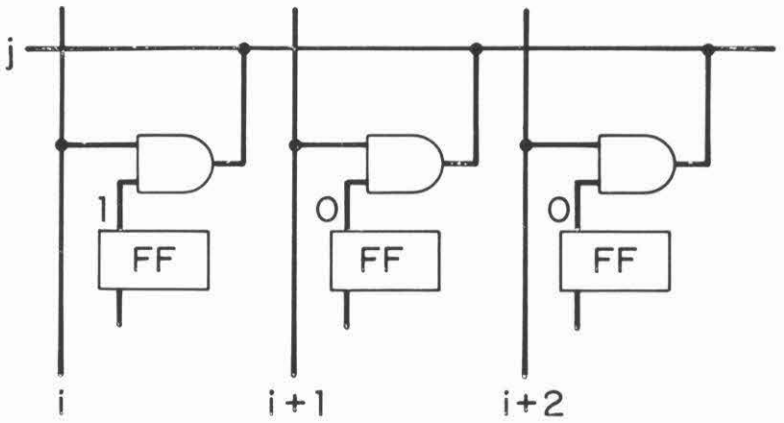
A third category utilizes dynamic reconfiguration to alter the system configuration during the running of a problem to increase machine efficiency by providing better utilization of resources to meet the changing computational requirements of the application.

Each of these objectives presents different requirements relative to diagnostic techniques, interface requirements, speed and frequency of restructuring, implications of volatility and permissible overhead.

Programmable Links — Volatility vs. Nonvolatility

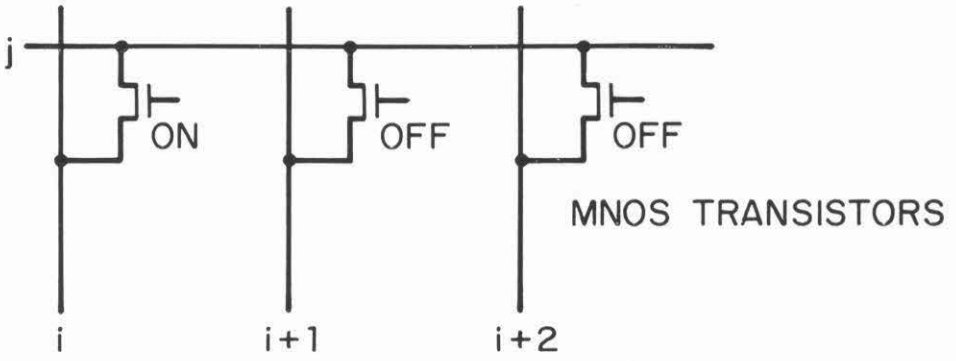
It is possible to excise faulty sections of logic and patch in healthy ones by switching path connectivity under the control of stored bit patterns which are alterable on the basis of test and diagnostic results. The approaches to this capability vary considerably depending on whether the control store is volatile or nonvolatile. In the former case testing and reconfiguration must be initiated each time power has been removed from the part in question. In general there is also significantly more area associated with volatile link control than with a nonvolatile structure, although the former has the advantage of using basic components which are identical to those used in the internal logic structure.

Figure 1 shows an example of a volatile programmable link and Fig. 2 shows an example of a non-volatile programmable link. For concreteness an MNOS structure[1] is assumed but nonvolatile operation has been achieved with other devices.[2,3] In addition to the larger area required, the volatile FF link control requires significant standby power or, if a dynamic storage cell



VOLATILE PROGRAMMABLE LINKS

FIG. 1



NONVOLATILE PROGRAMMABLE LINKS

FIG. 2

is used, periodic refreshing.

The general scheme for using nonvolatile programmable links to restructure a VLSI circuit is shown in Fig. 3. A typical "cell" which may be of MSI to LSI complexity is connected to its environment by horizontal lines fabricated on second level metal and vertical lines on first. In order to connect any output node to any input node, it is only necessary to connect three links: one output link, one crossover link, and one input link, thereby forming a continuous path from output to input.

Figure 4 shows an array of such cells and busses forming a chip along with estimates of the numbers of gates, cells, busses and links for a VLSI complexity varying between 10^5 and 10^7 gates. These numbers are purely speculative and are only meant to help focus on some of the implementation problems and architectural implications of this approach. Although busses are shown schematically to run the full chip length, it is proposed that these would, in fact, be segmented, providing a mix of both local and express runs for greater flexibility in routing in much the same way as has been proposed for Programmed Logic Arrays.[4]

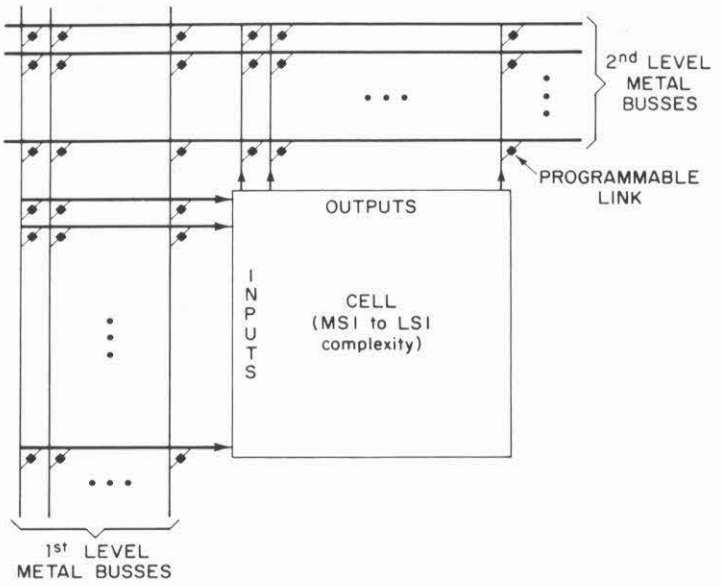
Nonvolatile Programmable Link Technology

The feasibility of programmable link control depends critically on the ability to access a sufficiently large number of links in order to provide a flexible reconfiguration capability from a practical number of extra programming nodes. A decoding tree structure must be developed that provides the necessary translation. Preliminary estimates indicate that the overhead associated with such a system may not be unreasonable. Estimates of area requirements for links themselves also seem to represent an acceptable percentage of total chip area, although there is a direct tradeoff between the link electrical conductivity and the area consumed, which requires detailed study. Figure 5 shows one proposed layout of a nonvolatile programmable link.

In order to program an NPL it is necessary to provide coincident voltages to the metal gate electrode and the orthogonal, isolated, n-silicon stripe whose intersection defines the MNOS transistor channel region. A scheme for fabricating high density, planar, isolated digit lines has been described previously.[5] A chip having 1000 cells each with 10 inputs, 10 outputs, and 20 horizontal and 20 vertical busses, would require 200 input links, 200 output links and 400 crossover links. An array of 1000 gate lines and 1000 silicon lines would then select one out of the approximate total of one million links thus requiring only 20 bits for programming control.

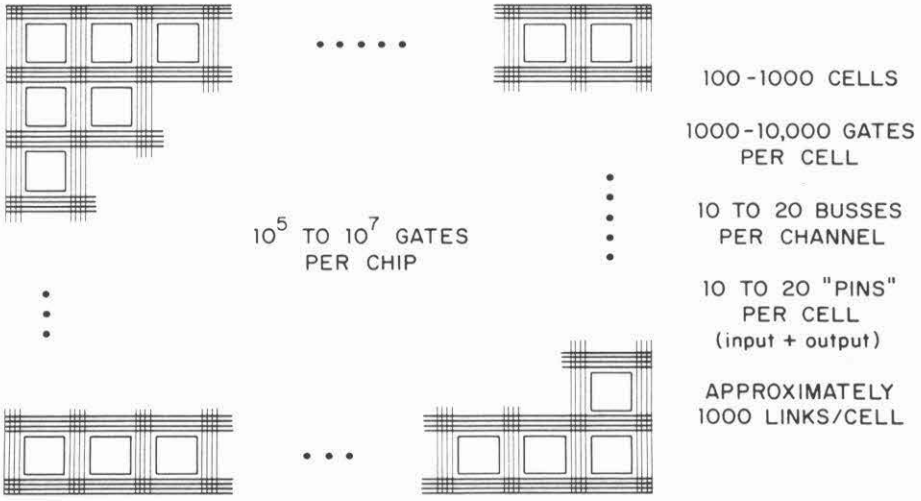
Testing, Diagnosis, Routing and Rerouting

Depending on the degree of restructurability desired the hardware for testing, diagnosing, routing and rerouting will vary considerably. In the simplest case where only defect avoidance is desired, this equipment may be located off-chip. This of course enormously simplifies the system by totally bypassing the question of how one tests the tester. Even in this simplest



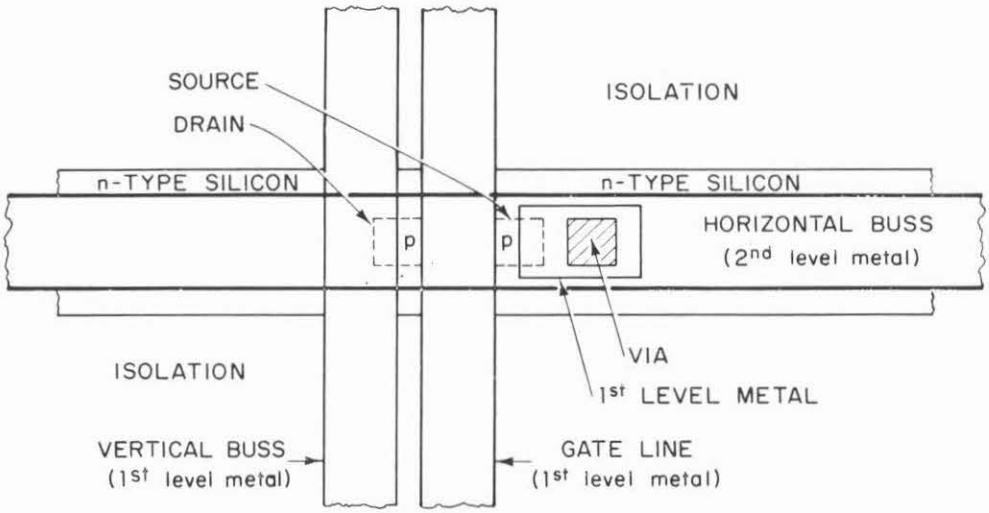
CELL, BUSSES AND PROGRAMMABLE LINKS

FIG. 3



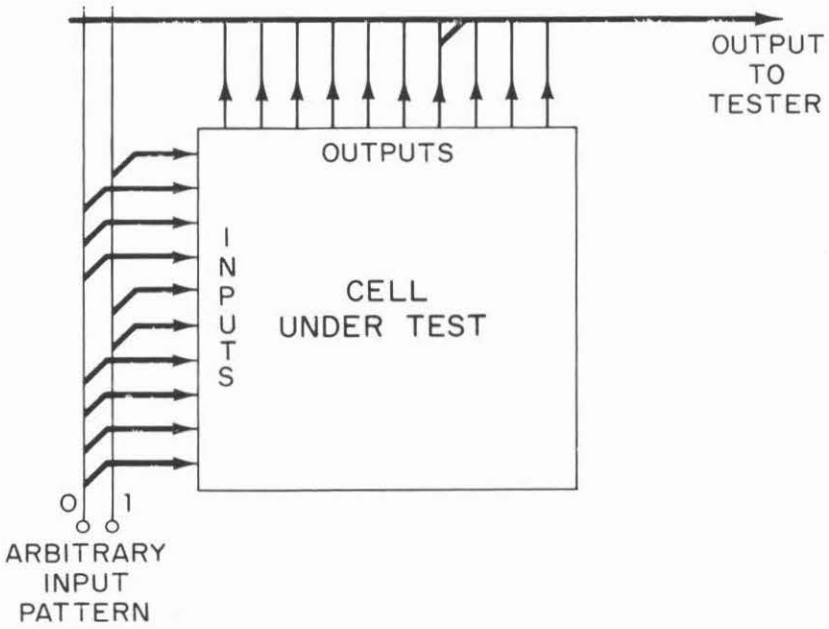
RESTRUCTURABLE VLSI USING
NONVOLATILE PROGRAMMABLE LINKS

FIG. 4



NONVOLATILE PROGRAMMABLE LINK

FIG. 5



TESTING A CELL WITH PROGRAMMABLE LINKS

FIG. 6

FABRICATION SESSION

case the problems of accessing, testing and rerouting around failed areas will require extremely sophisticated computer design aids.

One powerful feature of NPL's is that it is possible to isolate and access individual cells from the chip exterior. This is illustrated in Fig. 6 where links to a single cell are activated with all other cells disconnected from the buss. An arbitrary pattern of ONES and ZEROS may be presented at the cell inputs while a single cell output is monitored. Thus each cell may be tested singly before full chip routing is initiated.

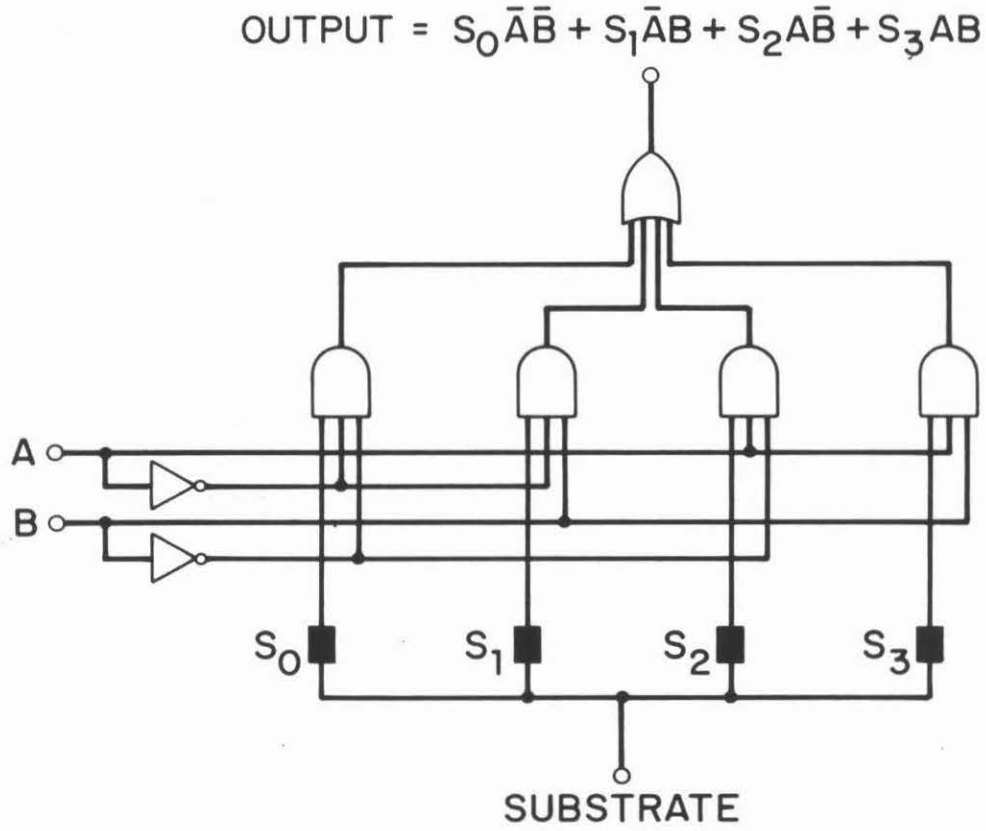
It may even be possible to allocate one region of the chip to implementation of a tester programmer (T-P). This on-chip circuit would be first tested, programmed and routed by use of an off-chip tester-programmer and, when fully configured, could then replace the external T-P for any testing or rerouting that occurs after system deployment. To insure the integrity of the on-chip T-P, classical error-correcting techniques may be employed.

Cell Design, Customization

An understanding of the spectrum of options available in cell or module design is critical to effective restructurability. These range from uniform arrays of standard cells with local customization links to totally customized cell and interconnect designs. Between these extremes lie a mix of standard cells and an array of custom cells on a regular grid structure. Each of these represents a different point on the tradeoff curve between flexibility and control simplicity. A special case must be made in the instance of memory. While, in principle, it is possible to reduce both logic and memory to basic AND/OR gates, the high usage of storage and the potential for high packing-density (because of locally connected topology) warrants providing special cells for memory.

The use of nonvolatile links to perform cell customization, as distinct from interconnect routing for defect avoidance, is illustrated by the simple example of Fig. 7. Here, local customization links are used to determine the logic function of a universal sub-cell which might be part of the larger cell shown in Fig. 3. By activating appropriate combinations of customization links S_0, S_1, S_2, and S_3 it is possible to produce the 16 truth tables for the 2 input variables A and B. Note that no extra busses are required since links are connected to the substrate.

One of the principal design parameters affecting restructurability is the degree of connectivity available for routing around failed sections. There has been considerable theoretical analysis of cellular logic with nearest neighbor connectivity, and simple buss-structured multiprocessor systems, etc. Unfortunately none of these provides much insight in treating highly connected random logic. There are basic engineering tradeoffs between module complexity and connectivity; the more complex the basic module unit the higher the gate-to-pin ratio and the smaller the control structure necessary to control all connection to its environment. Balancing this is the falloff in yield with module complexity. The use of intercell



LOCAL CUSTOMIZATION LINKS

FIG. 7

multiplexing could significantly affect this design balance. These relationships between gate count, pin out, yield and topology need to be thoroughly understood if reasonable strategies for restructurable logic are to evolve.

TABLE I

1. Routability and wiring strategies for cell interconnection — density and distribution of express and local runs
2. Cell complexity vs. link density tradeoffs
3. Universal cells for customization using local links
4. Electrical characterization of NPL's — area-speed tradeoffs
5. Development of testing procedures based on cell isolation — effects of link and interconnect failures
6. Development of test-contingent, automated routing techniques
7. Design of access decoder for testing and programming links
8. Effects of reducing device dimensions and increasing chip size

Table I, above, shows a summary of eight major areas for substantive research in the area of restructurable logic. At the present time little work has been done in any of these.* However, significant advances have been made in the fabrication technology of MNOS devices, which are now finding their way into a number of consumer markets. It is important to note the strong coupling between the device, circuit and architectural issues involved in the problem areas shown and to recognize that a unified, interdisciplinary approach is required if the promise of restructurable logic is to be realized.

---

* Since the original submission of this paper, a talk has been given at the International Telemetering Conference in Los Angeles, Cal., on 16 Nov. 1978 entitled, "Wafer integrated semiconductor mass memory," by W.A. Geideman and A.L. Solomon which describes the use of nonvolatile latches to restructure MNOS CCD memory.

References

1. D. Frohman-Bentchkowsky, "The metal-nitride-oxide-silicon (MNOS) transistor characteristics and applications," Proc. IEEE, vol. 58, Aug. 1970.

2. D. Kahng, S.J. Sundburg, D.M. Boulin, and J.R. Ligenza, "Interfacial dopants for dual-dielectric, charge-storage cells," Bell System Tech. J., vol. 53, pp 1723-1739, Nov. 1974.

3. M. Horiuchi and H. Katto, "A low voltage, high-speed alterable n-channel non-volatile memory," IEEE Proc., 1978 Tech. Digest IEDM, pp 336-339.

4. D.L. Greer, "An associative logic matric," IEEE J. Solid-State Circuits, vol. SC-11, no. 5, Oct. 1976.

5. J.A. Yasaitis, "Ion implantation of neon in silicon for planar amorphous isolation," Electronics Letters 14 (15), 460 (1978).