

# The Implementation of VLSI Systems

A.G. BELL

Xerox Corporation, Palo Alto Research Center, California, USA

## 1. Introduction

A large number of designers are creating VLSI systems. Engineers and scientists from many disciplines are applying VLSI design to their area of expertise. Many significant designs have been created including the Optical Mouse [Lyon, 1981], the Geometry Engine [Clark, 1980], the Risc Processor [Patterson and Sequin, 1980], and many others. These designers have one thing in common - they learned VLSI design from the textbook, *Introduction to VLSI Systems* [Mead and Conway, 1980].

While the textbook provided the information needed to learn design, the book alone could not have caused the cultural integration of these new methods. In order to cause the desired large scale spread of this method, certain key underlying structural, or "infrastructural", elements were needed. These included running courses, causing the creation of design environments, etc.

One of the key elements was the development of a new implementation methodology. The traditional methods were unable to scale up to support the new needs of this larger designer community. This paper will provide a detailed discussion of this new implementation methodology after first providing the background context by describing the other infrastructural components needed for the acceptance of these new methods.

## 2. Infrastructural Components

Figure 1 shows the organization of infrastructural components utilized for the evolution of the VLSI design methodology. This was an iterative process of experimentation to determine what would work and how it could be refined. Lynn Conway describes her process in *The MPC Adventures: Experiences with the Generation of VLSI Design and Implementation Methodologies* [Conway, 1981]. A brief description follows.

The first step was the generation of a simplified design methodology. Several years ago, the LSI Systems Area of Xerox Palo Alto Research Center and the Computer Science Department of Caltech collaborated to understand the essence of VLSI systems design. Through iteration and experimentation, they were able to create a methodology which can be learned quickly and utilized efficiently.

This design methodology was embodied in a collection of documents including a textbook written by Carver Mead, of CalTech, and Lynn Conway, of Xerox, *Introduction to VLSI System Design* [Mead and Conway,

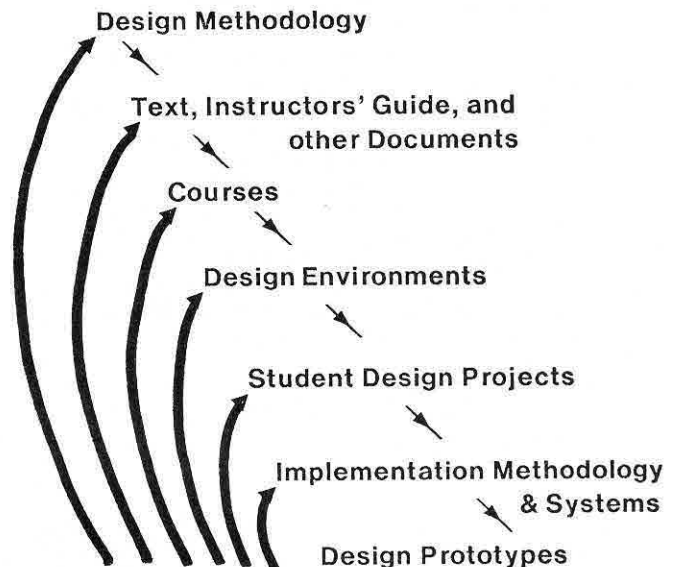


Figure 1. The Generation of a Culturally Integratable Methodologies (taken from "The MPC Adventures", Conway, 1981)

1980]. Lynn Conway also wrote an instructor's guide to aid the teaching of the material [Conway, 1979].

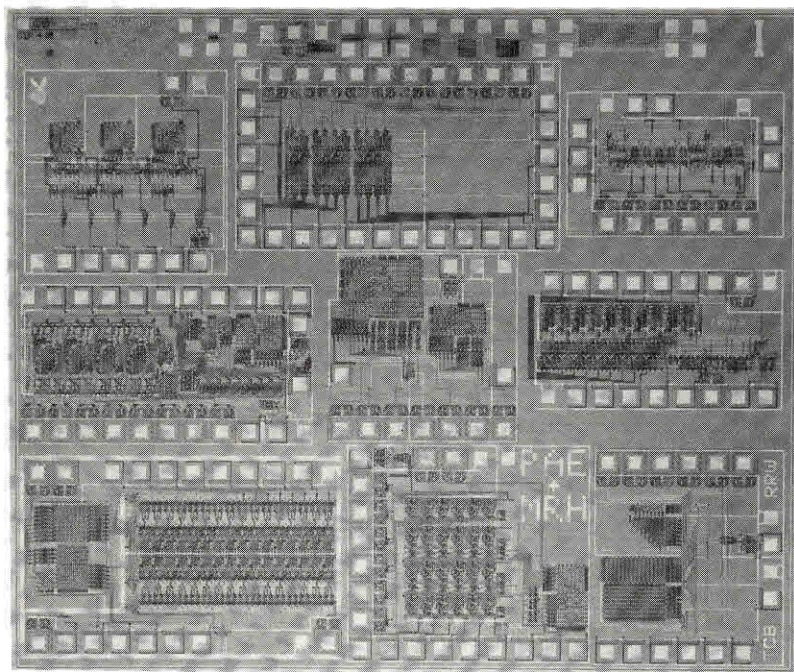
In the Fall of 1978, Lynn Conway ran an experiment to determine if the new methods and the book were really going to work. She used a draft version of the textbook to teach a one semester project-oriented course at MIT. The course was successful but illustrated many areas where refinements of the previous steps were needed.

After this debugging, a series of courses were organized and run by both Lynn and Carver to teach a group of highly motivated professors from several universities. The purpose was to teach them so they could teach their students.

These professors got courses started at their universities and interested many students in VLSI design. The students built design environments to make design creation easier and more successful.

A part of the courses was a design project. Creating a design motivated the students while forcing them to learn the material. To support this need, several approaches toward implementation were tried. Finally, a new implementation methodology was created.





**Figure 2. Photo of MPC79 Die-Type BI**

### 3. New Implementation Methodology

The existing method of fabricating one design per fabrication run did not scale up to handle the large number of designers. However, the new methodology had to be built upon the existing "Foundry" services. These vendors had the equipment and knowledge to build wafers but not to handle all the designs.

The designer needs to be shielded from the details of the fabrication process. He wants to see implementation as a process which he sends designs and receives implemented, packaged chips. There is a large amount of idiosyncratic knowledge required to interface with fabrication services. This knowledge is not captured in written documents; it is only learned by experience. Each designer should not have to know this information even if he could locate it.

The cost of an individual design prototype needs to be minimized. By sharing the silicon area between many designers, the cost per design is dramatically reduced [Conway, Bell, Newell, 1980]. The Implementation System aids this sharing by collecting designs from many designers.

The designer needs to get parts back shortly after submitting them. The turnaround time will affect the designer's motivation. If the time around is long, he will forget about the design, or do other things.

A large amount of information processing is required to generate the pattern generator tapes. The software to handle this information is complex. By centralizing this function, an economy of scale is realized.

### 4. Implementation Systems

A concept called an "Implementation System" was developed to deal with these problems. This system was created by a group of researchers at Xerox Palo Alto

Research Center. The Implementation System serves as a middleman between the many designers and the fabrication services. The function is similar to a time-shared operating system. Just as an operating system buffers the user from the obscure hardware, the Implementation System buffers the designer from the fabricators by formalizing the interfaces and capturing the knowledge.

The Implementation System consists of several sections shown in figure 3. The architecture and operation will be described.

The infrastructural role will be illustrated with "MPC79." In the Fall of 1979, twelve universities needed implementation. They had 124 designers design 84 projects to be implemented on MPC79. This was the first time an Implementation System was used.

### 5. User Interaction with an Implementation System

The Implementation System interacts with the designer via electronic mail on an electronic packet-switched network. This allows simple and standardized communication. MPC79 utilized the Defense Advanced Research Project Agency's (DARPA) ARPANET.

The initial communications are for information exchange. The designer can ask when the next run will be fabricated, what processes are supported, etc. To get this information, the designer sends electronic mail to the Implementation System's mail address. The system then parses the message, and responds appropriately. If the message is not in a form that the system can understand, a reply is returned that describes acceptably formed mail.

When the designer has completed his design, he sends an implementation request message. This message includes the design file in a standardized format - "CIF", Caltech Intermediate Form (Hon and Sequin, 1980). The Implementation System processes this request by



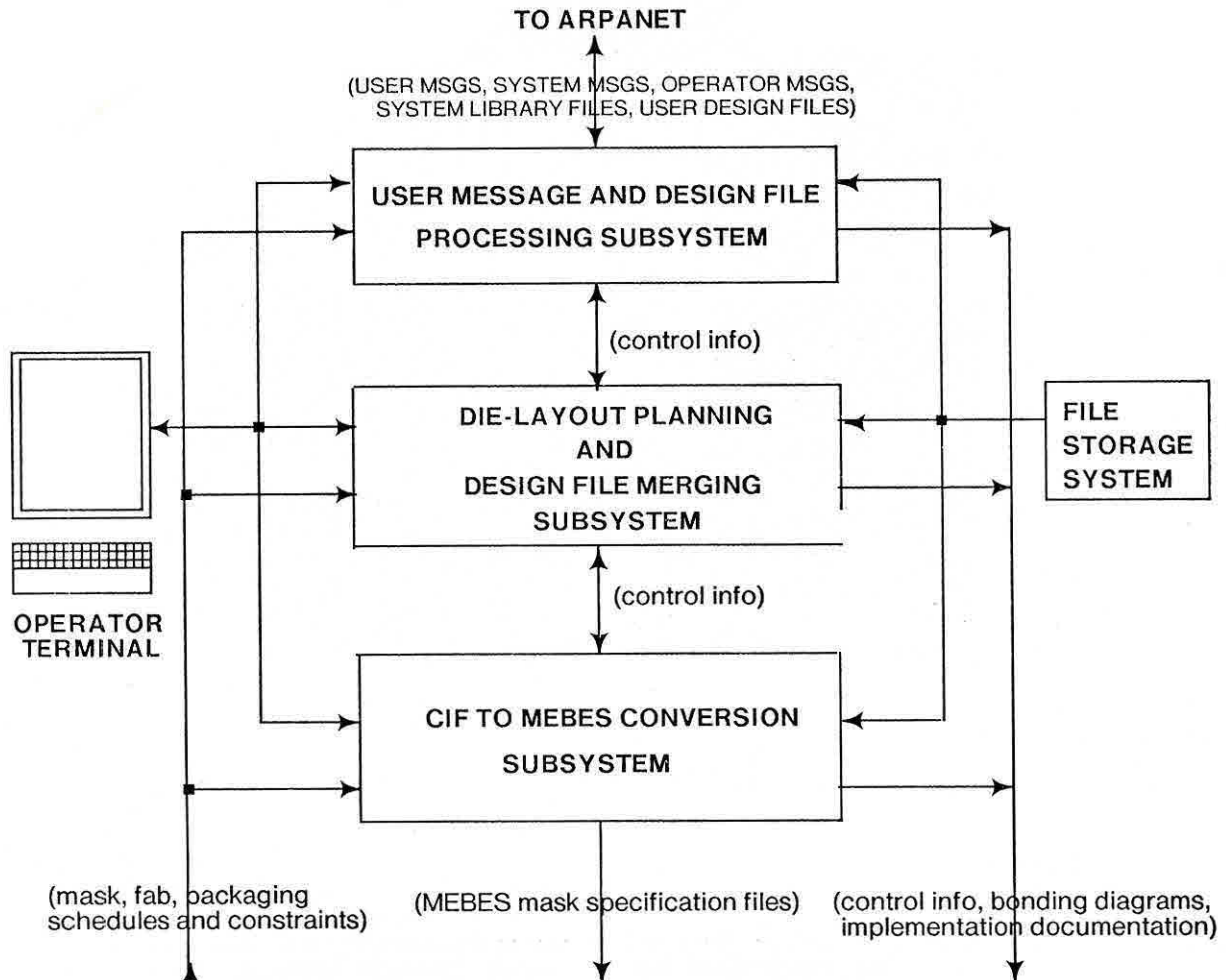


Figure 3. Block Diagram of the VLSI Implementation System

examining the design file. It ensures that it contains legal CIF code and determines the bounding box. The design file is stored for future use. The system replies to the user indicating that his request succeeded.

## 6. Information Processing

A fabrication run closing date is set and announced to the users. For MPC79, this date was December 5, 1979. On this date, the system examines the submittal queue. All designs in the queue are merged into the fabrication run. The system performs a set of planning, conversion, and information management actions.

The location of each project is planned. Each project must be placed on a die. Each die has to be located in several places on a wafer. These designs are placed in a standard "Starting Frame" which includes devices needed for fabrication such as scribe lines, critical dimensions, etc. This placement is aided by software which automatically selects the initial location of the projects. The operator then modifies this initial placement by graphically moving the boxes representing the projects. He may wish to optimize this placement because he wants to group certain design together, make designs more bondable, or better pack the designs. The system ensures that no projects overlap.

The foundry vendors need to be selected and the parameters specified. Various foundries have different requirements and capabilities. Design rules may be different; processes may vary. The Implementation System must specify parameters to accommodate these

differences. It can do that by specifying threshold parameters, and other process variables, and also by compensations. These compensations are small changes to the size of various layers so that the sizes that result will be what the designer specified.

After planning and parameter specification occur, conversion has to be performed. The CIF design files have to be converted to pattern generation format. This involves one or more machines computing. For MPC79, this took six hours on each of three powerful machines.

Vendor specific documentation is automatically created. A run letter is created for each foundry vendor. These letters include information such as the required parameters, locations of critical dimensions, etc. A bonding map is generated for each design.

## 7. Foundry Interaction

With all the data converted, commercial foundry services are accessed. First, the masks need to be made. This is done on electron-beam mask making equipment. E-Beam machines enable full wafer utilization. MEBES pattern generator format is universally accepted as input. The vendor writes the masks, then inspects, and corrects defects. Micromask made the first mask for MPC79 in one day. The delivery of the other masks was staggered to coincide with their need in the fabrication process.

The masks are taken to a fabrication foundry. They make the silicon wafers from these masks in a complicated procedure that takes from one week to six

## MPC79 Flowchart:

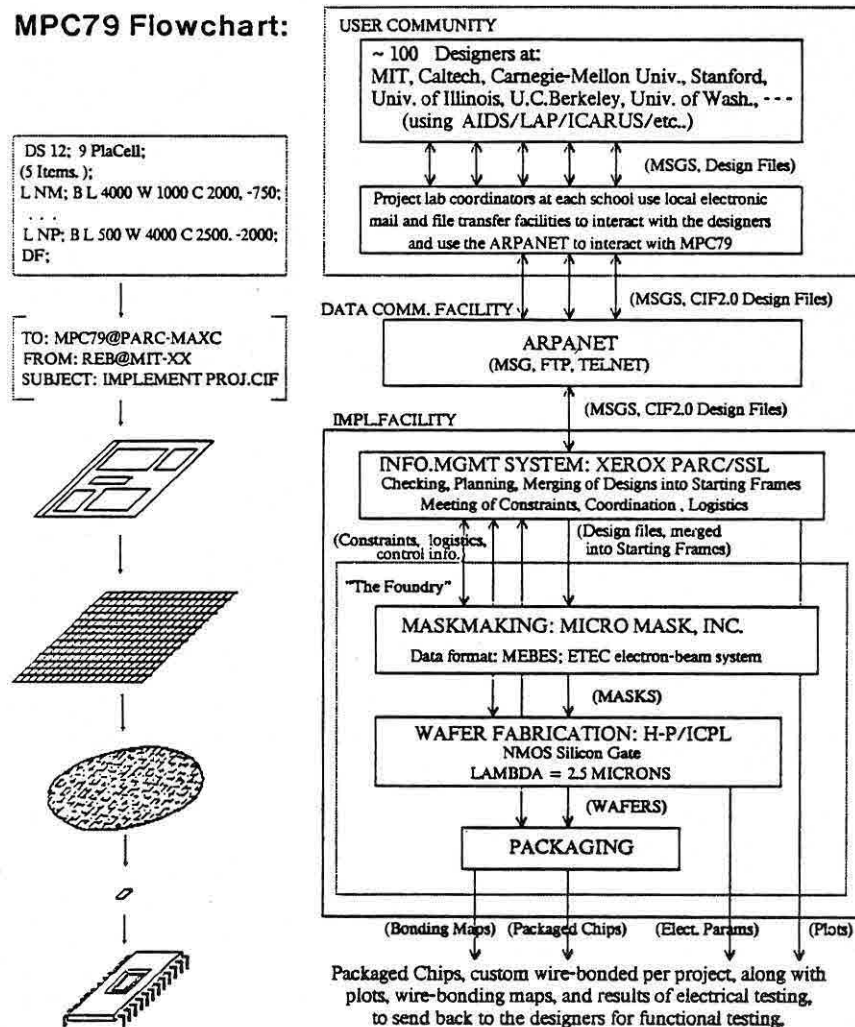


Figure 4. MPC79 Flowchart

weeks. Hewlett-Packard Integrated Circuit Laboratory, under the direction of Pat Castro, performed the wafer fabrication in about three weeks for MPC79.

When the wafers were finished, parametric tests are performed on the wafers to verify that the processing was done correctly. These tests check threshold voltages, resistances, etc.

The wafers, along with the bonding diagrams, are sent to the packaging vendor. The packaging vendor cuts up the wafers, mounts the dice, and then bonds them according to the supplied diagram.

The packaged chips are then returned to the designer. This whole process takes between two and eight weeks. MPC79 illustrated that fast-turnaround implementation was possible by returning design in less than one month from the cutoff date. In 1979, this was a surprising accomplishment, especially for such a large number of designers.

## 8. Future of Implementation Systems

This technology was transferred into ongoing operational use at USC Information Sciences Institute. They use their system, called MOSIS, to implement the designs from DOD ARPA supported universities on a monthly basis.

SynMos is offering implementation services commercially. They are doing runs monthly. Many companies, including Xerox, Hewlett-Packard, Digital Equipment Corp., have internal implementation systems and services.

MPC79 demonstrated that this methodology works. It is gaining widespread acceptance. The Implementation System concept allows fabrication foundries to handle the large number of designs resulting from the Mead-Conway design methodology.

## 9. References

Clark, J. H. (1980). "A VLSI Geometry Processor for Graphics", *Computer*, Vol. 13, No. 7.

Conway, L. A., Bell, A. G., Newell, M. E., Lyon, R. F., and Pasco, R. C. (1980). "Implementation Documentation for the MPC79 Multi-University MultiProject Chip-Set." *Xerox PARC/SSL Technical Memorandum*.

Conway, L. A., Bell, A. G., and Newell, M. E. (1980). "MPC79: The Large-Scale Demonstration of a New Way to Create Systems in Silicon." *Lambda*, Vol. 1, No. 2.

Conway, L. A. (1979). *The MIT '78 VLSI System Design*



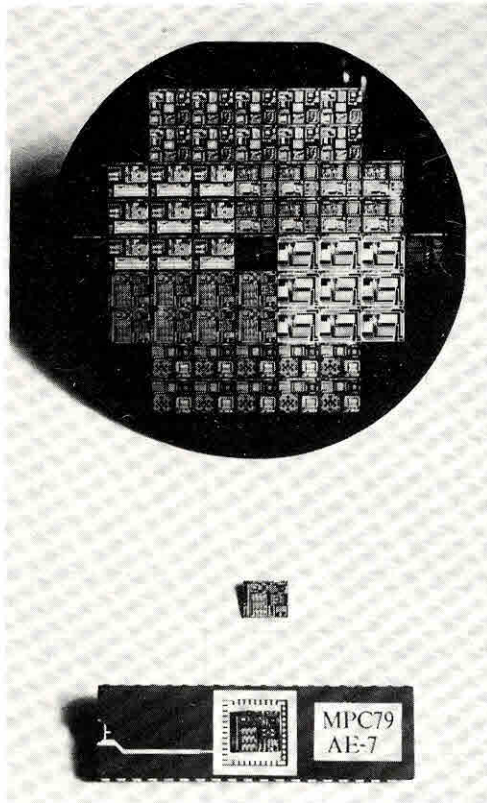


Figure 5. Photo of MPC79 type-A wafer, type-AE die, type-AE-7 packaged chip

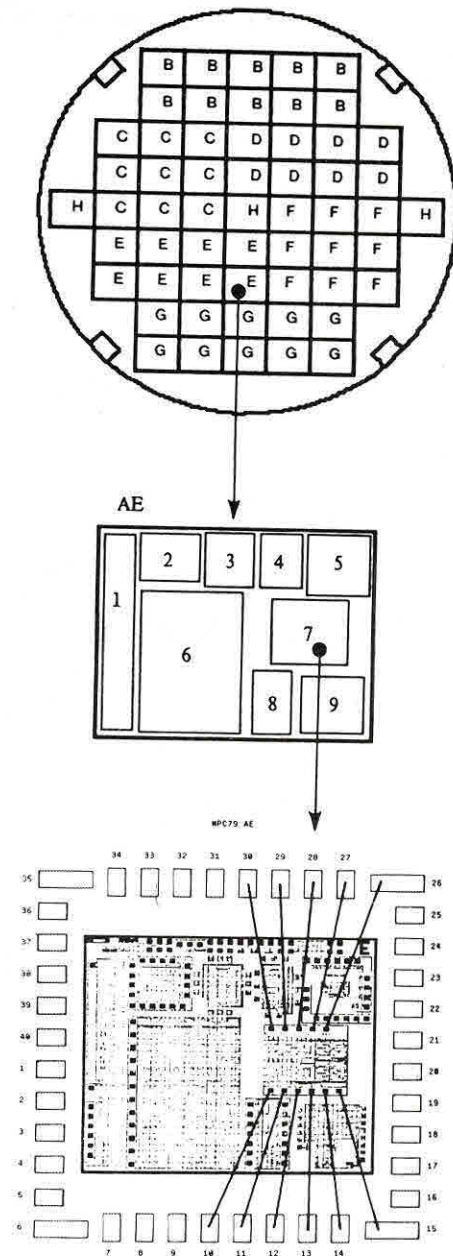


Figure 6. Corresponding Hierarchy of Informational Material

Course: A Guidebook for the Instructor of VLSI System Design. Xerox PARC.

Conway, L. A. (1981). The MPC Adventures: Experiences with the Generation of VLSI Design and Implementation Methodologies. Xerox PARC Technical Report VLSI-81-2, Palo Alto, California.

Hon, R. W., and Sequin, C. H. (1980). A Guide to LSI Implementation, 2nd Edition. Xerox PARC/SSL Technical Report SSL-79-7, Palo Alto, California..

Lyon, R. F. (1981). The Optical Mouse, and an Architectural Methodology for Smart Digital Sensors. Xerox PARC Technical Report VLSI-81-1, Palo Alto, California.

Mead, C. A. and Conway, L. A. (1980). Introduction to VLSI Systems. Reading Massachusetts, Addison-Wesley.

Patterson, D. A. and Sequin, C. H. (1980). "RISC I: A Reduced Instruction Set VLSI Computer," Proc. Eighth International Symposium on Computer Architecture, pp. 443-457.