UNIVERSITY OF SOUTHAMPTON

# Interconnection Schemes for Wafer Scale Reconfigurable Orthogonal Cellular Array

by

Rosli Bin Haji Mahat

1986

UNIVERSITY OF SOUTHAMPTON

## ABSTRACT

FACULTY OF ENGINEERING AND APPLIED SCIENCE

DEPARTMENT OF ELECTRONICS AND INFORMATION ENGINEERING

Master of Philosophy

# INTERCONNECTION SCHEMES FOR WAFER SCALE RECONFIGURABLE ORTHOGONAL CELLULAR ARRAY

by Rosli Bin Haji Mahat

Three schemes for configuring a functional, wafer-scale two-dimensional orthogonal cellular array are presented. The functional array is to be configured on a physical array of good and bad cells by using programmable gates on the intercellular connection lines. The yields for the three schemes were obtained from computer simulations of the configuration of the functional array by the three schemes. The aim is not just for high array yield but also for high cell utilisation. The yields obtained were found to be better than the yields obtained from other available interconnection schemes. With 80 % cell yield, an array yield of 96 % was obtained with 75 % cell redundancy for one of the schemes. Its cell utilisation was 55 %. For one of the schemes, the design of the control element for the configuration of the interconnections is also presented. Several modifications to the interconnection schemes were also tested. The extra row modification was found to improve the yields of the three schemes. The extra bypass line modification was found to reduce the drop in yields when the length of the interconnection lines was restricted.

# ACKNOWLEDGEMENT

This writer wishes to acknowledge the supports, assistances and encouragements of the following persons and institutions during the making of this thesis:

Dr. William Moore, for his guidance, suggestions and support during the supervision for most part of this thesis.

Dr. Chris Jesshope, for his willingness to takeover the supervision for the latter part of this thesis.

The Department of Electronics and Information Engineering, and the Department of Computing Service of the University of Southampton, for the use of their facilities during the research for this thesis.

The Malaysian Public Service Department and the University of Malaya, for their financial support during the stay at Southampton.

All staff and students at the Department of Electronics and Information Engineering, especially colleagues in Room 106, Zepler Building, for their suggestions and assistances during the research for this thesis.

Ahmad Afif Rosli and his mother, Noriyah Shaari, for their encouragement, patience and support during the last two years which make it all worthwhile.

## TABLE OF CONTENTS

## LIST OF FIGURES

# LIST OF TABLES

Chapter 1

# INTRODUCTION

One area of electronics that is being actively studied is Wafer Scale Integration (WSI). The development of WSI from Very Large Scale Integration (VLSI) is discussed in the first section. A design suitable for WSI is cellular array circuits. Of interest in this thesis is the interconnection among the cells in the cellular array. The various methods of forming the interconnection are discussed in sections 1.2 and 1.3. The objectives of this thesis are discussed in section 1.4.

## 1.1 DEVELOPMENT OF WAFER SCALE INTEGRATION

A trend in the development of Very Large Scale Integration (VLSI) is to increase the size and the density of a chip by putting more circuitry onto the chip. There is not just the advantage of a reduced packaging and assembling cost, but also of having higher reliability and better performance.

With large chips, the amount of interchip connection can be reduced. Some of the off-chip connections for an assembly of smaller chips are replaced by on-chip connections in the large chip. On-chip connection is more efficient and more reliable than off-chip connection. There is less delay, less noise and less power loss in the on-chip connection than in the off-chip connection. Futhermore, the space occupied by one big chip is smaller than that occupied by the assembly of the smaller chips.

As the chip is made bigger, the limit to its size would be the size of the wafer slice on which the chip is fabricated. A new area of study called Wafer Scale Integration (WSI) is opened for 'chips' of about the size of the wafer. Many theoretical studies and prototypes have been done for WSI since the mid-1960's [Mang84a,Mang84b,McDo84,Moor84, Moor85b].

Problems that are already difficult at the VLSI level, become more complicated at the WSI level. The testing and packaging of the WSI chip is more difficult and more expensive than for the VLSI chip. Because of its larger area, defects are more likely to be formed on the WSI chip than on the VLSI chip. Therefore, a WSI chip is more likely to be defective than a VLSI chip Even to design and to produce a prototype of a WSI chip is more tedious more expensive and takes a longer time. However, the advantage of having a fast, highly reliable device could overcome these disadvantages and makes WSI desirable.

The packaging for a WSI chip has to satisfy three main requirements. Firstly, it must be able to accommodate the large WSI chip. The size of a WSI chip could be up to more than 50 cm2. Secondly, the packaging must be able to accommodate the high number of I/O leads. The common dual-in-line package (DIP) or chip carrier are inadequate. New packagings such as the pin-grid array need to be developed so as to be able to handle the high number of leads for WSI [Bowl85,Neug84]. Thirdly, the packaging must also be able to dissipate the heat generated by the WSI chip. Air cooling may no longer be adequate for the WSI chip. Cooling using water or even a refrigerant needs to be considered [Blod83, John84, Pelt83].

As the area of a chip increases, defects are more likely to be formed on the chip. Therefore, the probability of the chip being defective also increases [Bert83, Stap83]. This loss of chips could be reduced by designing the chip in blocks or cells. Redundant cells are added to the design. Any cell containing defects would be disconnected from the other cells and replaced by one of the redundant cells.

Another problem that arises with increased density and increased size of the chip is the increased difficulty in testing the chip. This is because of the difficulty in accessing and testing the various parts of the chip. This problem could be simplified if the chip has been design into cells. It is easier to test each cell one at a time than to test the whole chip at once.

From the above two reasons, it can be seen that WSI is more suitable for circuits with repetitive cellular arrays such as memory and processor arrays than for random logic circuit. The testing and repair of defects are easier for the cellular arrays. Redundant cells are added to the cellular array so that any defective part of the chip can be isolated by disconnecting the cell with the defect from the array and replacing the cell with one of the redundant cells. A cellular array would also help to simplify the designing and the laying-out of the chip.

So far in this section, it has been shown that there are various difficulties in implementing WSI. Some of these difficulties can be reduced by designing the WSI circuit in a cellular array but the cost of processing, testing and packaging a WSI chip is still high as compared to the cost for a single VLSI chip. However, several VLSI chips are required in order to have the same computational capability as a WSI chip. Therefore, a major part of the cost for the WSI chip could be offset. Combining this with the fact that a WSI chip is more reliable and can operate at a faster speed than an assembly of VLSI chips, WSI is highly suitable for fast, high-volume data processing such as high density memory, systolic computation and image processing.

Many of the work in WSI has been in memory design [Bars77,Egaw80, Elme77,Hunt76,Hsia79,Kita80]. This is because of the relatively simple design and simple intercellular connection for memory.

Beside memory, WSI is also suitable for array or parallel processing where several identical processors are utilised at one time. The processors may be connected in a linear or a two-dimensional array. Many schemes has been tried for the linear processor array [Aubu79, Finn77,Fuss82,Mann77,Varm83]. In the linear array, each processor is connected to two other processors. A processor would processes data it received from one processor and passes the result to the other processor.

In a two-dimensional array, the processors are required to be connected to three or more other processors. It is more difficult to implement because of the increased difficulty in designing and configuring the interconnections among the cells.

In this thesis, the interconnection schemes for two-dimensional orthogonal cellular array are studied. An example of this type of array is shown in figure 1.1. In this array, each cell has four interconnection lines to its four nearest neighbours. An example of the configuration of a functional orthogonal array is shown in figure 1.2. The functional array of good cells is configured on a physical array of good and bad cells. The configured array is the array obtained after the configuration has been done.

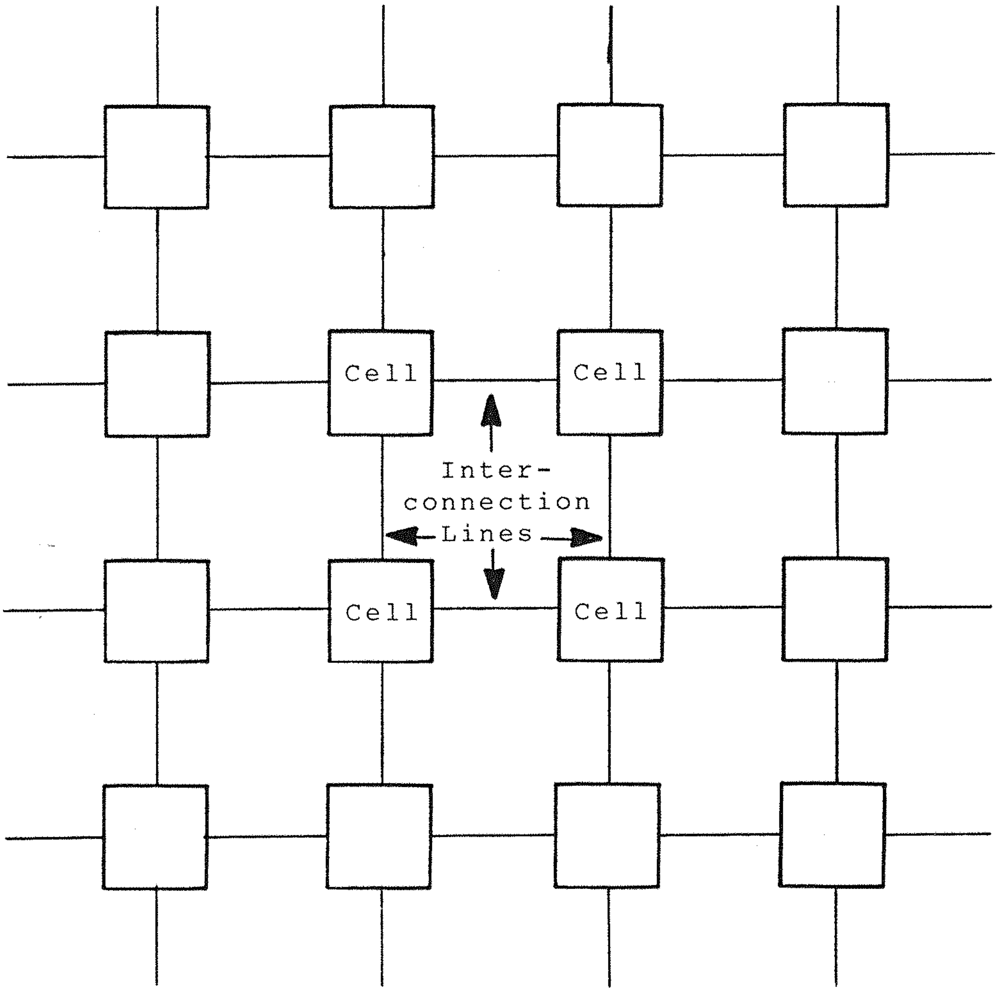
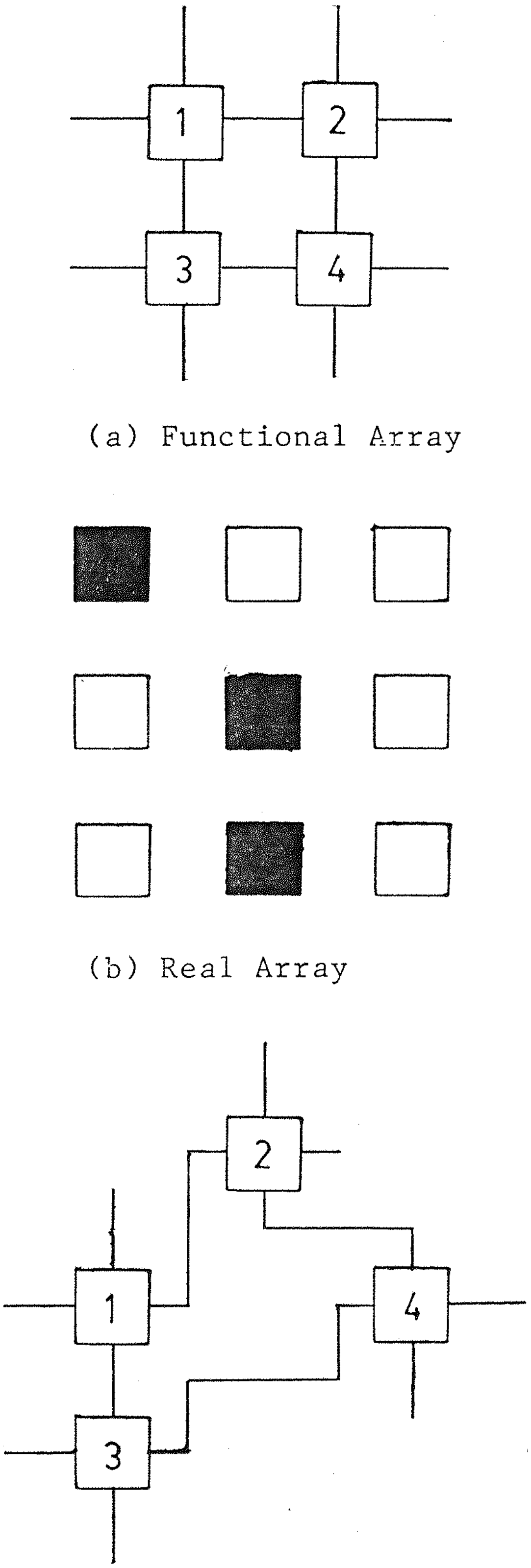


Figure 1.1. A 4X4 two-dimemsional orthogonal cellular array.



(a) Functional Array

(b) Real Array

(c) Configured Array

Figure 1.2. Configuration of a 2X2 functional array on a 3X3 physical array of good ( □ ) and bad ( ■ ) cells.

## 1.2 NON-VOLATILE INTERCONNECTION REPAIR TECHNIQUES

An important aspect in designing a configurable cellular array is the designing of the interconnections among the cells. An interconnection scheme must be able to form the connection between two good cells and also to totally isolate any bad or unused cells. Various methods have been developed to form the interconnections among the cells [Aubu78, Mang84b,Mang84c,Moor84,Siew82]. These various methods are discussed in this and in the next sections.

Early studies of intercellular connections have used discretionary wiring to form the interconnection [Bars77,Calh72,Lath67,Petr67]. In this method, the required interconnections among the cells are made by using additional levels of customised metalisation after the cells have been probe-tested.

There are two main drawbacks to discretionary wiring. Firstly, the metalisation and other processing steps after probe-testing must be fault-free and must not introduce new defects on the wafer. Secondly, each wafer needs a tailor-made mask for metalisation. This is very expensive. Recent development in electron beam lithography has reduced this problem by providing a direct write capability on the wafer but the repair job is still tedious [Berg85,Donl85,Frie84].

In other repair techniques, near-complete intercellular connections are laid out on the wafer during the processing. The required array is then formed by connecting and/or breaking the interconnections after the cells has been tested. These can be done by using laser beam, electron beam, high current or reprogrammable gates. Among these methods, only the reprogrammable gates is volatile. Repairs using reprogrammable gates are discussed in the next section.

In the laser technique, laser beam is used to make or to cut the interconnections [Chap85,Gave83,John84,Logu81,Posa81,Raff83]. The laser beam can make links by heating a high resistance polysilicon into a low resistance connections [Mina82], or by welding the metal connections [Chap85,Schu76,Wu82]. To break the links, the laser beam is used to cut across the interconnection lines. Chips using this technique are probe-tested and repaired at the finished but still unpackaged stage.

Another method to make or break the connections is to use electron beam [Mang84b,Mang84c,Shav83]. Unlike the laser beam, the electron beam has an inner body heating capability. Hence it can be used to heat points within the wafer that are away from the surface [Mang84b]. The laser beam can only be used for region that is within about 1 $\mu$m of the surface. Therefore, the advantage of using the electron beam is that the repair sites can be anywhere within the wafer whereas the repair sites for the laser beam must be near the surface. A big disadvantage of using the electron beam is that the repair operation must be done in a vacuum. This would increased the repair time and cost of the wafer.

Beside the disadvantage of requiring a beam generating system, the laser and the electron beam techniques require additional hardware for precision beam positioning. A method that does not require as much test equipments as both these techniques is the high current technique. The high current can be used to blow up fusible links

[Lanc83,Spaw82,Stop85]. It can also be used to change the physical characteristic of polysilicon to make links [Line82,Mano82]. The voltage used for the repair is about 11-20 volts. An advantage of this method is that the testing and programming can be done after the WSI chip has been packaged when new defects are unlikely to be introduced. However, additional space on the wafer is required for the high current control transitors and supply lines.

## 1.3 RECONFIGURABLE INTERCONNECTION REPAIR TECHNIQUE

All the repair techniques in the previous section are non-volatile. The interconnections cannot be changed after the configuration has been done. A reconfigurable repair technique is to use reprogrammable gates [Aubu79, Catt81, Finn77, Fuss82, Hedl82, Hsia79]. On-chip control elements are used to set the gates on the interconnection lines. The interconnection may later be reconfigured to overcome defects that may occur during the lifetime of the wafer or to change to a different cellular array.

In this technique, the control elements must be fault-tolerant in order for the right configuration to be done. Thus, considerable area of the wafer is taken up by the control elements. Compared with the other repair techniques, this method requires the most area overhead for the control logic. On the other hand, this method does not require any expensive or bulky test and repair equipments as in the other repair methods.

One of the simplest reconfigurable interconnection scheme for two-dimensional orthogonal array was tried by Manning [Mann75, Mann77]. A configuration using this scheme is shown in figure 1.3. A switching element and a processing element is associated with each cell. Each cell can be connected directly to its four nearest neighbours. The cell can also be used as a bridge to connect any two of its four nearest neighbours but with the loss of the processing capability of the cell. This scheme requires a high cell yield and a high amount of redundancy. To form a 16X16 array of good cells, an array with a high cell yield of 97.5 % would still need a redundancy of 144 %.

Hedlund proposed several schemes for the Blue CHiP (Configurable Highly Parallel) computer [Hedl82,Hedl84]. One of the schemes is shown in figure 1.4. With numerous switches and connection lines, several interconnecting paths can be formed between any two cells. This allows for not just the orthogonal array but also for other types of arrays to be formed. A problem with this design is the difficulty in configuring the interconnection because of the high number of choice for the interconnection between any two cells. There is also the long delay on the interconnection lines because of the numerous switches involved.

To limit the length of the interconnection lines and also to simplify the programming, Hedlund used a two-level hierarchy for the orthogonal array. The array is divided into blocks of cells. After configuration has been done within each block, the blocks are then connected together to form the required array. To produce a 16X16 functional array with 65 % cell yield, the best array yield is obtained by having a 9X9 array of blocks with 3X4 cells each. A 2X2 array of good cells is configured in each blocks. Among the blocks that are able to produce the required array, a 8X8 block array is then formed.

Manning and Hedlund used redundant rows and redundant columns for their schemes. In this study, this type of redundancy is called the two-dimensional redundancy. Generally, configuration of the functional array on array with two-dimensional redundancy is more difficult than on array with one-dimensional redundancy. With one-dimensional redundancy, an array can only has either redundant rows or redundant columns of cells. The initial strategy for the interconnection schemes of this thesis is to use the one-dimensional redundancy. It may not have a high level of connectability among the cells but the configuration of the functional array would be relatively simple.

The one-dimensional redundancy was tried by a group at the Milan Polytechnic, Italy [Sami83]. In their schemes, redundant columns of cells are used. One of their schemes is shown in figure 1.5. For the column connection, each cell can be connected to one of three cells in the row just above it and to another one of three cells in the row just below it.

The programming of the interconnection is done by the two multiplexers and two link controls for each cell. The programming allows for only one bad cell per row. Therefore, only one redundant column is necessary. Another scheme has two redundant columns which allow for two bad cells per row, but it requires much more interconnection and control lines among the cells. Instead of three choices, each cell can be connected to any one of five cells in the rows above or below it. Because of the limited allowances for defective cells per row, these two schemes are suitable only for array with very high cell yield. Even with cell yield of 95 %, the array yield of 15X15 array is only about 50 % with the two-redundant-column scheme.

Another scheme for one-dimensional redundancy is tried by Evans, McCanny and Wood [Evan85]. In their scheme, redundant rows of cells are used. For the row connection, each cell is to be connected to one of three cells in each of the two columns that is just to the right and just to the left of the cell.

The control elements of each cell uses 12 control lines to communicate with its six nearest neighbours as shown in figure 1.6. The programming of the interconnection is based on the 'request' (REQ) and 'available' (AVAIL) signals from the neighbouring cells and on the result of the self-testing of the cell. Because there is no limit on the number of defective cells in the array, this scheme is able to give a higher array yield than the schemes of Milan Polytechnic and can also be used for array with lower cell yield. It can form a 10X10 array with 65 % cell yield by using 22 redundant rows.

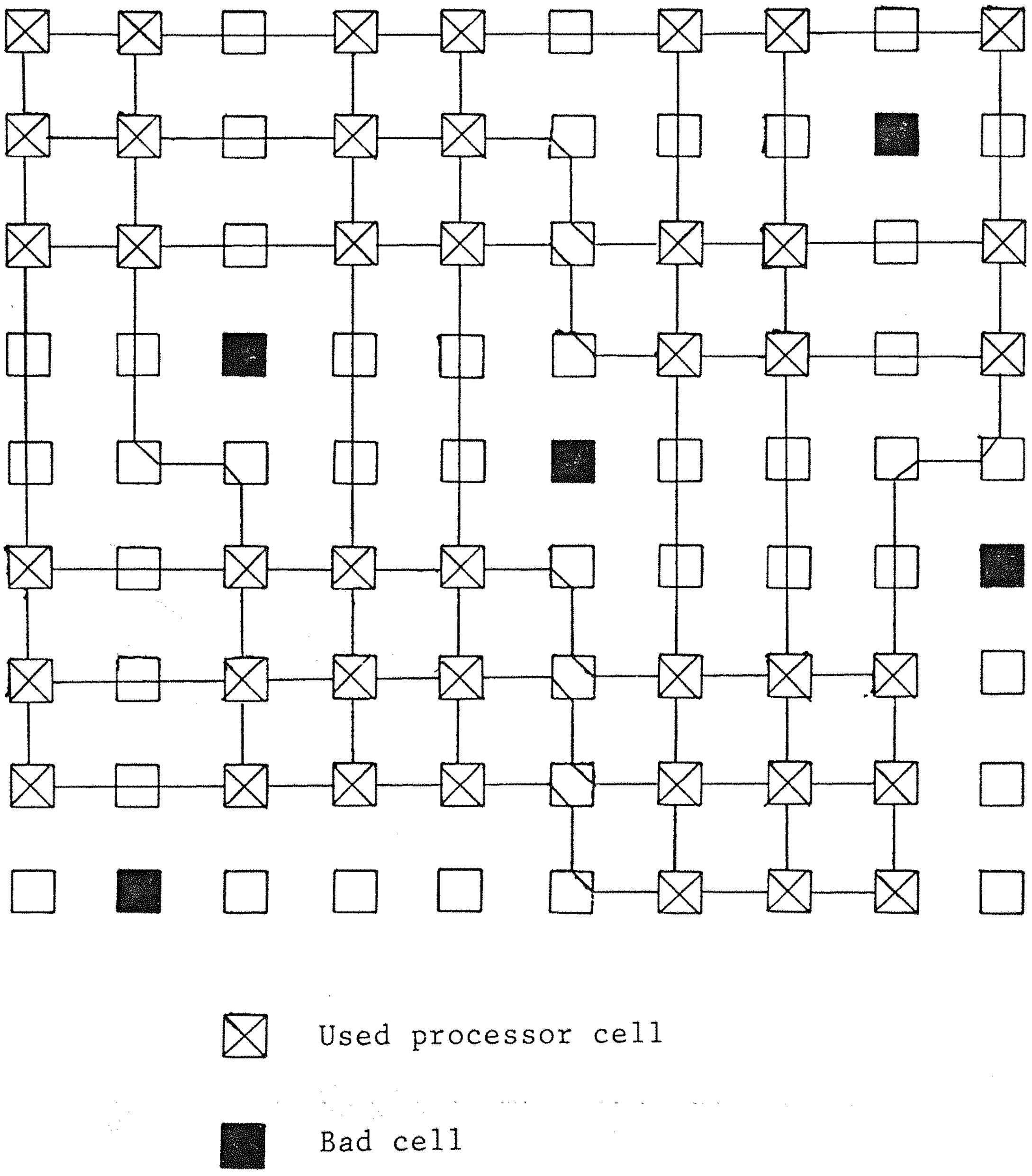


Figure 1.3 Manning's interconnection scheme configuring a 6X7 functional array on a 9X10 physical array.

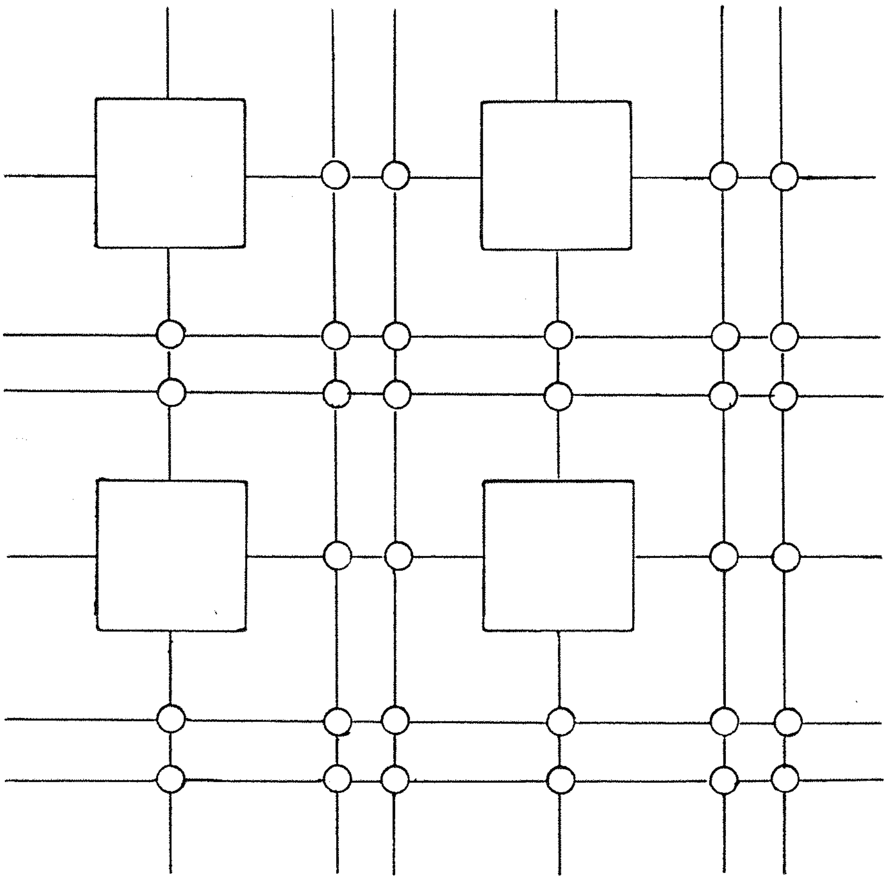


Figure 1.4  One of Blue CHiP interconnection schemes.

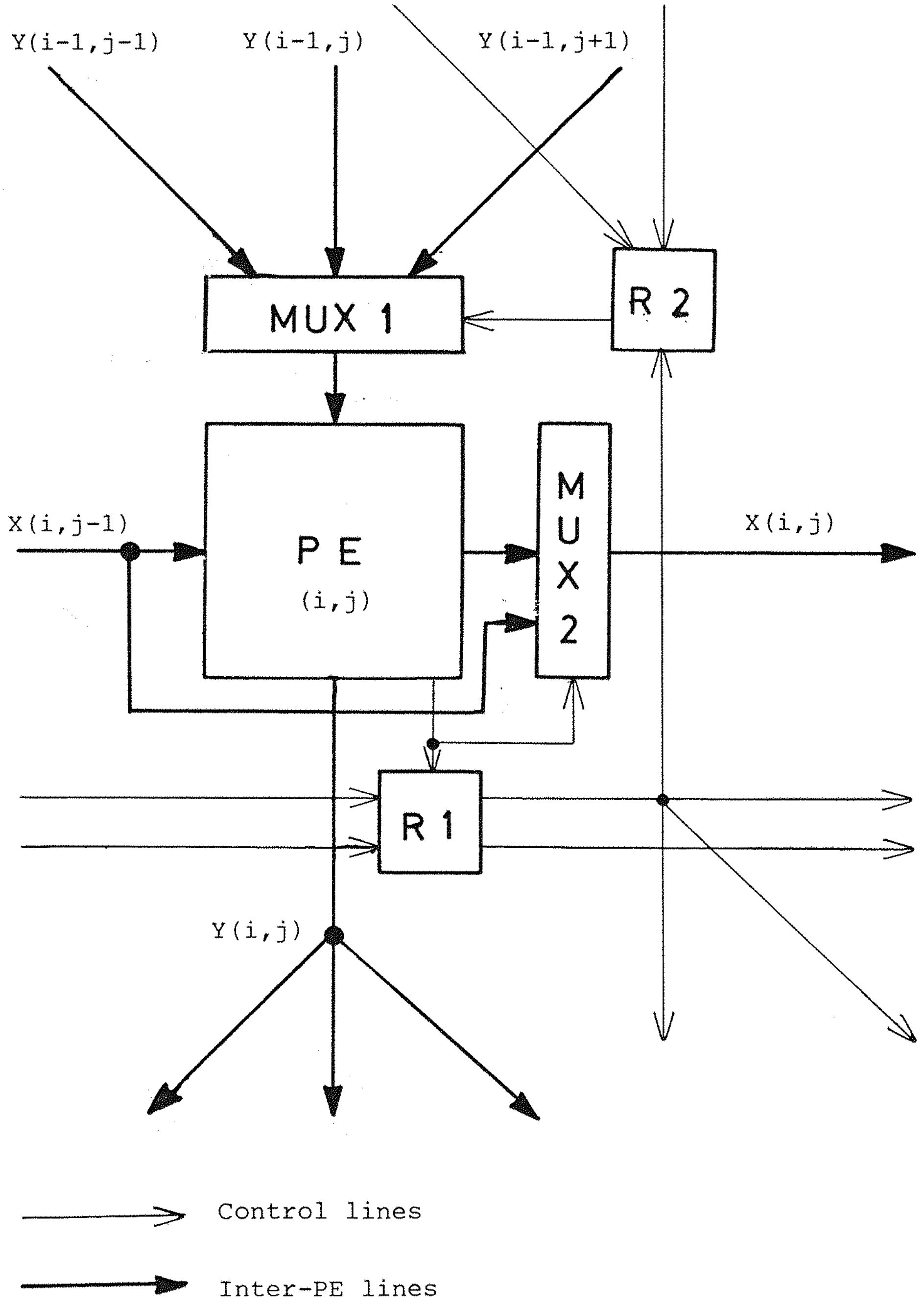


Figure 1.5 A cell of Sami's interconnection scheme for a physical array with one redundant column.

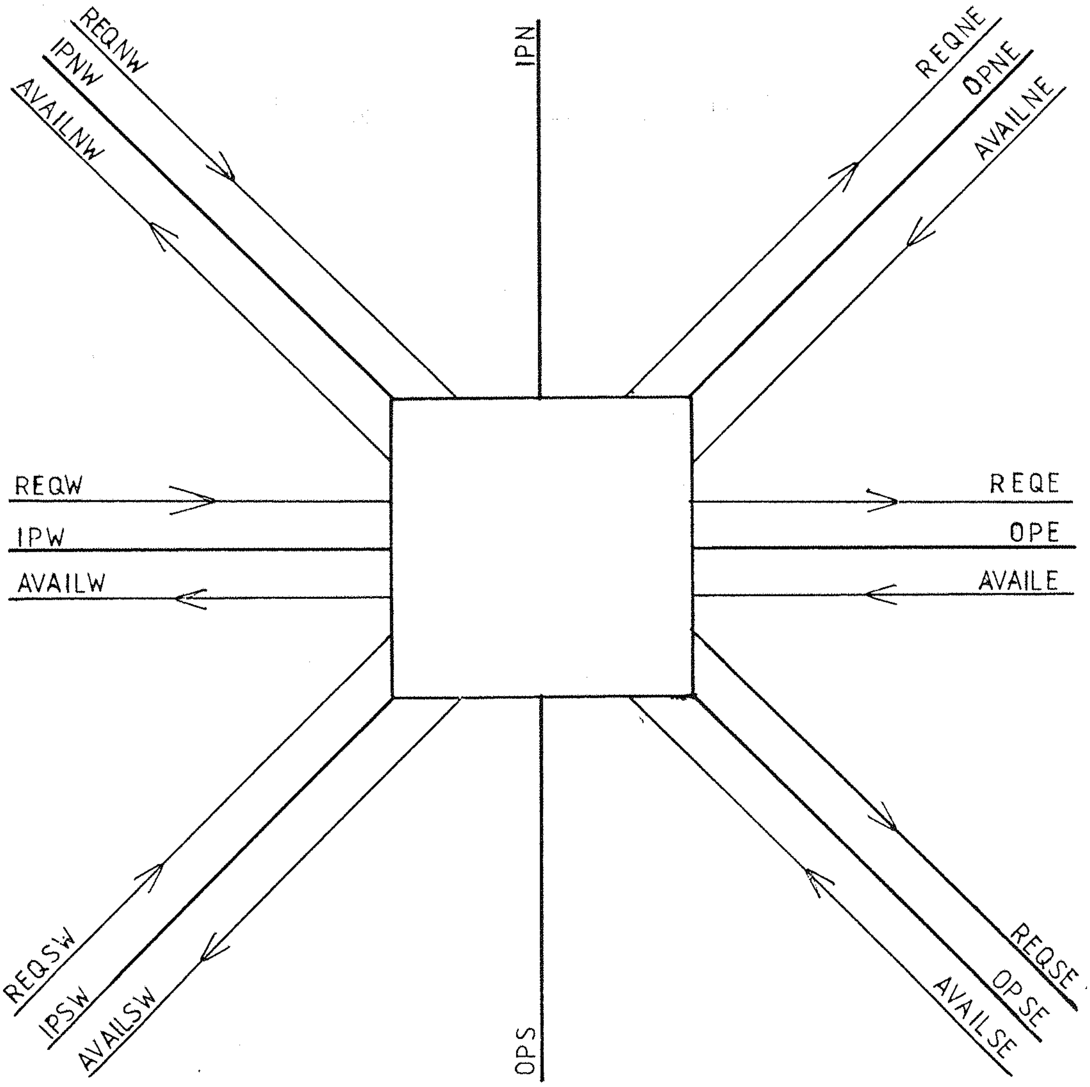


Figure 1.6 A cell of the interconnection scheme of Evans showing the direction of flow of control signals for the interconnection.

## 1.4 OBJECTIVES OF THESIS

From the first section, it can be seen that WSI could be superior over VLSI for fast, reliable, high volume processing. The main reasons that have prevented wide-spread use of WSI have been low yield and high production cost [Moor85b].

In this thesis, ways of improving the yield by using interconnection schemes with high connectability and easy programming are studied. There are various types of array configuration that could be studied. However, this thesis will be restricted to the two-dimensional orthogonal array like the one shown in figure 1.1. There have already been numerous studies into the linear one-dimensional array.

The objective of this thesis is to develop a suitable interconnection scheme for the orthorgonal cellular array. This is done by using computer simulation to test three interconnection schemes and also to test some modified versions of the three schemes. The aim is not just for high array yield but also for high cell utilisation.

The interconnection schemes to be studied in this thesis are shown in the next chapter. The yield simulations of the schemes are done in chapter 3. A summary of the results and the area for future research are discussed in chapter 4. A more detailed comparision of the expected yield of the various schemes described in section 1.3 and of the various schemes used in this study is also shown in chapter 4.

Chapter 2

# INTERCCNNECTION SCHEMES FCR CRTHOGONAL ARRAY

In this chapter, three interconnection schemes for two-dimensional orthogonal cellular array are presented. The physical description of the interconnection schemes are shown in section 2.1. The methods of testing the cells are discussed in section 2.2. The various steps in the configuration of the functional array are described in section 2.3. Section 2.4 describes in detail the working of the control element for one of the schemes.

## 2.1 INTERCCNNECTICN SCHEMES

In the interconnection schemes to be investigated, redundant columns of cells are used. Each interconnection scheme consists of two parts, one for column configuration and the other for row configuration. A row configuration scheme is used to connect together the required number of good cells in each row of the physical array. The column configuration scheme is used to connect one good cell from each row in order to form the column of good cells for the functional array. In this study, three column configuration schemes with the same row configuration scheme were tested. Initial studies of these schemes have shown promising results [Burg82,Kent83,Moor85a].

A simple row configuration scheme is shown as scheme 1 in figure 2.1(a). Each cell has three gates. A good cell is connected by opening the two gates on its sides. To bypass the cell, these two gates are closed and the third gate on the bypass line is opened.

Scheme 1 is vulnerable to single point defect on the interconnection lines. This fault could render the whole row unusable. An alternative scheme is shown as scheme 2 in figure 2.1(b). This scheme is functionally identical to scheme 1 but has an increased fault tolerance. Two input are required for the control of each gate.

Even though it has more gates per cell than scheme 1, scheme 2 uses one less gate on the connection between any two cells. For a connection with N bypassed cells, scheme 1 requires N+2 gates whereas scheme 2 requires only N+1 gates. In this study, the row configuration of scheme 2 is used.

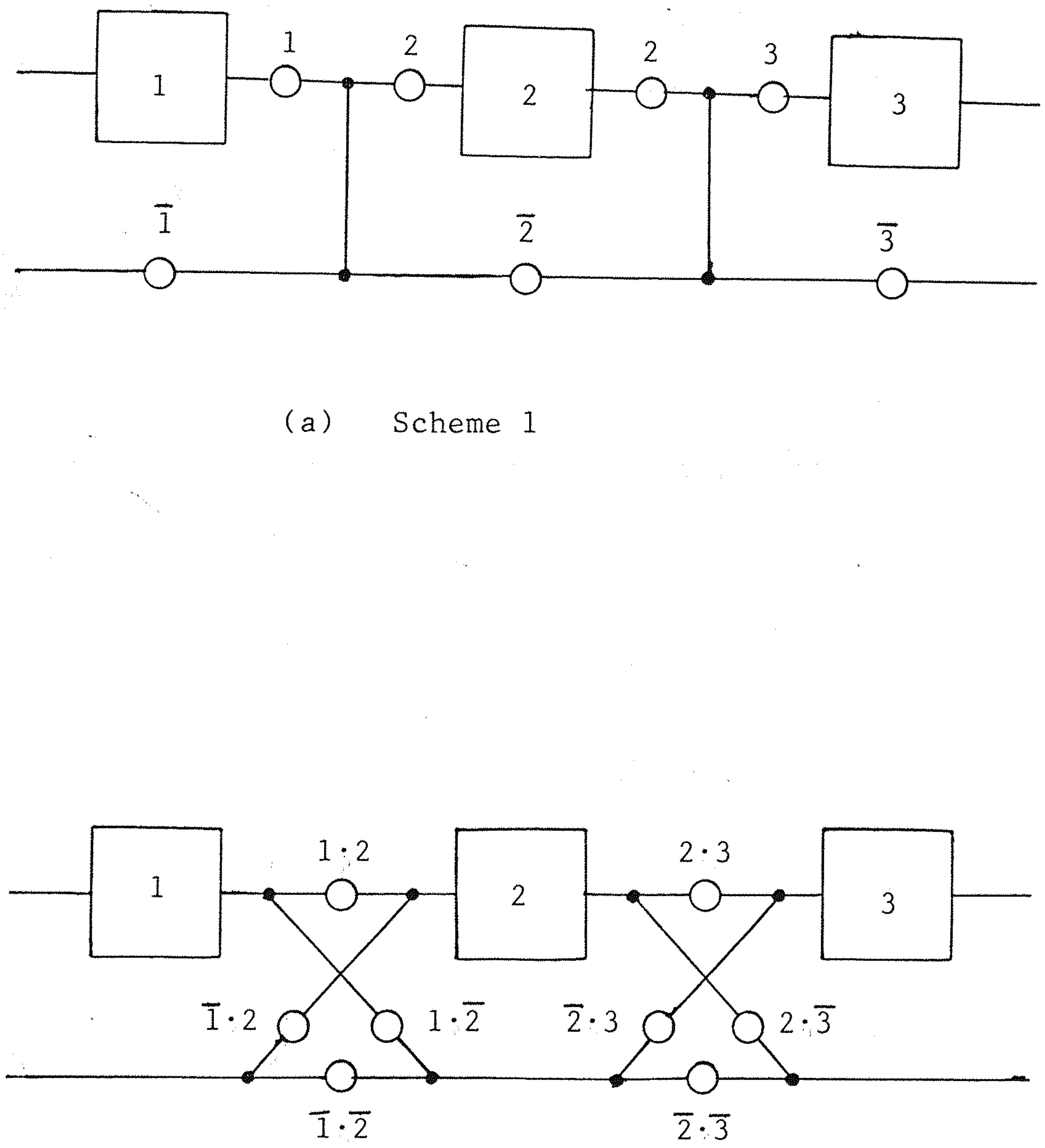
There are three designs used for the column configuration schemes. The simplest design is shown as scheme A in figure 2.2. Two cells in the same column are connected by opening the two gates between them and closing the two adjacent gates on the column shifting line. The column can be shifted left or right by opening the corresponding gate on the column shifting line, and closing the gates to the cells to be bypassed.

An improved design is shown as scheme B in figure 2.3. Unlike scheme A, it is possible to connect a good cell to the functional array in scheme B when the cell above or below it is connected to another functional column. There are two routes for connecting two cells in the same physical column. This can be done by opening the two gates on either the left or the right side of the diamond-shaped network. To shift physical column, one of these four gates is opened depending on the direction of shift and the cell to be connected. To bypass a physical column, the centre gate is opened and all the other four gates are closed.

Further improvement to scheme B can be made by having two column bypass lines as shown by scheme C in figure 2.4. It works in the same way as scheme B but with an increased connectability among the cells from the double bypass lines. The interconnection line can be shifted from one bypass line to the other when required.

Figure 2.5 shows the functional arrays configured on physical arrays with the same defective cells by the three schemes. In term of cell utilisation for the functional array, scheme B is better than scheme A with scheme C being the best since it can configure the most number of columns.

Beside the three interconnection schemes already presented, several modified versions of the schemes were also tested. These modified schemes are described in section 3.3.



(a) Scheme 1

(b) Scheme 2

Figure 2.1 Two schemes for row configuration.

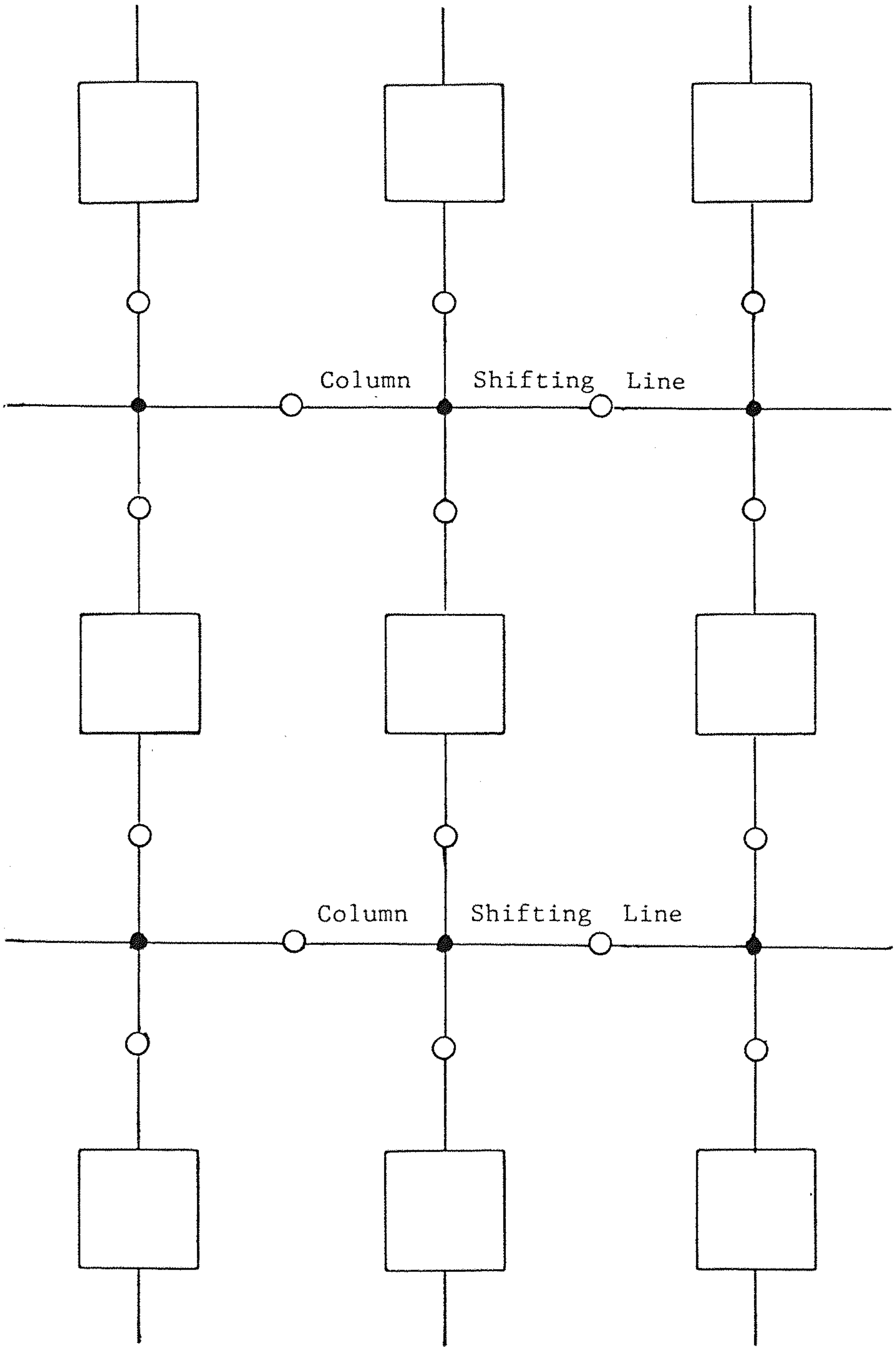


Figure 2.2 Column configuration scheme A.

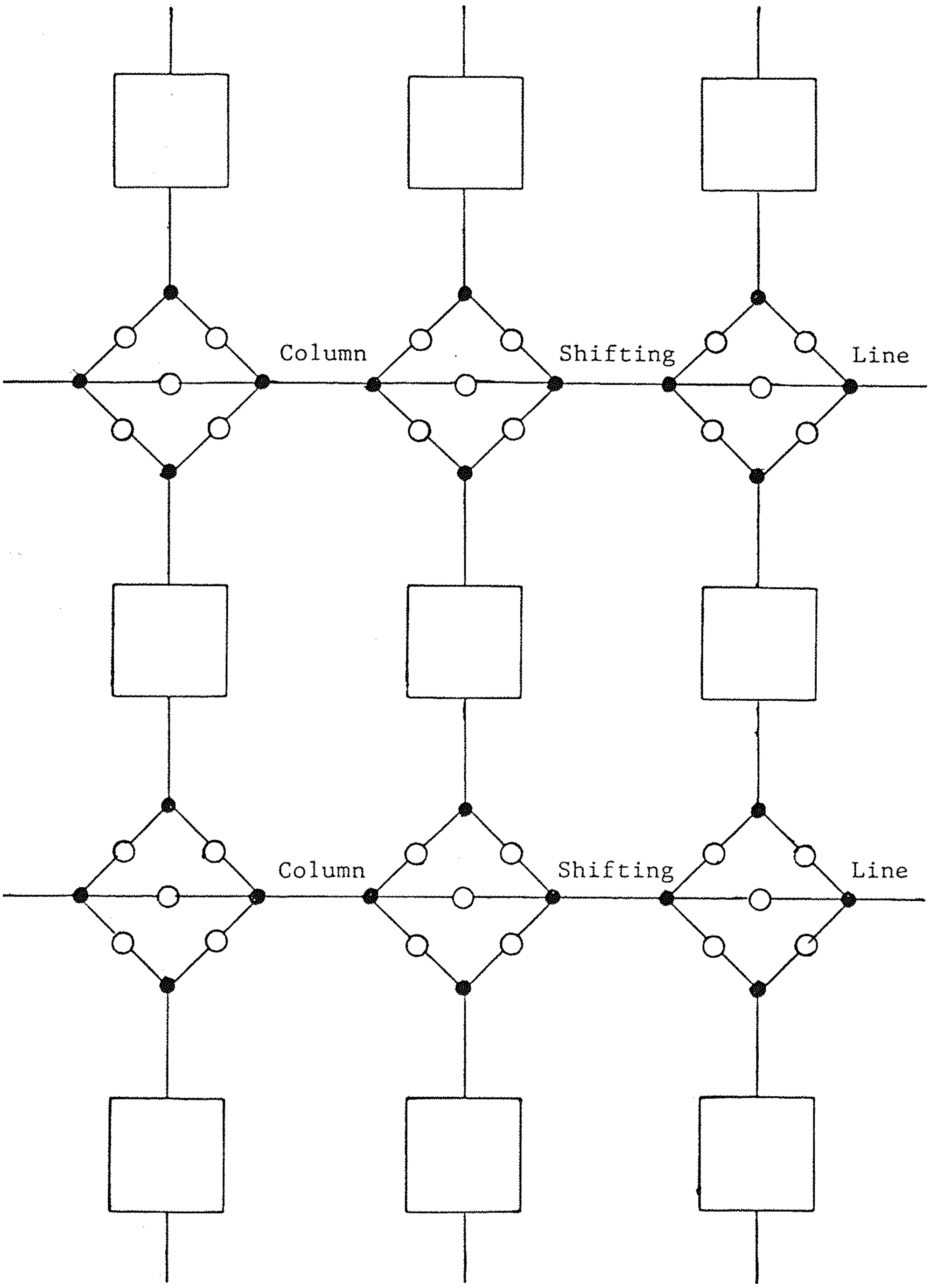


Figure 2.3 Column configuration scheme B.

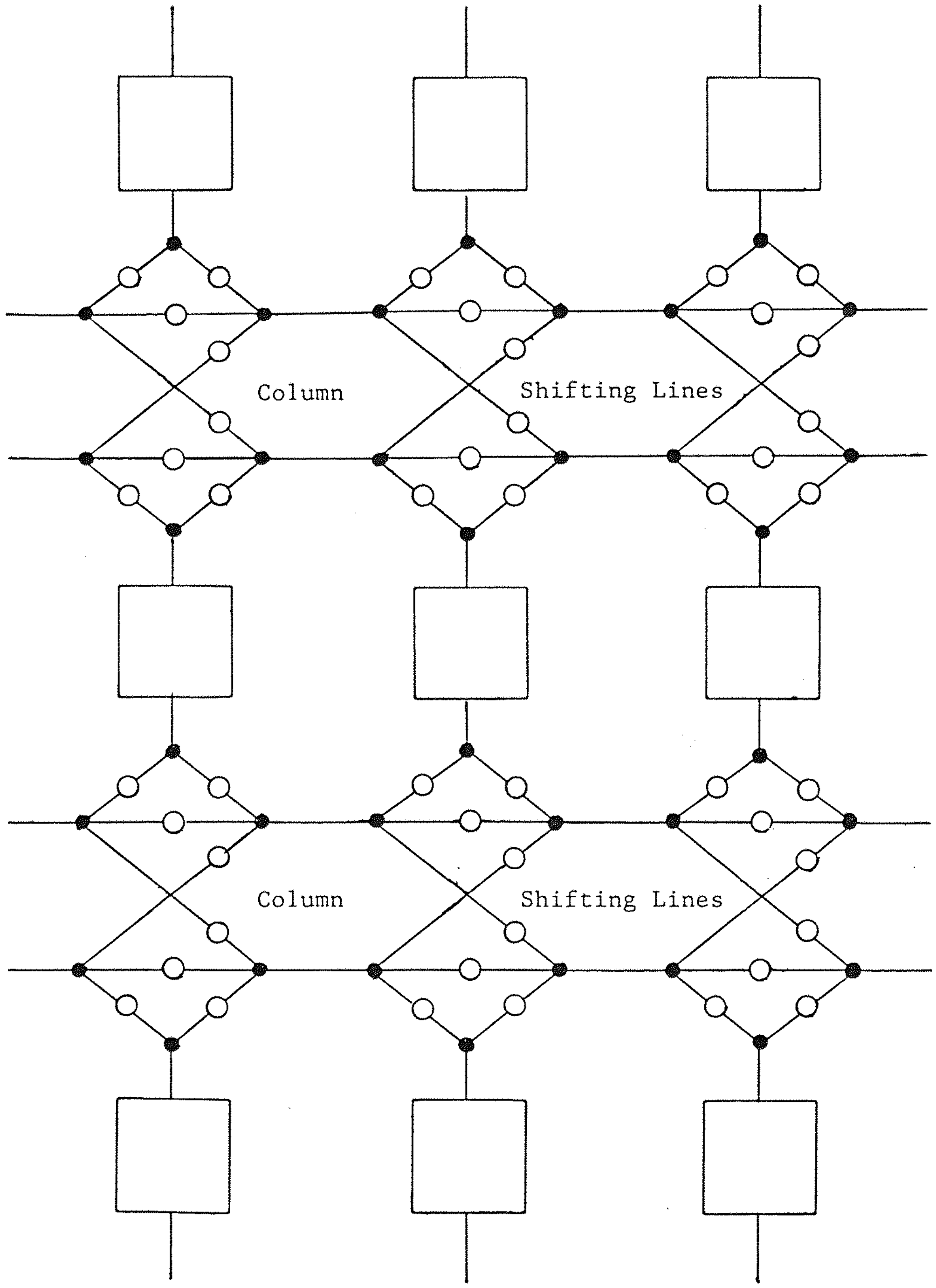
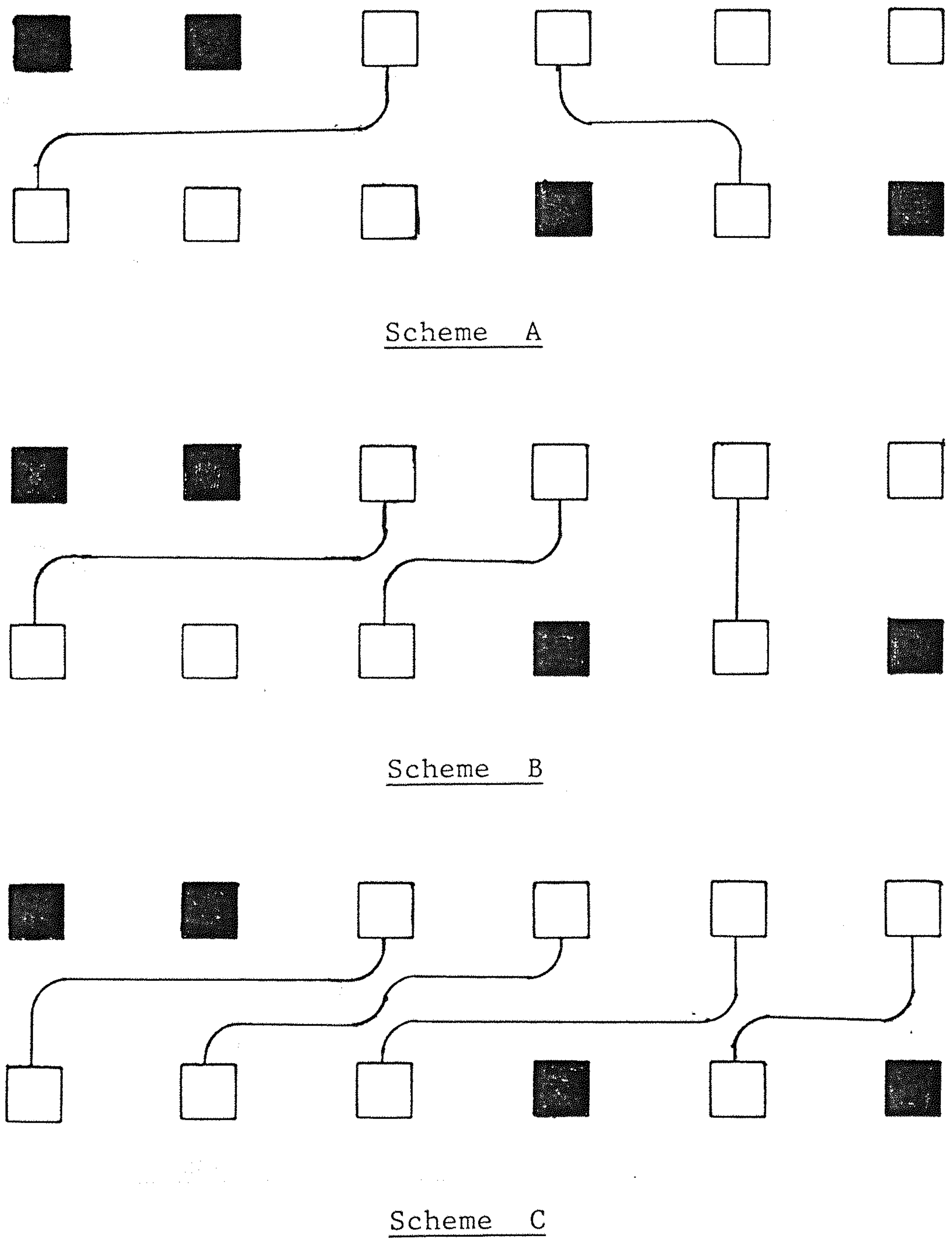


Figure 2.4 Column configuration scheme C.



Scheme A

Scheme B

Scheme C

Figure 2.5 Column configuration by schemes A, B and C on a physical array of good and bad cells.

## 2.2 TESTING OF CELLS

When configuring the intercellular connections, some of the cells in the physical array are required to be tested. The testing includes the testing of the condition of the cells and also of the connection between the cells. There are two approaches that can be taken for testing the cells. The cells can be tested either externally or locally within the array.

In external testing, the test elements are located outside the physical array. Communication lines between the test elements and cells have to be laid out. Since every cell would need access to a test element, the area overhead for the communication lines is tremendous. There is also the difficulty in laying-out these lines on the wafer. External testing is more appropriate for small size array or where the cost of the test element is high.

An approach more appropriate for a large size array is self-testing [Prad80,Soma84,Will82]. Each cell would have its own test element. Even though a self-testing array requires more test elements, there are no long communication lines as in the external testing. In this study, the self-testing approach is taken.

Testing of each devices in each cells is unpractical and almost impossible. A more suitable method of testing the cells is to test for the correct functioning of the cells. A cell is tested by observing its output after a string of data have been input into the cell. The output of the cell is then compared for compatibility with the input. Any uncompatibility between the input and output would meant a possible defect within the cell. The test sequence is selected such that it covers all the possible defects within the wafer. Therefore, a preliminary study of the possible defects within the cells must be done before the test sequence can be made.

## 2.3 CONFIGURATION OF FUNCTIONAL ARRAY

Figure 2.6 shows the three main components of a self-testing cell. The processing element (PE) is the main processor which receives, processes and distributes data within the functional array. The configuration of the interconnections for each cell is done by its control element (CE). The function of the test element (TE) is to test the PE and the inter-PE connections for defects. Signals B and G are used for the setting of the gates on the inter-PE lines for scheme A as shown in figure 2.7.

The configuration of the interconnection for a PE by its CE is based on the information received from the CE's of its top, bottom and left neighbours, and from the result of testing of the PE by its TE. The configuration also depends on the state of the cell. There are six states which the cell can be in. These states are :

1) Free (F)

This is the initial state of the cell. It has not been selected nor tested for the functional array. Its inter-PE connections have not yet been set. Depending on the instructions received, an F-state cell can change into waiting (W), testing (T) or bypassed untested (U) states. Signals B and C are both false.

2) Waiting (W)

The cell is waiting for the start of the next test cycle after which it will change to the testing (T) state. There can be only one W-state cell in each row at any one time. A W-state cell can also change to the bypassed untested (U) state on the instructions it receives before the start of the next test cycle. Signals B and G are still false.

3) Testing (T)

The PE is tested if the cell is in the W-state at the start of a test cycle. The PE is also tested if the cell is in F-state and its left neighbour has been tested bad (B-state). The result from the testing would put the cell either in the connected good (G) or bypassed bad (B) state. Signals B and G are still false.

4) Connected Good (G)

This is one of the three final states of the cell. The final states are G, B and U. The PE has been tested good and is to form part of the functional array. Signal G is true but B is false.

5) Bypassed Bad (B)

The PE has been tested bad and cannot be used in the functional array. The PE is to be disconnected from the other PE's. Signal B is true but G is false.

6) Bypassed Untested (U)

An F-state or W-state cell can be changed to this state by its CE depending on informations from the neighbouring cells. The PE is bypassed because no connection could be made into the functional array eventhough the PE may be good. Both signals B and G are false.

At the start of the configuration, all the cells in the physical array are in the F-state except for those in the left-most column which are in the W-state. The functional array is formed column by column, starting from the left in a series of test cycles.

During each test cycle, one column for the functional array is formed by connecting the left-most available good PE in each row. At the start of each test cycle, a 'start testing' (START) signal is fed into all the CE's in the physical array. This would initiate the testing of the PE's of W-state cells. There can be only one W-state cell in each row at the start of the test cycle.

If a PE is found to be bad, a true B signal is given out. The cell is put in the B-state and its PE is bypassed. The true B signal is also fed into the CE of its next right neighbour. The next right PE is tested immediately on the receipt of the true B signal without waiting for the START signal. If this PE is found to be bad, the selection is repeated to the next right PE until a good PE is found. Therefore, each test cycle may include the testing of several bad PE's but of only one good PE in each row.

After a PE has been tested good, its CE will give out a true G signal. This will connect the PE into the functional array. The CE will also give out a 'test during next cycle' (TNCY) signal to the next right cell in order to set the W-state cell for the next test cycle.

For scheme A, the PE that is directly above or below a cell that becomes G or E-state during the current test cycle cannot be used as waiting cells for the next test cycle. A cell on reaching one of these two states will give out a 'unavailable' (UNAV) signal to the top and bottom neighbours. If the cell becomes U-state during the current test cycle, the UNAV signal is delayed until the next test cycle. This is because its top and bottom neighbours could be used as waiting cells for the next test cycle but not for the test cycles following that.

For scheme B, the UNAV signal is given only when the PE is bypassed (B or U-state). The UNAV signal from a U-state cell is also delayed until the next test cycle as in scheme A. Unlike in scheme A, the cells above and below a G-state cell can be used as the waiting cell for the next test cycle. For scheme C, the UNAV signal is also given out when the cell is in the B or U-state but the signal is delayed. For UNAV from a B-state cell, it is delayed until the next test cycle. For UNAV from a U-state cell, it is delayed until after the next test cycle.

If an F-state cell received a TNCY signal from its left neighbour but did not received any UNAV signal from its top or bottom neighbour, then the cell will change to the W-state. On the other hand, if it received a TNCY signal from its left neighbour and also a UNAV signal from its top or bottom neighbour, then the PE is bypassed untested. The CE of the U-state cell will then give out a TNCY signal to the next CE on the right.

The various changes of state of the cell are illustrated in table 2.1. It shows the values of the input before the change of state and also the values of the output after the change of state. Before the cell reaches one of the three final states, no output is given out.

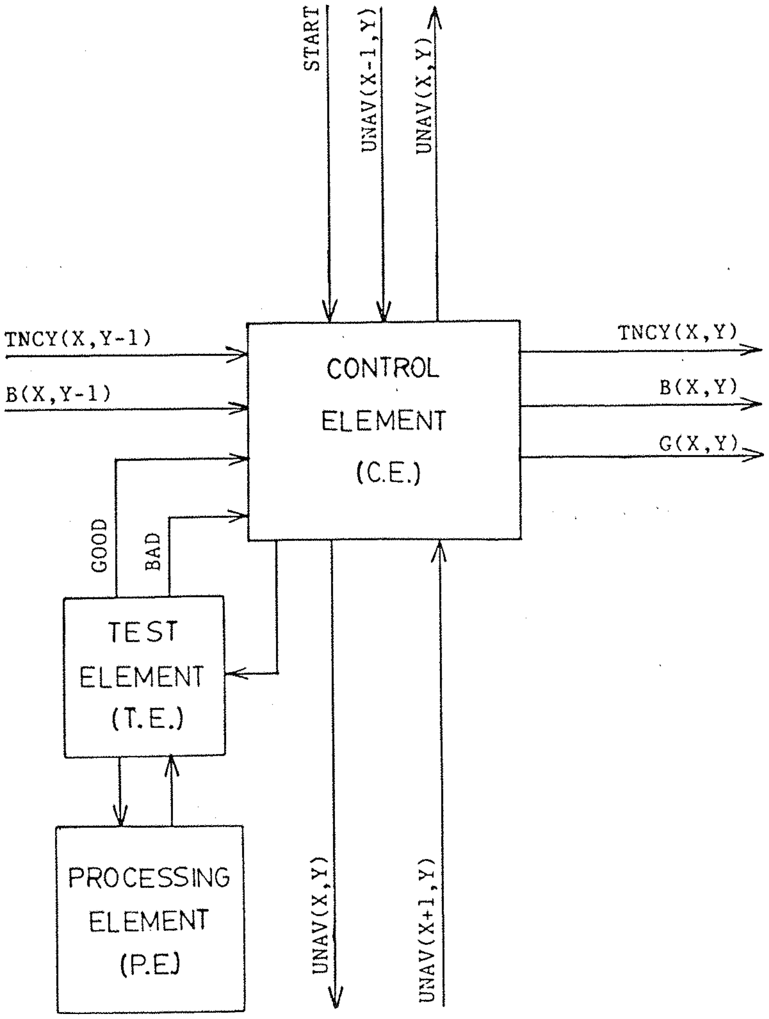


Figure 2.6 Main components of a self-testing cell for a reconfigurable orthogonal array.

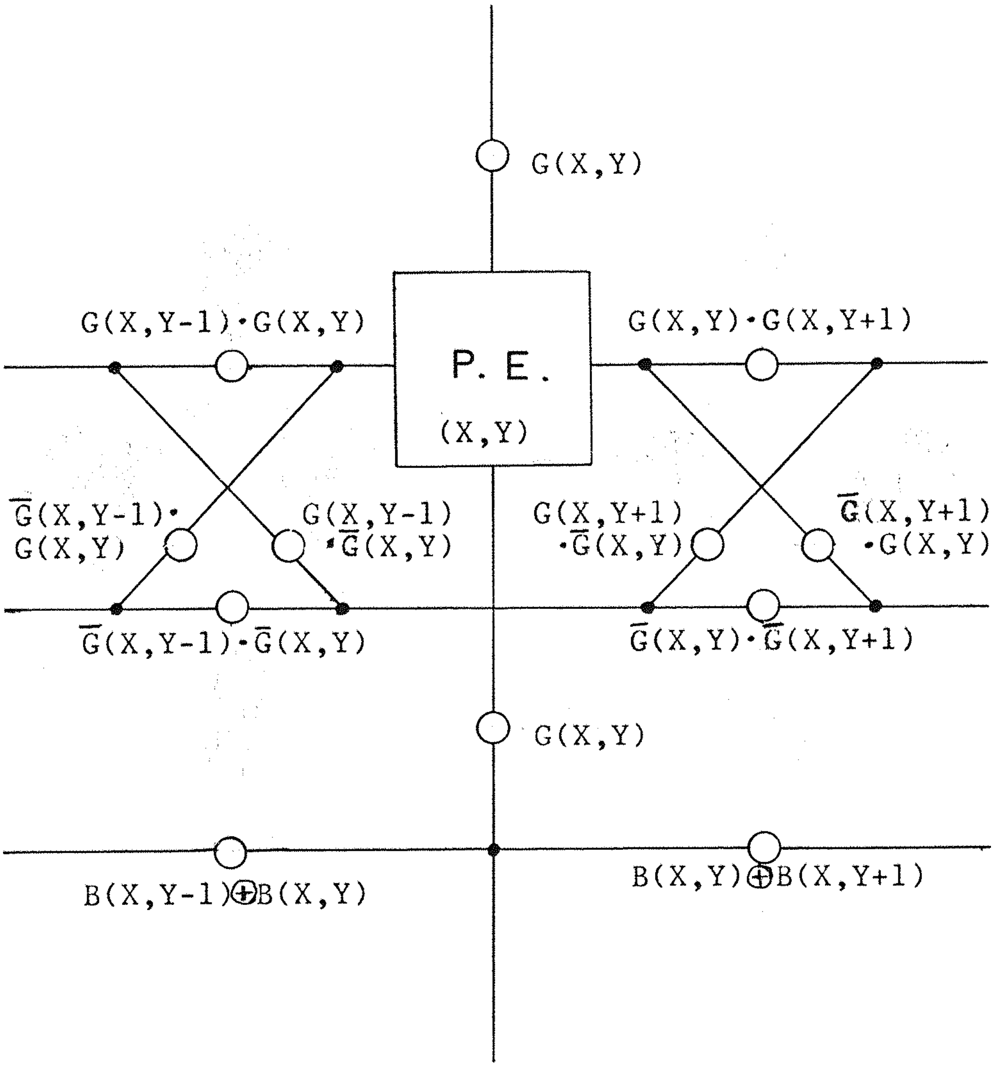


Figure 2.7 Control for the gates around a cell of interconnection scheme A.

| Initial State | Input | | | | Final State | Output | | | |
|---|---|---|---|---|---|---|---|---|---|
| | TNCY(X,Y-1) | B(X,Y-1) | OR1 | START | | TNCY(X,Y) | G(X,Y) | B(X,Y) | UNAV(X,Y) |
| F | 1 | 0 | 0 | 0 | W | 0 | 0 | 0 | 0 |
| F | 0 | 1 | 0 | 0 | T | 0 | 0 | 0 | 0 |
| F | 0 | 1 | 1 | 0 | T | 0 | 0 | 0 | 0 |
| W | 1 | 0 | 0 | 1 | T | 0 | 0 | 0 | 0 |
| T | 0 | 1 | 0 | 0 | G | 1 | 1 | 0 | 1 |
| T | 0 | 1 | 0 | 0 | B | 0 | 0 | 1 | 1 |
| T | 0 | 1 | 1 | 0 | G | 1 | 1 | 0 | 1 |
| T | 0 | 1 | 1 | 0 | B | 0 | 0 | 1 | 1 |
| T | 1 | 0 | 0 | 1 | G | 1 | 1 | 0 | 1 |
| T | 1 | 0 | 0 | 1 | B | 0 | 0 | 1 | 1 |
| F | 1 | 0 | 1 | 0 | U | 1 | 0 | 0 | 1* |
| W | 1 | 0 | 1 | 0 | U | 1 | 0 | 0 | 1* |

* Output delayed until next test cycle

Table 2.1 Values of input before and output after the changes of state of cell (x,y) with interconnection scheme A.

## 2.4 CONTROL ELEMENT

The logic design of the control element of cell (x,y) for scheme A is shown in figure 2.8. It has been drawn for simplicity in explaining the functioning of the CE and not for any particular process technology. A more appropriate IC design is shown in figure 2.9. The OR and the AND gates of figure 2.8 has been replaced by NOR and NAND gates in figure 2.9.

Refering back to figure 2.8, the testing of a W-state PE is initiated by an input to the test element (TE) from OR2. At the beginning of each test cycle, a true START pulse is fed into all the cells in the array. The main control for the start of testing of the PE is AND1. The testing is to start when TNCY(x,y-1) and START are true; while UNAV(x+1,y) and UNAV(x-1,y) are false. NOR is used so that a final state PE is not retested. The testing of the PE can also be initiated by a true B(x,y-1) input into an F-state cell. This is controlled by AND3. After the testing is completed, a true GOOD or a true BAD signal is given out by the TE depending on the result of the testing. The output is permanent and is maintained even after the configuration has been completed.

The control for a PE to be bypassed untested (U-state) is AND2. This is when TNCY(x,y-1) is true with either UNAV(x+1,y) or UNAV(x-1,y) being true. A true output from NOT2 is used to verify that the cell is not already in the G or B-states.

When a cell is in any one of the final states, the signal UNAV(x,y) is given out. This is done by OR6 which is controlled by OR3 and AND4.

OR5 is used to delay the true output of AND4 from a U-state cell until the start of the next test cycle. This is because the cells above and below a U-state cell can be a waiting cell for the next test cycle but not for the following test cycles.

The gate OR4 is used to control the output TNCY(x,y). The output is true when the cell is either in the G or U-state. A G-state cell would also produced a true G(x,y). The output B(x,y) is true when the cell is in the B-state.

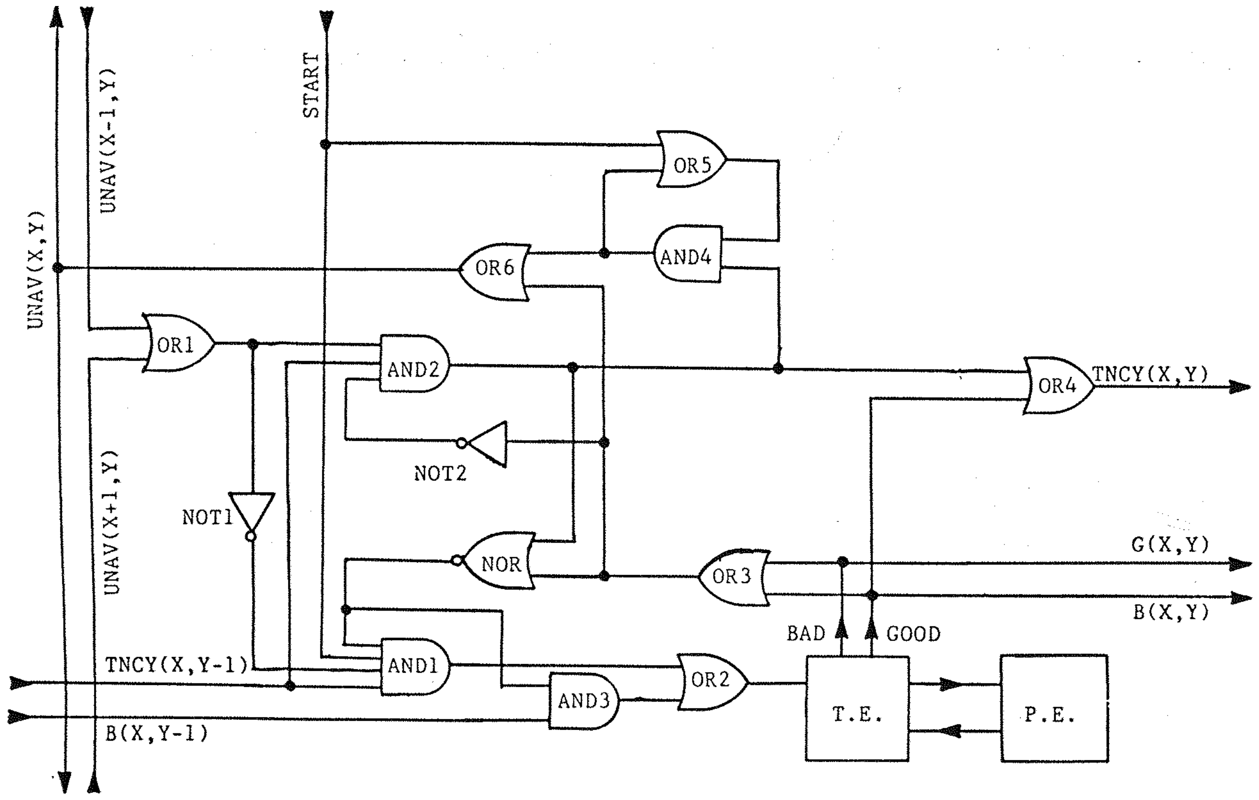


Figure 2.8 Logic design for the control element of interconnection scheme A.

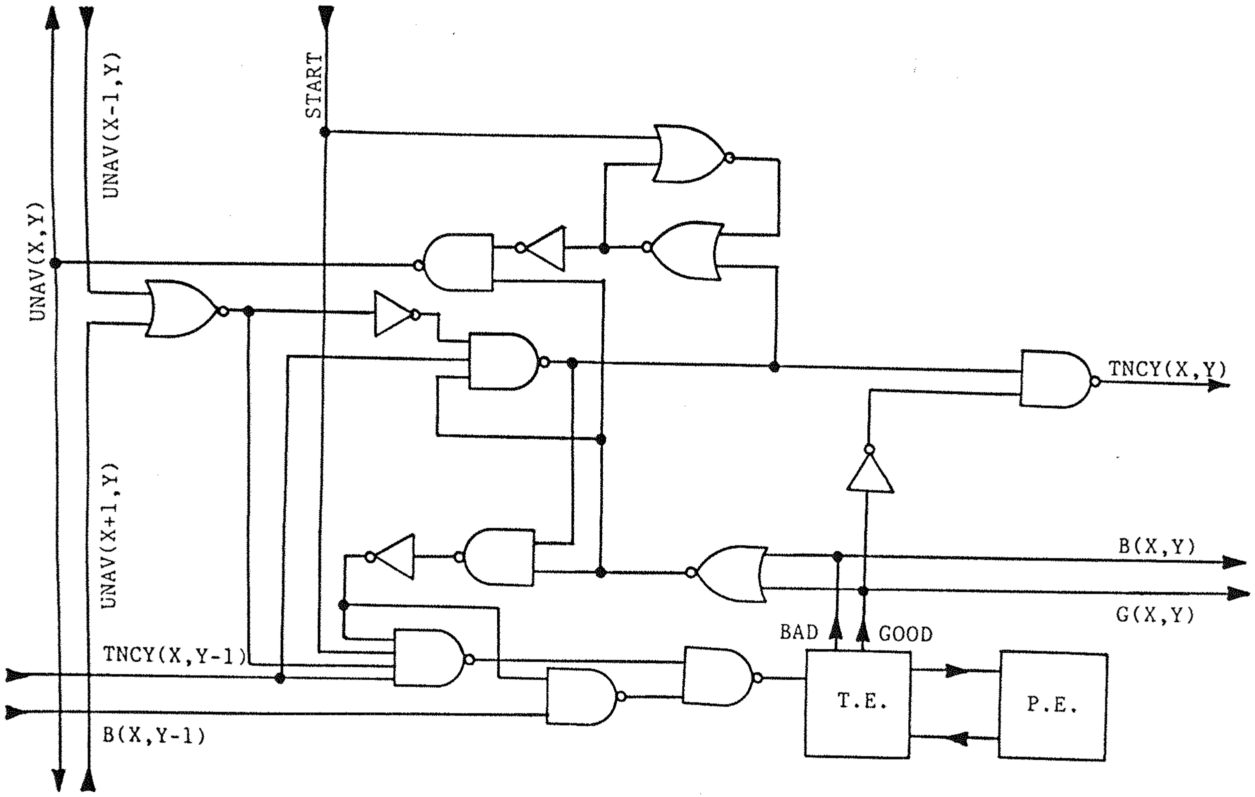


Figure 2.9 Layout-ready design of the circuit in figure 2.8.

Chapter 3

# YIELD SIMULATIONS

The interconnection schemes that have been presented in chapter two were evaluated using computer simulations. The functional arrays were configured on physical arrays with different amount of defective cells. The way in which the defective cells were distributed in the physical arrays is discussed in section 3.1. The results of the yield simulations of the interconnection schemes are presented in section 3.2. Some modifications of the interconnection schemes were also tested. The results of the yield simulations of the modified schemes are presented in section 3.3.

## 3.1 WAFER DEFECTS

The model for the cellular array with some defective cells can be made by studying the defect distribution on a real wafer. Defects on a wafer consist of various types, and cause various degree of damage to the wafer [Mang84a,Pelt83,Stap80].

Large scale defects which effect large area of the wafer are generally fatal and unrepairable. This type of defect are mainly caused by incorrect processing or missalignment of the photomasks. It also includes scratched, chipped or broken wafer. However, this type of defect is not very common.

The more common type of defect is point defect. This includes shorts, broken connections, and also spikes and pinholes in the various layers of the wafer. Point defects are commonly caused by dirt from the enviroment which gets onto the wafer or the photomask.

Point defects are generally distributed randomly throughout the wafer, but there is also a tendency for them to cluster. A cause for defect clustering may be aggregate of dirt that has collected during processing [Stap83]. When shaken loose, these clump of particles would form a cloud which would later settled on the wafer.

Beside defect clustering, there is also non-uniformity of the defect distribution in the radial direction. It has been found that there is a higher number of defects at the edge of the wafer than at the centre [Paz77,Yana72]. Some of the reasons for the radial variation are electrostatic attraction of the dirt to the edge of the wafer, radial temparature variation and bad handling of the wafer.

It has also been found that there can be a slight increase in defect density at the centre of the wafer [Perl81]. The increase in defect density could be caused by the resist being thicker at the centre of the wafer. An example from IC chip production which illustrates this type of variation is shown in figure 3.1.

It is difficult to produce a model which correctly reproduces the joint-effect of random defect distribution, defect clustering and radial defect variation. For a simple model, two assumptions can be made. Firstly, it is assumed that parts of the wafer other than the processing elements (PE's) are fault-free. These fault-free parts includes the control elements, the test elements, the interconnection lines and the various interconnection gates. Generally, the area occupied by these components are small as compared with the area of the PE's. The yield of fault-free cell with the PE and other components together can be expected to be only slightly lower than the yield of good PE alone. However, a detailed comparision of the layout area and of the defect distribution in the various components of each interconnection schemes should be done in future in order to do a more accurate modelling of the yield.

The second assumption is that the defective PE's are randomly distributed throughout the wafer. Each PE in the physical array would have the same probability of being defective. The effect of defect clustering and radial defect variation are not considered because of non-availability of experimental data from processing of WSI chips. However, this second assumption can still be valid if the following additional assumptions are made.

All the PE's are assumed to be located in a central region of the wafer that is about half the size of the wafer. Looking at figure 3.1, the radial variation of defective PE within this region is small and assumed to be negligible. As for defect cluster, it is assumed that the size of the cluster is small as compared with the size of the PE. It would be unlikely for the defect cluster to cause a cluster of defective PE's.

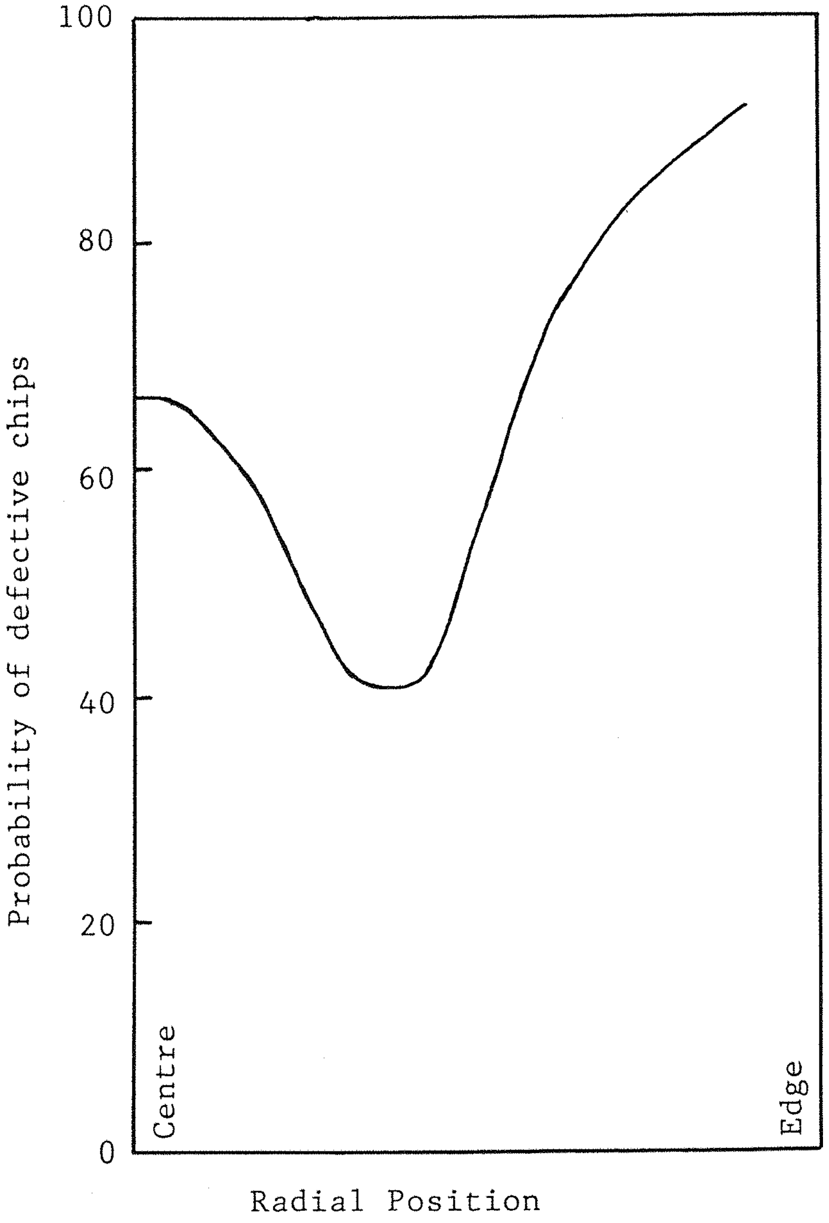


Figure 3.1 Radial variation in the probability of being defective for IC chips on a wafer. (From Perl81)

## 3.2 YIELD SIMULATIONS OF THE BASIC SCHEMES

The yields for the various interconnection schemes were obtained from computer simulations of the configuration of the functional array using the various schemes. The interconnection schemes used were the three column configuration schemes A, B and C with row configuration scheme 2. The algorithm for the configuration using the three schemes is shown in appendix I. This includes the flow chart and the full program for the configurations. First, a physical array of good and bad cells is formed. This is done by using a pseudorandom number generator which generates numbers between 0.0 and 100.0. For a given cell yield of Y %, a cell is good if the number generated for it is less than or equal to Y. Otherwise, the cell is bad.

After the physical array has been formed, the functional array is then configured on the physical array. Columns of functional cells are formed one by one starting from the left side of the physical array. Starting from the top of the array, the left-most available good cell in each row is chosen for each functional column.

In this thesis, the yield for a particular interconnection scheme was obtained by doing 1000 simulations of the configuration of a 16X16 functional array using the particular scheme. The size of the array on which the configurations were done was chosen such that there were more than enough redundant columns for all the configurations.

For each simulation, the size of the configured array was recorded. The yield for a particular physical array size was obtained by counting the number of configured array with size less or equal to the particular array size. The number obtained is then divided by 10 to give the percentage for the array yield.

Another value given together with the array yield as the results of the yield simulations is the cell utilisation. The cell utilisation is defined as the percentage of the total number of cells used in the configured arrays out of the total number of cells in both the configured and unconfigurable arrays. It is calculated from the array yield by using the equation:

$$U = YF/P$$

where U is the cell utilisation, Y is the array yield, F is the number of cells in the functional array, and P is the number of cells in the physical array.

A scheme with a high array yield is not good if its cell utilisation is low. This is because low utilisation means a high amount of redundancy is used. Therefore, the best yield does not necessary mean the highest array yield but does mean the highest cell utilisation. Throughout this thesis, the terms 'best yield' and 'highest cell utilisation' are treated as synonymous.

The results for the yield simulation of the three basic schemes are shown in figures 3.2 and 3.3 for cell yields of 65 and 95 % respectively. The figures show the array yield and the cell utilisation at various physical array sizes. The different array sizes are shown as the percentage of redundancy in the physical arrays. Table 3.1 lists the array size, array yield and cell utilisation of the best yield for each scheme at different cell yield.

As has been suggested earlier, the interconnection schemes C produced the best yield. This is followed by scheme B and then by scheme A. Looking at the result in table 3.1, the difference in yield of scheme B and C becomes less with increasing cell yield. At very high cell yield, scheme B is almost as good as scheme C. The yield for scheme A is much poorer than for scheme B even at very high cell yield.

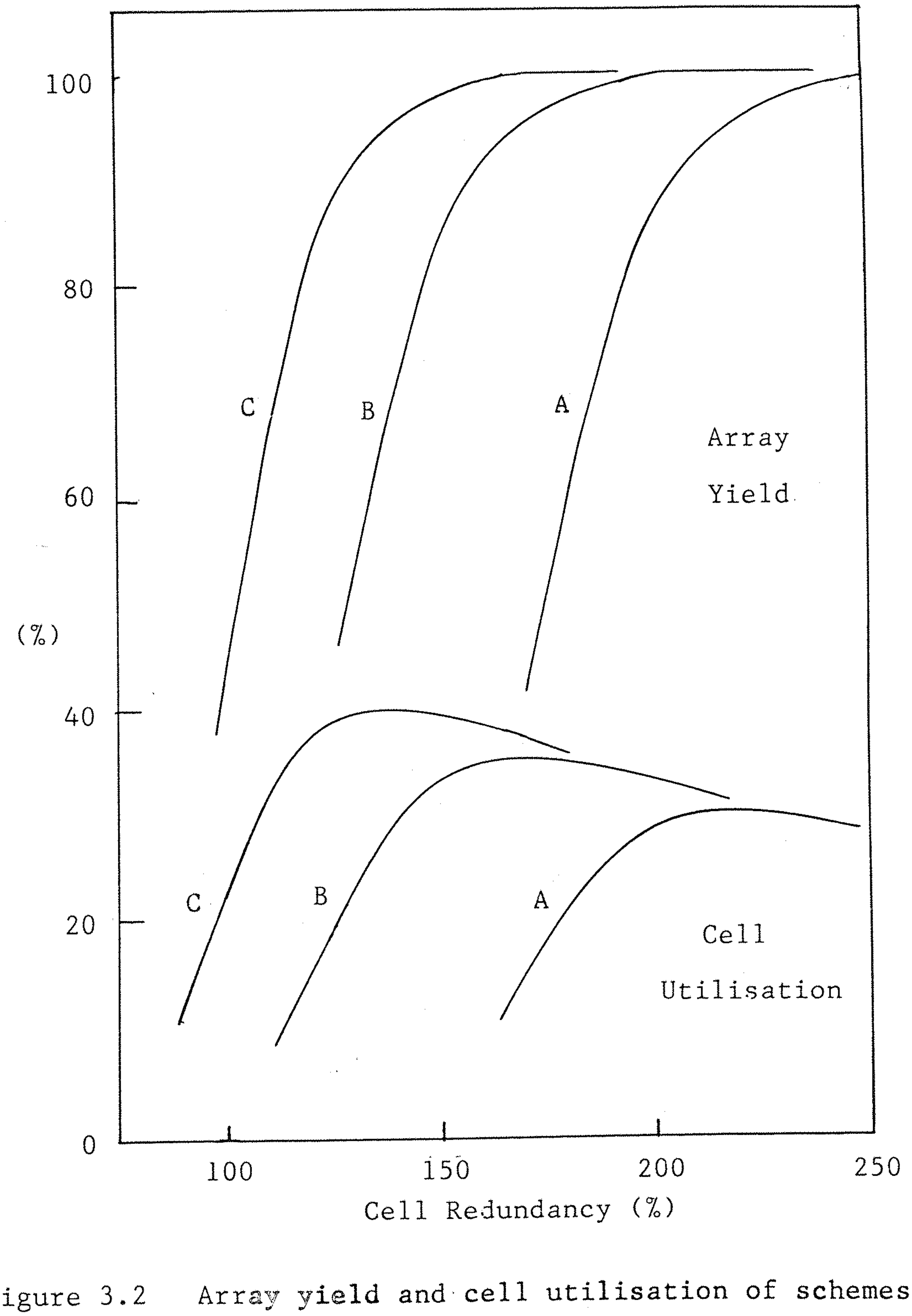


Figure 3.2 Array yield and cell utilisation of schemes A, B and C as a function of the amount of cell redundancy in a physical array with 65 % cell yield.

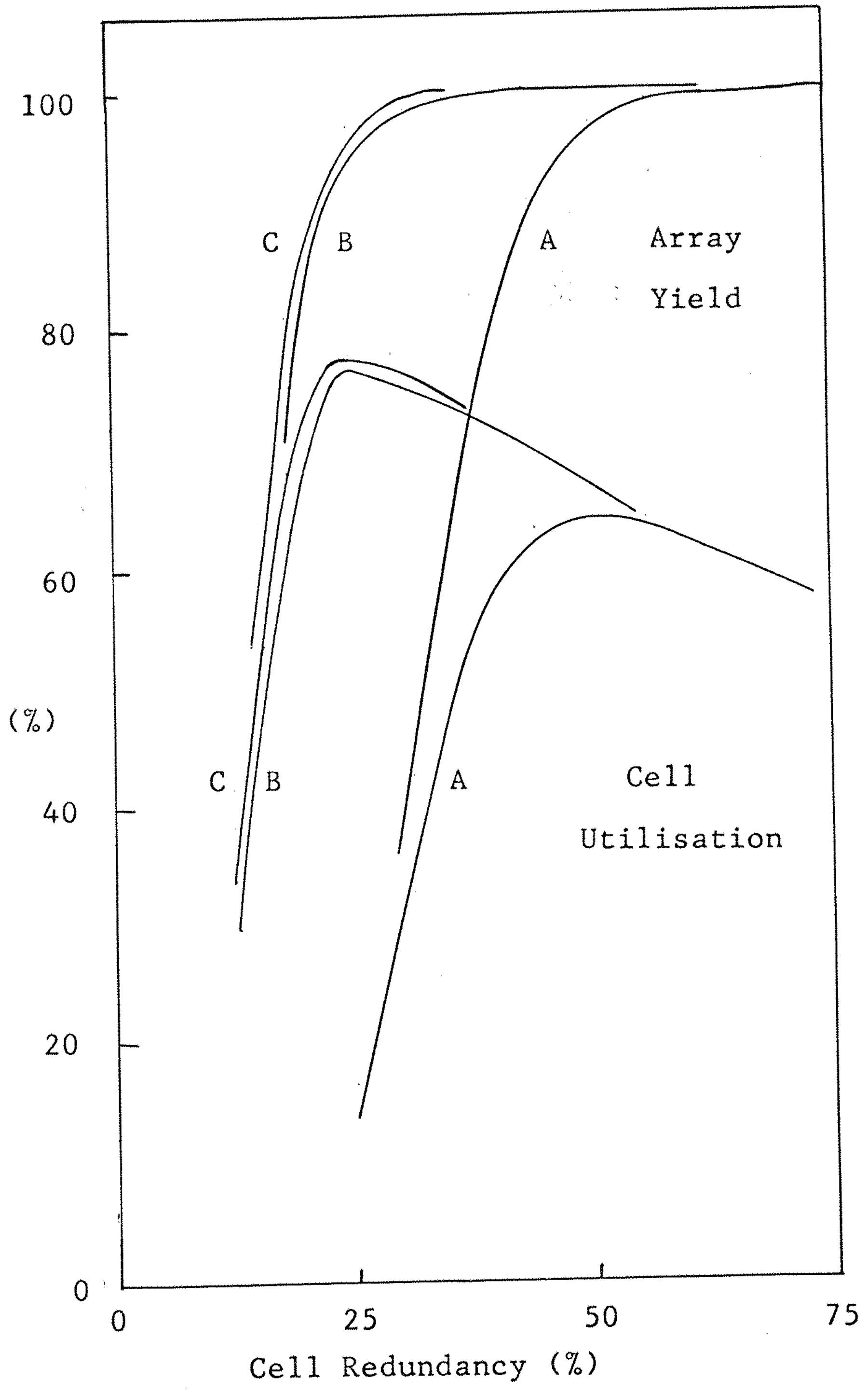


Figure 3.3 Array yield and cell utilisation of schemes A, B and C as functions of the amount of cell redundancy in a physical array with 95 % cell yield.

| Cell Yield | Scheme | Array Size | Array Yield | Cell Utilisation |
|---|---|---|---|---|
| 65 | A | 16X52 | 98 | 30 |
| 65 | B | 16X43 | 95 | 35 |
| 65 | C | 16X38 | 95 | 40 |
| 80 | A | 16X36 | 96 | 43 |
| 80 | B | 16X30 | 95 | 51 |
| 80 | C | 16X28 | 96 | 55 |
| 95 | A | 16X24 | 97 | 65 |
| 95 | B | 16X20 | 95 | 76 |
| 95 | C | 16X20 | 97 | 77 |

Table 3.1 Size and yield of array producing the highest cell utilisation of the various schemes at different cell yield.

## 3.3 YIELD SIMULATIONS OF MODIFIED SCHEMES

Several modifications to the three basic interconnection schemes were also tested. These modifications were used in order to try to improve the yields of the basic schemes and also to try to limit the length of the intercellular connections.

As they are, the basic interconnection schemes do not put any limit on the length of the interconnection lines. Long interconnection lines would have huge delay, not just from the long length but also from the high number of gates on the lines. The delay on an interconnection line is roughly proportional to the square of the number of gates on the line. There are several ways of reducing the delay such as by adding buffers on the line but the best reduction that could be obtained is to reduce the proportionality factor to the number of gates on the line. Therefore, there must be limits on the length of the interconnection lines and on the number of gates in order to limit the delay.

The number of gates on the interconnection lines can be restricted by limiting the number of PE's being bypassed. For an inter-PE connection line with N bypassed PE's, the number of gates on the interconnection line is N+1. This is true for all the configuration schemes where N is greater than 2 except for the column configuration scheme A which requires N+2 gates. Additional control circuit would be required to keep track of the number of PE's being bypassed. The simulations of the configurations with this modification were done by modifying the algorithm in appendix I to record the number of bypassed PE's.

When limiting the number of gates on the interconnection lines, the array yield can be expected to be reduced. This reduction can be seen in figure 3.4. It is the result from 1000 simulations of the configuration using scheme B at 80 % cell yield. The limits used were 5 and 8 gates.

Another way of indirectly limiting the interconnection length is to use the two-level hierarchy as tried by Hedlund [Hedl82]. The physical array is divided into smaller blocks of cells. A small functional array is formed within each blocks. The whole functional array is then formed by connecting the small functional array together. The limit on the interconnection length is set by the size of the blocks. No circuit is required to control the length of the interconnection lines but additional circuit would be required for the inter-block connection.

The array sizes used for the two-level hierarchy modification were 2X2 cells in 8X8 blocks, 4X4 cells in 4X4 blocks and 8X8 cells in 2X2 blocks. For each blocks, 1000 simulation were made for each scheme at 65 % cell yield. In each simulation, the number of redundant columns in the configured block was recorded. From this result, the block yield can then be obtained. The wafer yield was then obtained by raising the block yield to the number of blocks in the physical array.

The yield obtained for scheme B at 65 % cell yield with the two-level hierarchy modification is shown in table 3.2. It lists the array size, the array yield and cell utilisation of the best yield. The yield obtained is poorer than the yield obtained with the basic scheme. Smaller sized block was found to produce lower cell utilisation.

Examples of the configurations using the three basic interconnection schemes at 80 % cell yield are shown in appendix I. Two useful features can be seen from the examples. Firstly, it can be seen that most of the long interconnection lines are from the PE's in the right-most functional column to the right edge of the array. The delay on these lines could be reduced if the PE's were given direct access to the edge. Secondly, among these long interconnection lines, the length of the lines for the top few rows and bottom few rows are usually longer than the length for the other rows. The long lines for the top few and bottom few rows could be reduced by removing some of the cells in the top and bottom right corners of the physical array.

To allow for the PE's in the right-most functional column to have direct access to the right edge, an extra bypass line is added to the row configuration scheme. A trial of this modification was made by modifying the row configuration scheme 2. The modified scheme is shown in figure 3.5. Five of the PE's can be connected directly to the extra bypass line such that the number of gates on the interconnection lines for these PE's to the right edge is just two. The extra bypass line also reduces by five the number of gates on the interconnection lines to the right edge for the other PE's in the row.

The yield with the highest cell utilisation obtained using the extra bypass line modification is shown in table 3.3. It is compared with the yield obtained without the extra bypass line modification and also with the yield obtained with the basic scheme. At a high cell yield of 95 %, there is no difference in the cell utilisation among the various modifications since the number of bypassed cells is small.

The difference in yield becomes more significant at lower cell yield. Limiting the interconnection length does greatly reduced the yield. The extra bypass line modification helps to increase the yield but not as high as when without the interconnection length limitation.

Another modification is to remove a few PE's from the top right and the bottom right corners of the physical array. This is to take advantage of the fact that the right-most functional cells in the top few and bottom few rows have longer interconnection lines to the right edge of the array than those of the other rows. Therefore, the PE's at the top right and at the bottom right corners of the physical array are seldom connected into the functional array. Simulations were done on physical arrays with one, three and six cells removed from the two corners. The result for this modification is shown in table 3.4. No significant change in yield was found for this modification when only one or three cells were removed from the corners. However, the yield was reduced when six cells were removed from the corners.

Another modification that was tried was to extend the one-level redundancy of the basic interconnection schemes to two-level redundancy by adding redundant rows of PE's. A row bypass scheme is used to bypass any unrequired rows. In this way, functional rows with long interconnection lines or functional rows that require more PE's than there are available in the physical rows, can be bypassed and not connected to the functional array. Furthermore, rows with defects on the interconnection lines, in the control elements or in the test elements can also be bypassed.

The results for the redundant rows modification are shown in tables

3.5 and 3.6. The number of redundant rows tested were one, two, three and four. The tables show that the redundant row modification does improved the yield. The best yield for each schemes were obtained by having just one extra row. Even with two or three redundant rows, the yields obtained were better than the yields for the basic schemes.

All of the above simulations used arrays with randomly distributed defective PE's. The last set of simulations tried was to test physical arrays with cluster of defective PE's. It was tried in order to observe the effect of defect clusters on the yield.

No model was available for the formation of an array with defect clustering. A model was developed based on the fact that a cell would have a high probability of being defective if it has a high number of defective neighbours. To form an array of a required cell yield with defect clustering, a randomly distributed array with half the defect probability is first formed. Some of the good cells are then changed to bad according to the number of bad cells among their eight nearest neighbours such that the overall probability of having a good cell is equivalent to the required cell yield. A positive linear relationship is assumed between the probability of change from good to bad and the number of bad neighbours.

The algorithm for the formation of an array using this kind of distribution is shown in appendix II. A more detailed explanation of the formation of the physical array with defect cluster and an example of such array are also presented in appendix II. Eventhough there is no experimental data to support the model, the physical array formed does have more clusters of defective cells than the random defect distribution. The result for the simulations using this defect distribution is shown in table 3.7. The yields obtained for the basic schemes using this distribution is slightly lower than the yield obtained using the random distribution. The different in yield becomes greater when the number of gates limitation is imposed on the simulations.

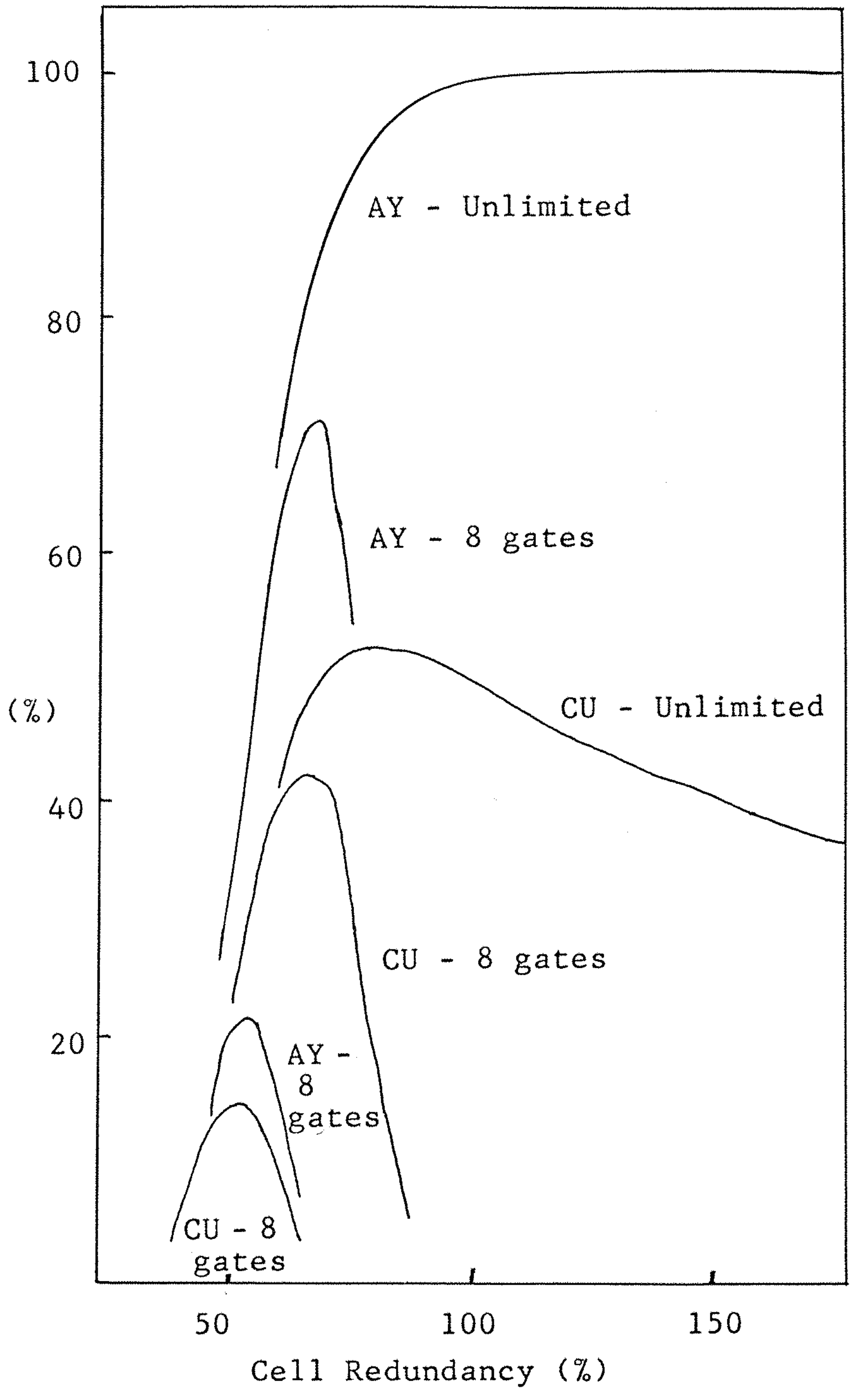


Figure 3.4 Array yield (AY) and cell utilisation (CU) of scheme B at 80 % cell yield with 5 and 8 gates limitation, and with unlimited number of gate.

| Scheme | No. of Block | Block Size | Array Yield | Cell Util. |
|---|---|---|---|---|
| A | 2X2<br>4X4<br>8X8 | 8X27<br>4X17<br>2X10 | 92<br>98<br>94 | 27<br>23<br>19 |
| B | 2X2<br>4X4<br>8X8 | 8X24<br>4X14<br>2X10 | 94<br>91<br>99 | 31<br>26<br>20 |
| C | 2X2<br>4X4<br>8X8 | 8X22<br>4X14<br>2X10 | 94<br>94<br>99 | 34<br>27<br>20 |

Table 3.2 Yields of the various schemes with two-level hierarchical modification at 65 % cell yield.

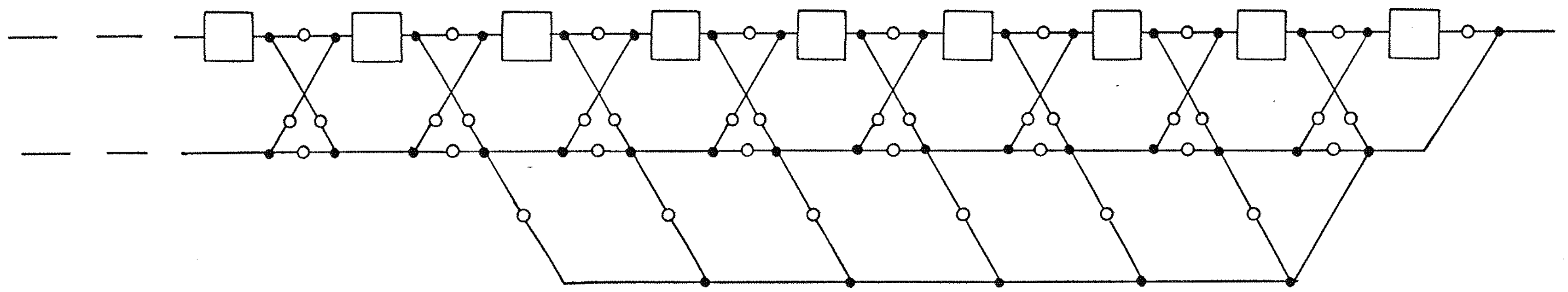


Figure 3.5 Extra bypass line modification of row configuration scheme 2.

| Modification | Cell Yield | Array Size | Array Yield | Cell Util. |
|---|---|---|---|---|
| 5 gates limit | 80<br>95 | 16X24<br>16X20 | 20<br>98 | 13<br>78 |
| 8 gates limit | 65<br>80<br>95 | 16X34<br>16X26<br>16X20 | 20<br>68<br>98 | 9<br>42<br>78 |
| Extra bypass line with 5 gates limit | 80<br>95 | 16X26<br>16X20 | 49<br>98 | 30<br>78 |
| Extra bypass line with 8 gates limit | 65<br>80<br>95 | 16X37<br>16X28<br>16X20 | 54<br>90<br>98 | 23<br>52<br>78 |
| Unlimited number of gates (No modification) | 65<br>80<br>95 | 16X42<br>16X29<br>16X20 | 96<br>95<br>98 | 37<br>52<br>78 |

Table 3.3 Yields for scheme B at various cell yields with the number of gates limitation and with the extra bypass line modification.

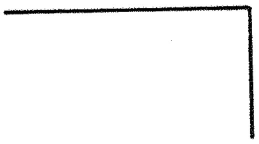
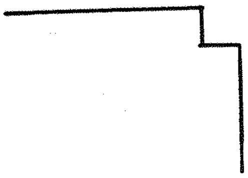
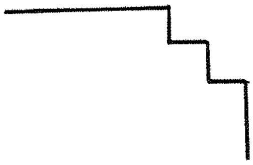
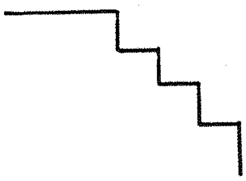
| Shape of corner | Unlimited gate | | | 5 gates limit | | |
|---|---|---|---|---|---|---|
| | Array Size | Array Yield | Cell Util. | Array Size | Array Yield | Cell Util. |
|  | 16X29 | 95 | 53 | 16X25 | 26 | 17 |
|  | 16X29 | 95 | 53 | 16X25 | 26 | 17 |
|  | 16X30 | 97 | 53 | 16X25 | 22 | 15 |
|  | 16X30 | 97 | 51 | 16X26 | 17 | 11 |

Table 3.4 Yields of scheme B at 80 % cell yield with various corner modifications.

| Scheme | No. of rows | Array Size | Array Yield | Cell Util. |
|---|---|---|---|---|
| A | 16 | 16X52 | 98 | 30 |
| | 17 | 17X46 | 96 | 31 |
| | 18 | 18X45 | 96 | 30 |
| | 19 | 19X45 | 96 | 29 |
| | 20 | 20X43 | 94 | 28 |
| B | 16 | 16X43 | 95 | 35 |
| | 17 | 17X38 | 94 | 37 |
| | 18 | 18X37 | 94 | 36 |
| | 19 | 19X37 | 94 | 34 |
| | 20 | 20X36 | 95 | 34 |
| C | 16 | 16X38 | 95 | 40 |
| | 17 | 17X32 | 93 | 44 |
| | 18 | 18X32 | 97 | 43 |
| | 19 | 19X32 | 97 | 41 |
| | 20 | 20X29 | 92 | 41 |

Table 3.5 Yields of the various schemes with extra row modification at 65 % cell yield.

| Scheme | No. of rows | Array Size | Array Yield | Cell Util. |
|---|---|---|---|---|
| A | 16 | 16X24 | 97 | 65 |
| | 17 | 17X23 | 98 | 64 |
| | 18 | 18X22 | 95 | 61 |
| | 19 | 19X22 | 95 | 58 |
| | 20 | 20X22 | 97 | 56 |
| B | 16 | 16X20 | 95 | 76 |
| | 17 | 17X19 | 98 | 78 |
| | 18 | 18X18 | 95 | 75 |
| | 19 | 19X18 | 95 | 71 |
| | 20 | 20X18 | 98 | 70 |
| C | 16 | 16X20 | 97 | 77 |
| | 17 | 17X18 | 96 | 80 |
| | 18 | 18X18 | 99 | 78 |
| | 19 | 19X18 | 99 | 74 |
| | 20 | 20X17 | 98 | 74 |

Table 3.6 Yield of various schemes with extra row modification at 95 % cell yield.

| Scheme | Cell Yield | Random Dist. | | | Clustering Dist. | | |
|---|---|---|---|---|---|---|---|
| | | Array Size | Array Yield | Cell Util. | Array Size | Array Yield | Cell Util. |
| A | 95 | 16X24 | 97 | 65 | 16X24 | 97 | 65 |
| | 80 | 16X36 | 96 | 43 | 16X36 | 95 | 42 |
| B | 95 | 16X20 | 95 | 76 | 16X21 | 98 | 75 |
| | 80 | 16X30 | 95 | 51 | 16X31 | 96 | 50 |
| C | 95 | 16X20 | 97 | 77 | 16X20 | 95 | 76 |
| | 80 | 16X28 | 96 | 55 | 16X28 | 94 | 54 |
| B with 5 gates limit | 95 | 16X20 | 98 | 78 | 16X21 | 98 | 74 |
| | 80 | 16X24 | 20 | 13 | 16X24 | 9 | 6 |

Table 3.7 Comparision of yields obtained using the random defect distribution and using the clustering defect distribution.

Chapter 4

# DISCUSSION AND CONCLUSION

The results from the yield simulations of the various interconnection schemes are summerised in section 4.1. Possible area for future research which would complement the study in this thesis is dicussed in section 4.2. Some final concluding statements are presented in section 4.3.

## 4.1 SUMMARY OF RESULTS

Three schemes for configuring an orthogonal array of functional processing elements (PE's) were presented. The functional array was configured on a physical array consisting of good and bad PE's. For simplicity, only the columns were shifted during the configuration of the functional array. There was no shifting of rows and thus, the number of rows was the same in the physical and the functional arrays. Therefore, only redundant columns of cells were required. Another strategy used was to have the schemes capable of configuring long interconnection lines among the PE's. In this way, a PE can be connected to a wide choice of PE's.

The result of computer simulations of the yields obtained using the three schemes is shown in table 4.1. It shows the array size and the array yield which produce the highest cell utilisation for each scheme and at different cell yields. The yields are also compared with the results from other currently available interconnection schemes

[Evan85,Hedl82,Mann75,Sami83]. The yields for Evans' and Sami's schemes were derived from the graphical results presented by them. It is clear from the table that the three schemes are capable of producing better yields than or at least comparable yields with the other schemes.

Among the various interconnection schemes, Hedlund's scheme has the highest degree of connectability. However, its yield is low because of the difficulty in configuring the interconnections. Hedlund used the two-level hierarchy to simplify the configuration by dividing the functional array into several smaller sub-arrays. This modification requires a high number of redundant cells and thus, a lower cell utilisation.

On the other hand, Sami's schemes which have easy configuration, have low connectability among the PE's. This produces a much lower yield. Therefore, the right balance between high connectability and easy configuration is required in order for an interconnection scheme to have a good yield.

The only scheme that is able to produce a comparable yield with the three schemes is the scheme of Evans et al. This scheme also follow the strategy of using one-dimensional redundancy for easy programming. However, the amount of connectability among the cells is slightly less for Evans' scheme than for scheme b. Its yield is expected to be slightly less than the yield for scheme B.

Table 4.1 also shows that there were differences in yield among the three schemes been studied. Scheme C produced the highest yield and this was followed by scheme B and then by scheme A. At high cell yield, the yield of scheme B is almost as good as the yield of scheme C. Scheme B could be better than scheme C when the bigger area for the control elements and for the interconnection lines of scheme C is considered. The difference in yield between scheme A and B is wide even at high cell yield.

Beside the three basic schemes, various modifications of the schemes were also tested. These modifications were tried in order to find ways of improving the yield and also of limiting the interconnection length without reducing the yield too much. One modification was also tried to simulate the effect of high defect clustering.

The length of the interconnection line may be required to be limited due to excessive delay on long interconnection line. When this requirement was imposed, the yield was found to be reduced. A way to improve on the reduced yield is to use extra bypass line modification on the row configuration scheme. This allows for the PE's in the last few columns to the right to have direct access to the right edge. This produced a higher yield but not as high as when without the interconnection length limitation.

However, it is found that there is not much differences in yield between the basic schemes and the above modifications at high cell yield. This can be seen in table 3.3. The number of bad cells is small at high cell yield and thus, the number of bypassed cells is also small. Therefore, it is not necessary to have any modification to limit the interconnection length when the cell yield is high.

Another modification called the hierarchical modification which can indirectly limit the interconnection length by the size of its sub-arrays was also tried. This modification also produced a reduced yield. The best yield for the modification was found when the sub-array physical size was big. This meant that there were long interconnection lines within the sub-array. This can be seen from the large number of redundant columns required for the sub-array, as shown in table 3.2. Therefore, the hierarchical modification does not help to limit the interconnection length.

A modification that can improve the yield of the basic schemes is the extra row modification. It was found that the yields for all the three schemes can be improved by having a few redundant rows of cells. The best yields were obtained by having just one redundant row.

Another modification that was tried was the corner modification. A few cells from the top-right and bottom-right corners of the physical array were removed. No significant change in yield was found when only a few cells were removed from the corner. When more cells were removed, the yield was found to be reduced.

A theoretical model has also been developed to test for defect clustering. It showed slight reduction in yield from that obtained with random defect distribution. Eventhough the defect clustering model may not be accurate, it did showed that a lower yield could be expected when a true model for the defect distribution on the wafer has been developed.

| Scheme | Functional Array | Cell Yield | Array Size | Array Yield | Cell Util. |
|---|---|---|---|---|---|
| Scheme A | 16X16 | 65 | 16X52 | 98 | 30 |
| | | 80 | 16X36 | 96 | 43 |
| | | 95 | 16X24 | 97 | 65 |
| | | 97.5 | 16X21 | 95 | 72 |
| | | 98.5 | 16X20 | 96 | 77 |
| Scheme B | 16X16 | 65 | 16X53 | 95 | 35 |
| | | 80 | 16X30 | 95 | 51 |
| | | 95 | 16X20 | 95 | 76 |
| | | 97.5 | 16X19 | 98 | 83 |
| | | 98.5 | 16X18 | 96 | 85 |
| Scheme C | 16X16 | 65 | 16X38 | 95 | 40 |
| | | 80 | 16X28 | 96 | 55 |
| | | 95 | 16X20 | 97 | 77 |
| | | 97.5 | 16X19 | 98 | 83 |
| | | 98.5 | 16X18 | 96 | 85 |
| Evans' | 10X10 | 65 | 10X32 | 100 | 32 |
| | | 80 | 10X19 | 100 | 53 |
| | | 95 | 10X12 | 100 | 83 |
| Hedlund's | 16X16 | 65 | (3X4)(9X9) | 97 | 25 |
| Sami's | 15X15 | 95 | 15X17 | 48 | 42 |
| | | 97.5 | 15X17 | 92 | 81 |
| | | 98.5 | 15X17 | 98 | 86 |
| Manning's | 16X16 | 97.5 | 25X25 | 100 | 41 |
| | | 98.5 | 20X20 | 100 | 64 |

Table 4.1 Comparision of the yields of schemes A, B and C with some currently available interconnection schemes.

## SUGGESTIONS FOR FUTURE WORK

In section 2.4, the control element (CE) for scheme A has been presented. Eventhough the CE functions correctly as an individual cell, no simulation of the working of the CE's in an array has been done. What has been done were 'paper and pencil' simulations of small arrays of less than ten cells. Computer simulations for bigger size array are needed to be done in order to be sure of the correct functioning of the CE's.

Another area to be looked at is the control elements for scheme B and C. A CE that could set the states of the cells can be easily made by slight modification of the CE of scheme A. However, the difficult part is in programming the gates on the interconnection lines. This difficulty in designing the CE's for schemes B and C has been due to the higher number of gates in these schemes than in scheme A.

When the designs for the control elements of all the three schemes are avalaible, a better comparision of the schemes can be made. The area of the control elements and the interconnection lines can be taken into account when evaluating the array yield and the cell utilisation.

Two modifications for the interconnection schemes has been found to improve the yield. These are the extra bypass line and extra row modifications. Circuit designs for the implementation of these modifications are needed to be done. The effect of the extra circuitary on the yield also needs to be studied.

Most of the simulations done used random defect distribution as model for the real array of good and bad PE's. Eventhough other theoretical studies of WSI cellular array also used this model, a more realistic approach would be to use a model which takes defect clustering and radial defect variation into account. A model with defect clustering was tried in this study. The yield obtained was found to be lower than the yield obtained using random defect distribution. Eventhough there is no experimental support for the model, it does show that a lower yield is to be expected when a more realistic defect model is used.

Another area for future study is to find ways of implementing a global interconnection for the array. The global interconnection is to be used for the clock and the power supply to the cells. It could also be used to access each cell directly from outside the array for any instruction or data.

There was not much attention being given to the processing element (PE) in this thesis. The design for the processing element should be considered in future study. When this is done, the layout of the processing, control and testing elements can then be made. A prototype of a wafer scale reconfigurable orthogonal cellular array can then be developed.

## 4.3 CONCLUSION

Three designs has been presented for the intercellular connection of a reconfigurable two-dimensional orthogonal cellular array. For one of the schemes, a detailed study into the configuration of a functional array and into the design for its control element has been done. The yield simulations for the three schemes has been done by using computer simulations of the configuration of a functional array by the various schemes.

The three schemes been studied were labeled as schemes A, B and C. Scheme C was found to produce the best yield with scheme B being second best. For schemes B and C, the difference in yield became smaller with higher cell yield. When the effect of the more complex design in scheme C is considered, scheme B could be better than scheme C at high cell yield of about 80 % or more. The yield for scheme A was much lower as compared with the other schemes even at high cell yield.

The yield of the three schemes can be improved further by adding a few redundant rows of cells. The highest improvement could be obtained by adding just one redundant row. A scheme needs to be developed for the row bypass.

The length of the intercellular connection may be required to be restricted in order to reduce the delay on the interconnection lines. When this condition was imposed, the yields for the various schemes were reduced. The amount of yield reduction can be reduced by giving cells in the right-most few columns of the physical array direct access to the right edge of the array.

The results also shows the effectiveness of the strategy used in making the interconnection. With one-dimensional redundancy, a simple, orderly method of selection of the cells for the functional array is obtained. The selection of cells and the programming of the interconnection would be more difficult for two-dimensional redundancy eventhough there may be a larger selection of possible interconnections.

This thesis has shown the feasiblity of the three interconnection schemes for configuring a functional orthogonal array. From computer simulations, the yield for the three schemes can be expected to be higher than the other currently available interconnection schemes. More studies are needed to be done in order for a more precise comparision of the various interconnection schemes.

# REFERENCES

[Aubu78] R.C. Aubusson & I. Catt,
'Wafer scale integration - a fault tolerant procedure',
IEEE J. Solid State Circuits,
Vol. SC-13, No. 3 (June 1978) pp. 339-344.

[Aubu79] R.C. Aubusson,
'Wafer scale integration of semiconductor memory',
Ph.D. Thesis, Council for National Academic Award,
Middlesex Polytechnic (April 1979).

[Bars77] H. Barsuhn,
'Functional wafer - a new step in LSI',
European Solid State Circuit Conf., IEEE (Sept 1977) pp. 79-80.

[Berg85] A. Bergendahl et al,
'Thick film stripline transmission line interconnection for wafer scale integration',
Spring Meeting on VLSI, ECS (May 1985) Vol. 85-5, pp. 175-184.

[Bert83] W.J. Bertram,
'Yield and reliability',
VLSI Technology, S.M. Tze, Ed., McGraw-Hill (1983) pp. 599-637.

[Blod83] A.J. Blodgett, Jr.,
'Microelectronic packaging',
Scientific American, Vol. 249, No. 1 (July 1983) pp. 86-96.

[Bowl85] R. Bowlby,
'The DIP may take its final bows',
IEEE Spectrum, Vol. 22, No. 6 (June 1985) pp. 37-42.

[Burg82] N. Burgess,
'Yield enhancement of VLSI chip for an array computer',
M.Sc. Dissertation, Department of Electronics,
University of Southampton (1982).

[Calh72] D.F. Calhoun & L.P. McNamee,
'A means of reducing custom LSI Interconnection requirements',
IEEE J. Solid State Circuits,
Vol. SC-7, No. 5 (Oct 1972) pp. 395-404.

[Catt81] I. Catt,
'Wafer scale integration',
Wireless World, Vol. 87, No. 1546 (July 1981) pp. 57-59.

[Chap85] G.H. Chapman,
'Laser-linking technology for RVLSI',
Int. Workshop on Wafer-Scale Integration,
University of Southampton (July 1985).

[Donl85] B.J. Donlan et al,
'Wafer scale integration using discretionary microtransmission line interconnections',
Int. Workshop on Wafer-Scale Integration,
University of Southampton (July 1985).

[Egaw80] Y. Egawa et al,
'A 1 M-bit full wafer MOS RAM',
IEEE J. Solid State Circuits,
Vol. SC-15, No. 4 (Aug 1980) pp. 677-686.

[Elme77] B.R. Elmer et al,
'Fault-tolerant 92160 bit multiphase CCD memory',
Int. Solid State Conf., IEEE (Feb 1977) pp. 116-117.

[Evan85] R.A. Evans, J.V. McCanny & K.W. Wood,
'Wafer scale integration based on self-organisation',
Int. Workshop on Wafer Scale Integration,
University of Southampton (July 1985).

[Finn77] C.A. Finnila & H.H. Love,
'The associative linear array processor',
IEEE Trans. Computer, Vol. 26, No. 2 (Feb 1977) pp. 112-125.

[Frie84] M. Friedman et al,
'The R.P.I. wafer scale integration silicon compiler',
Int. Conf. Computer Design, IEEE (Oct 1984) pp. 107-114.

[Fuss82] D. Fussell & P. Varman,
'Fault-tolerant wafer-scale architecture for VLSI',
9th. Symp. Computer Architectures, IEEE (April 1982) pp. 190-198.

[Gave83] S.L. Gaverick,
'A single wafer 16-point 16-MHz FFT processer',
Custom Integrated Circuits Conf., IEEE (May 1983) pp. 244-248.

[Hedl82] K.S. Hedlund,
'Wafer scale integration of parallel processors',
Ph.D. Thesis, Purdue University (Aug 1982).

[Hedl84] K.S. Hedlund & L. Snyder,
'Systolic architecture - a wafer-scale approach',
Int. Conf. Computer Design, IEEE (Oct 1984) pp. 604-610.

[Hsia79] Y. Hsia, G.C.C. Chang & F.D. Erwin,
'Adaptive wafer scale integration',
11th. Conf. Solid State Devices, Jap. J. Applied Physics,
Vol. 19, Supp. 19-1 (1979) pp. 193-202.

[Hunt76] J.C. Hunter,
'Database retrival using superchip',
Symp. Advanced Memory Concepts, SRI (June 1976) pp. V450-V470.

[John84] R.R. Johnson,
'The significance of wafer scale integration in computer design',
Int. Conf. Computer Design, IEEE (Oct 1985) pp. 107-114.

[Kent83] P.F. Kent,
'Yield enhancement of integrated circuits by fault-tolerant design', B.Sc. Project Report, Department of Electronics, University of Southampton (April 1983).

[Kita80] K. Kitano et al,
'A 3 M-bit full wafer ROM',
IEEE J. Solid State Circuits,
Vol. SC-15, No. 4 (Aug 1980) pp. 686-693.

[Lanc83] A. Lancaster et al,
'Technique for improving engineering productivity of VLSI design', IBM J. Research and Development,
Vol. 25, No. 3 (May 1981) pp. 107-115.

[Lath67] J.W. Lathrop et al,
'A discretionary wiring system as the interface between design automation and semiconductor array manufacture',
Proc. IEEE, Vol. 55, No. 11 (Nov 1967) pp. 1977-1988.

[Line82] J.R. Lineback,
'Vertical short add row to memory',
Electronics, Vol. 55, No. 18 (8 Sept 1982) pp. 50-52.

[Logu81] J.C. Logue et al,
'Technique for improving engineering productivity of VLSI design', IBM J. Research and Development,
Vol. 25, No. 3 (May 1981) pp. 107-115.

[Mang84a] T.E. Mangir,
'Sources of failures and yield improvement for VLSI and restructurable interconnects for RVLSI and WSI :
Part I - Sources of failure and yield improvement for VLSI',
Proc. IEEE, Vol. 72, No. 6 (June 1984) pp. 690-708.

[Mang84b] T.E. Mangir,
'Interconnect Technology issues for testing and reconfiguration of WSI',
Int. Conf. Computer Design, IEEE (Oct 1984) pp. 127-131.

[Mang84c] T.E. Mangir,
'Sources of failures and yield improvement for VLSI and restructurable interconnects for RVLSI and WSI :
Part II - Restructurable interconnects for RVLSI and WSI',
Proc. IEEE, Vol. 72, No. 12 (Dec 1984) pp. 1687-1698.

[Mann75] F.B. Manning,
'Automatic test configuration and repair of cellular array',
Ph.D. Dissertation, Massachusetts Inst. of Technology (May 1975).

[Mann77] F.B. Manning,
'An approach to highly integrated, computer-maintained
cellular arrays', IEEE Trans. Computers,
Vol. C-26, No. 6 (June 1977) pp. 536-552.

[Mano82] T. Mano et al,
'A redundancy circuits for fault-tolerant 256 k MOS RAM',
IEEE J. Solid State Circuits,
Vol. 17, No. 4 (Aug 1982) pp. 726-731.

[McDo84] J.F. McDonald et al,
'The trials of wafer-scale integration',
IEEE Spectrum, Vol. 21, No. 10 (Oct 1984) pp. 32-39.

[Mina82] O. Minato et al,
'A hi-CMOS 8k X 8 bit static RAM',
IEEE J. Solid State Circuits,
Vol. 17, No. 5 (Oct 1982) pp. 793-797.

[Moor84] W.R. Moore,
'A review of fault-tolerant techniques for
enhancement of integrated circuit yield',
GEC J. Research, Vol. 2, No. 1 (April 1984) pp. 1-15.

[Moor85a] W.R. Moore & R. Mahat,
'Fault-tolerant communications for wafer scale integration
of a processor array', Microelectronics & Reliability,
Vol. 25, No. 2 (1985) pp. 291-294. (See appendix III)

[Moor85b] W.R. Moore,
'A critical review of fault-tolerant chips and WSI',
Int. Workshop on Wafer-Scale Integration,
University of Southampton (July 1985).

[Neug84] C.A. Neugebauer,
'Approaching wafer scale integration from the packaging point of
view', Int. Conf. Computer Design, IEEE (Oct 1984) pp. 115-120.

[Paz77] O. Paz & T.R. Lawson,
'Modification of Poisson statistics : Modelling defects
induced by diffusion', IEEE J. Solid State Circuits,
Vol. 12, No. 5 (Oct 1977) pp. 540-546.

[Pelt83] D.L. Peltzer,
'Wafer scale integration : The limit of VLSI?',
VLSI Design (Sept 1983) pp. 43-47.

[Perl81] D.S. Perloff et al,
'Microelectronics test chips in integrated circuit manufacturing',
Solid State Technology, Vol. 24, No. 9 (Sept 1981) pp. 75-80.

[Petr67] R.L. Petritz,
'Current Status of LSI Technology',
IEEE J. Solid State Circuits,
Vol. SC-2, No. 4 (Dec 1967) pp. 130-147.

[Posa81] J.G. Posa,
'What to do when the bits go out',
Electronics, Vol. 54, No. 15 (28 July 1981) pp. 117-120.

[Prad80] D.K. Pradhan & J.J. Stiffler,
'Error-correcting codes and self-checking circuits',
Computer, Vol. 13, No. 3 (March 1980) pp. 27-37.

[Raff83] J.I. Raffel et al,
'A demonstration of very large area integration using
laser restructuring', Int. Symp. Circuits and Systems,
IEEE (May 1983) pp. 781-784.

[Sami83] M. Sami & R. Steffanelli,
'Reconfigurable architectures for VLSI processing arrays',
National Computer Conf., AFIPS (May 1983) pp. 565-577.

[Schu76] S.E. Schuster & P.W. Cook,
'Laser personalisation of integrated circuits',
Proc. Soc. Photo-Optical Instru. Eng., Vol. 86 (1976) pp. 102-104.

[Shav83] D.C. Shaver et al,
'Electron-beam programmable 128K-bit wafer-scale EPROM',
IEEE Electron. Device Lett.,
Vol. EDL-4, No. 5 (May 1983) pp. 153-154.

[Siew82] D.P. Siewiorek & R.S. Swarz,
The Theory and Practice of Reliable System Design,
Digital (1982).

[Soma84] A.K. Somani & V.K. Agrawal,
'System level diagnostic in systolic system',
Int. Conf. Computer Design, IEEE (Oct1984) pp. 445-450.

[Spaw82] W. Spaw, A. Folmsbee & G. Canepa,
'A 128 k EPROM with redundancy',
Int. Solid State Circuits Conf., IEEE (Feb 1982) pp. 112-113, 304.

[Stap80] C.H. Stapper, A.N. McLaren & M. Dreckmann,
'Yield model for productivity optimization of VLSI memory
chips with redundancy and partially good products',
IBM J. Research and Development, Vol. 24 (May 1980) pp. 398-409.

[Stap83] C.H. Stapper, F.M. Armstrong & K. Saji,
'Integrated Circuit Yield Statistics',
Proc. IEEE, Vol. 71, No. 4 (April 1983) pp. 453-470.

[Stop85] H. Stopper,
'A Wafer with Electrically Programmable Interconnection',
IEEE Int. Solid-State Circuit Conf. (Feb 85) pp. 268-269.

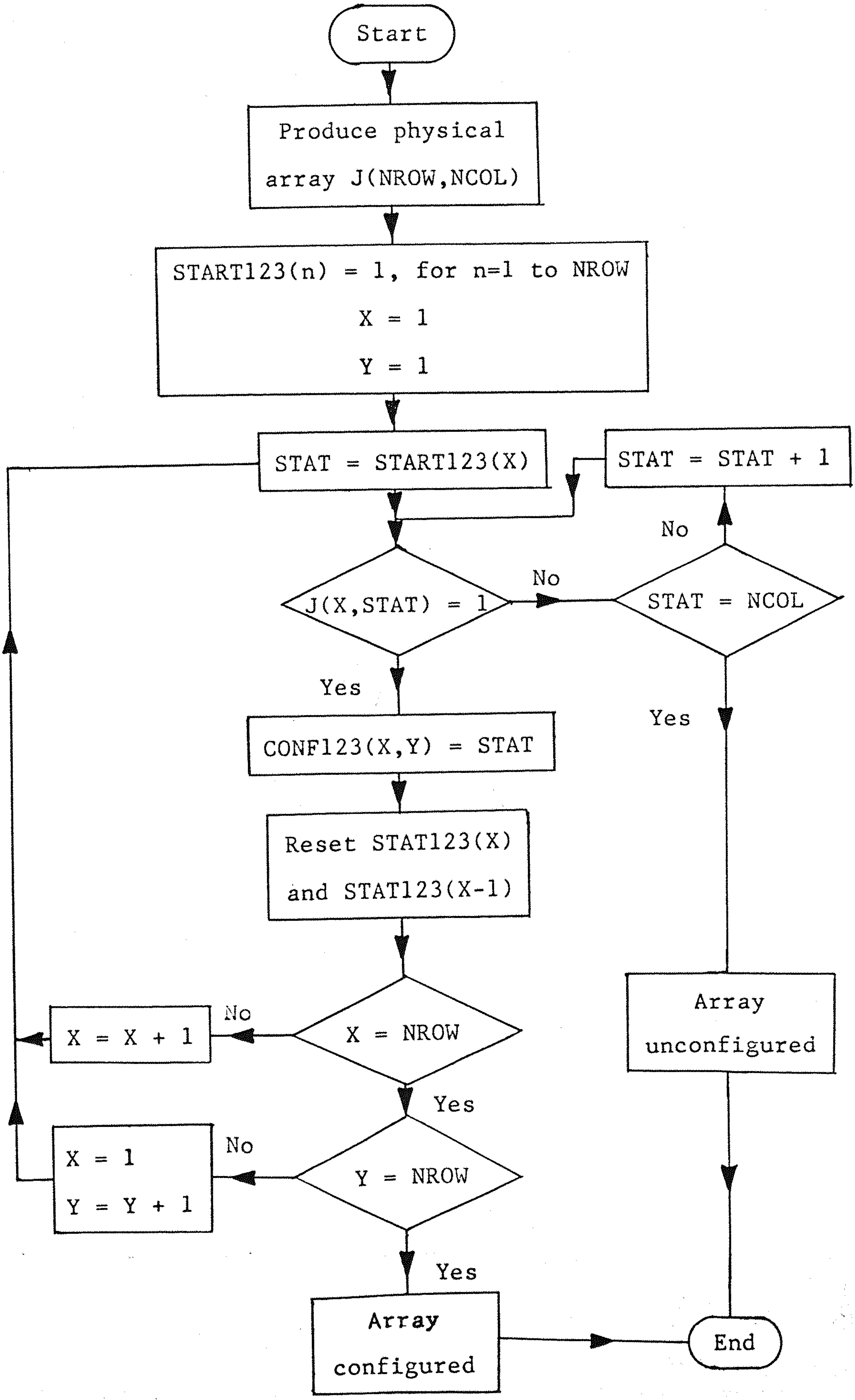
[Varm83] P.J. Varman,
'Wafer-scale Integration of linear processor arrays',
Ph.D. Dissertation, University of Texas at Austin (Aug 1983).

[Will82] T.W. Williams & K.P. Parker,
'Design for testability - a survey',
IEEE Trans. Computer, Vol. C-31, No. 1 (Jan 1982) pp. 2-15.

[Wu 82] W.W. Wu,
'Automated welding customises programable logic arrays',
Electronics, Vol. 55, No. 14 (14 July 1982) pp. 159-162.

[Yana72] T. Yanagawa,
'Yield degradation of integrated circuits due to spot defects',
IEEE Trans. Electron. Devices, Vol. 19 (1972) pp. 190-197.

Appendix I : Algorithm for the configuration of functional array using schemes A, B and C.



```
      PROGRAM ABC
C
C     This program configures (NROW x NROW) functional arrays
C     from a (NROW x NCOL) physical array with NYP cell yield,
C     using schemes A, B and C. The programming is done on
C     the University of Southampton ICL 2976.
C
      INTEGER J(16,35),CONF1(16,16),CONF2(16,16),CONF3(16,16),
     *STAT,STAT1(16)/16*1/,STAT2(16)/16*1/,STAT3(16)/16*1/,X,Y
      DATA NO,N1,MAX1,MAX2,MAX3/5*0/
      NCOL=35
      NROW=16
      NYP=80
C
C     This produces an array with NROW rows and NCOL columns,
C     and containing randomly distributed good and bad cells.
C     0 is for bad cell and 1 is for good cell. The probability
C     of each cell being good is NYP %. G05CBF and G05DAF are
C     pseudorandom number generators in library NAGF1. G05DAF(A,B)
C     produces a number between A and B.
C
      CALL G05CBF(1)
      DO 50 M=1,NROW,1
      DO 40 N=1,NCOL,1
      IR=INT(G05DAF(0.0,100.0))
      J(M,N)=0
      IF (IR.LE.NYP) J(M,N)=1
   40 N1=N1+J(M,N)
      NO=(NROW*NCOL)-N1
      PRINT 45,(J(M,K),K=1,NCOL,1)
   45 FORMAT(/40(I1,2X))
   50 CONTINUE
      PRINT 60,N1,NO
   60 FORMAT(//'NO. OF GOOD CELLS = ',I3,20X,'NO. OF BAD CELLS = ',I3)
C
C     This is the configuration using scheme A. CONF1(x,y) records
C     the column number of the location in the physical array of
C     the functional cell (x,y). STAT1(x) is the column number of
C     the waiting cell in row x.
C
      DO 130 Y=1,16,1
      DO 130 X=1,16,1
      DO 100 STAT=STAT1(X),NCOL,1
      IF (J(X,STAT).EQ.1) GOTO 110
  100 CONTINUE
      GOTO 160
  110 CONF1(X,Y)=STAT
      STAT1(X)=STAT+1
      IF (X.EQ.1) GOTO 130
      IF (STAT1(X).GT.STAT1(X-1)) STAT1(X-1)=STAT1(X)
      IF (STAT1(X).LE.CONF1((X-1),Y)) STAT1(X)=CONF1((X-1),Y)+1
  130 CONTINUE
      DO 150 X=1,16,1
      PRINT 140, (CONF1(X,Y),Y=1,16,1)
  140 FORMAT (/16(I2,3X))
      IF (CONF1(X,16).GT.MAX1) MAX1=CONF1(X,16)
  150 CONTINUE
```

```
 160  PRINT 170,MAX1
 170  FORMAT (/'RECONFIGURATION BY SCHEME A'/
     *'MAX. NO. OF COLUMNS = ',I2)
C
C     This is the configuration using scheme B. CONF2(x,y) records
C     the column number of the location in the physical array of
C     the functional cell (x,y). STAT2(x) is the column number of
C     the waiting cell in row x.
C
      DO 230 Y=1,16,1
      DO 230 X=1,16,1
      DO 200 STAT=STAT2(X),NCOL,1
      IF (J(X,STAT).EQ.1) GOTO 210
 200  CONTINUE
      GOTO 260
 210  CONF2(X,Y)=STAT
      STAT2(X)=STAT+1
      IF (X.EQ.1) GOTO 230
      IF (CONF2(X,Y).GT.STAT2(X-1)) STAT2(X-1)=CONF2(X,Y)
      IF (STAT2(X).LT.CONF2((X-1),Y)) STAT2(X)=CONF2((X-1),Y)
 230  CONTINUE
      DO 250 X=1,16,1
      PRINT 140,(CONF2(X,Y),Y=1,16,1)
      IF (CONF2(X,16).GT.MAX2) MAX2=CONF2(X,16)
 250  CONTINUE
 260  PRINT 270,MAX2
 270  FORMAT (/'RECONFIGURATION BY SCHEME B'/
     *'MAX. NO. OF COLUMNS = ',I2)
C
C     This is the configuration using scheme C. CONF3(x,y) records
C     the column number of the location in the physical array of
C     the functional cell (x,y). STAT3(x) is the column number of
C     the waiting cell in row x.
C
      DO 330 Y=1,16,1
      DO 330 X=1,16,1
      DO 300 STAT=STAT3(X),NCOL,1
      IF (J(X,STAT).EQ.1) GOTO 310
 300  CONTINUE
      GOTO 360
 310  CONF3(X,Y)=STAT
      STAT3(X)=STAT+1
      IF ((Y.EQ.1).OR.(X.EQ.1)) GOTO 330
      IF (STAT3(X-1).LT.CONF3(X,(Y-1))) STAT3(X-1)=CONF3(X,(Y-1))
      IF (STAT3(X).LT.CONF3((X-1),(Y-1))) STAT3(X)=CONF3((X-1),(Y-1))
 330  CONTINUE
      DO 350 X=1,16,1
      PRINT 140,(CONF3(X,Y),Y=1,16,1)
      IF (CONF3(X,16).GT.MAX3) MAX3=CONF3(X,16)
 350  CONTINUE
 360  PRINT 370,MAX3
 370  FORMAT (/'RECONFIGURATION BY SCHEME C'/
     *'MAX. NO. OF COLUMNS = ',I2)
      STOP
      END
```

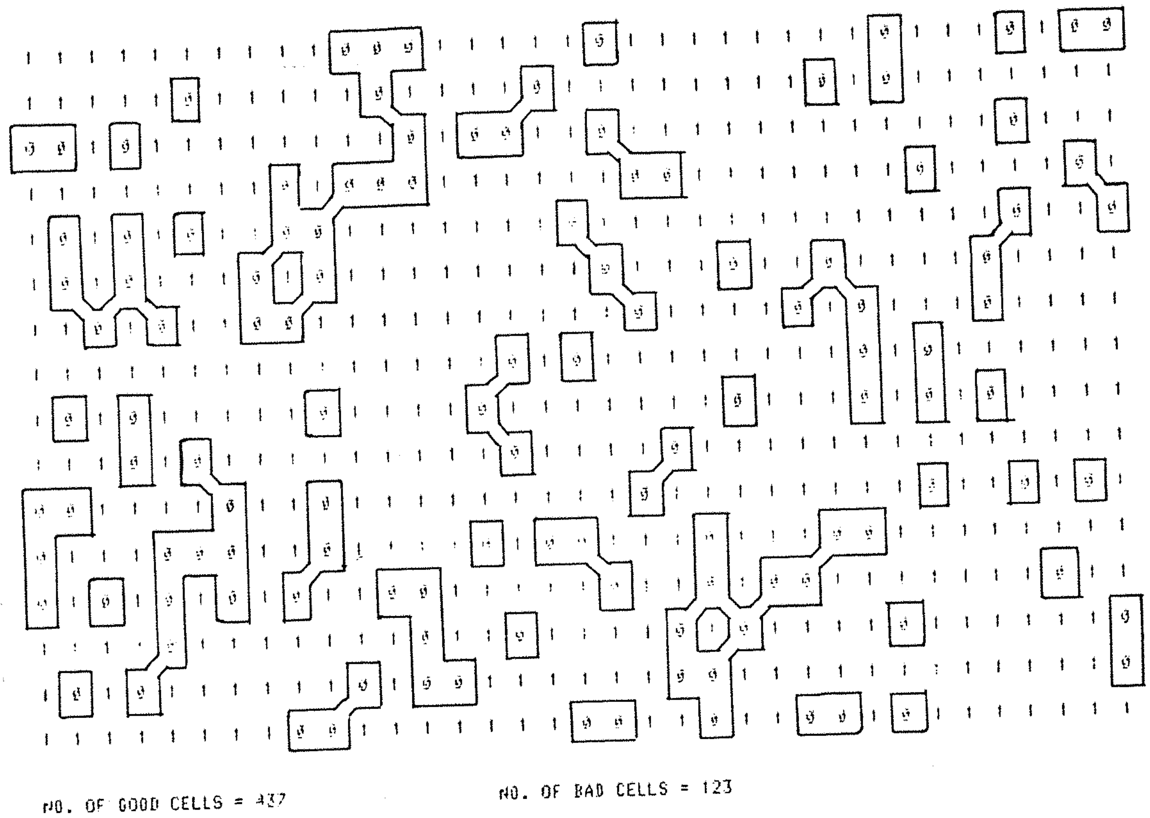


Figure I.1 Physical array with random defect distribution formed with 80 % cell yield.

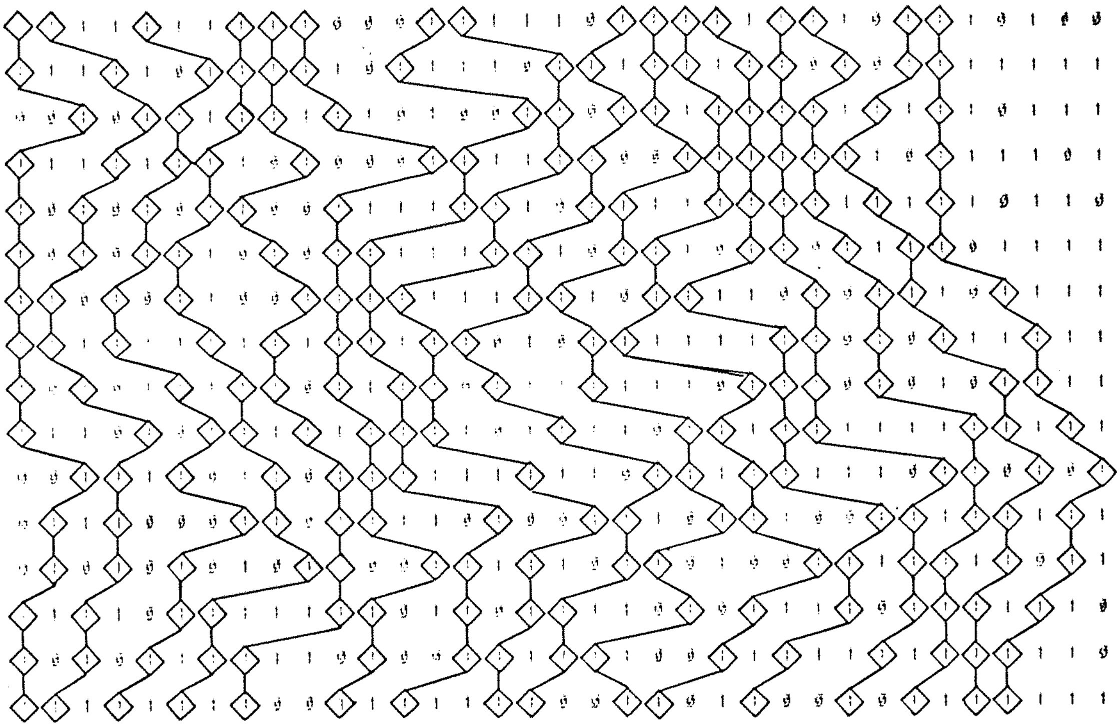


Figure I.2 Columns for 16X16 functional array configured using scheme A.

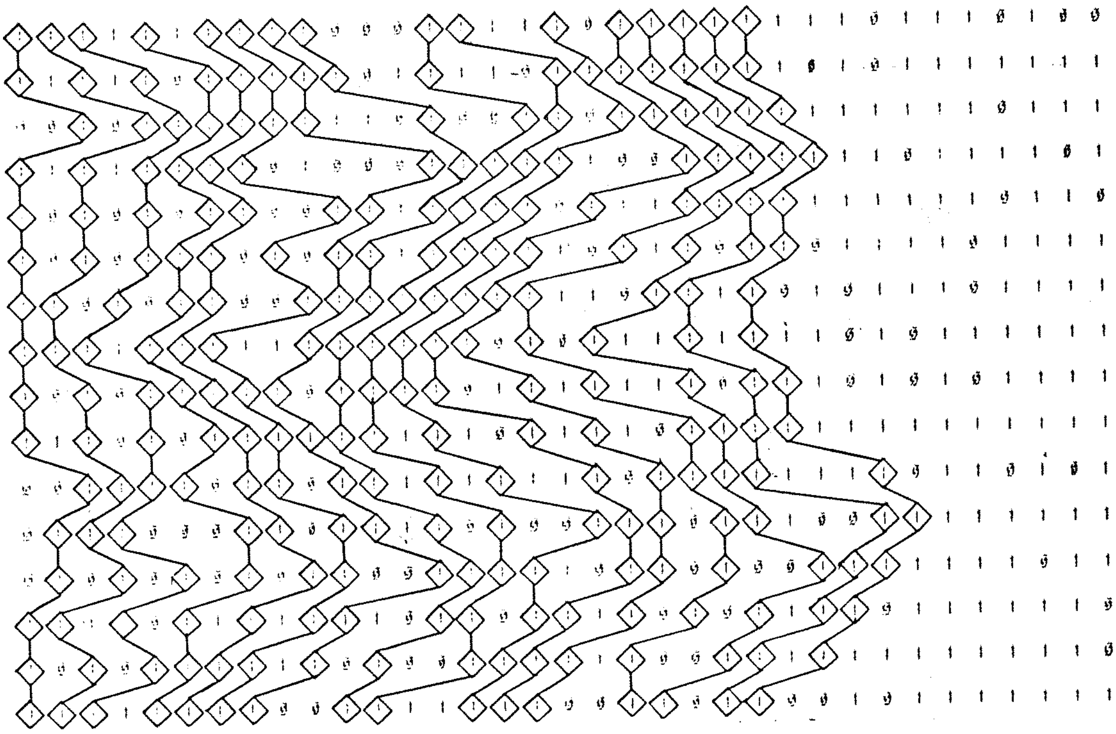


Figure I.3 Columns for 16X16 functional array configured using scheme B.

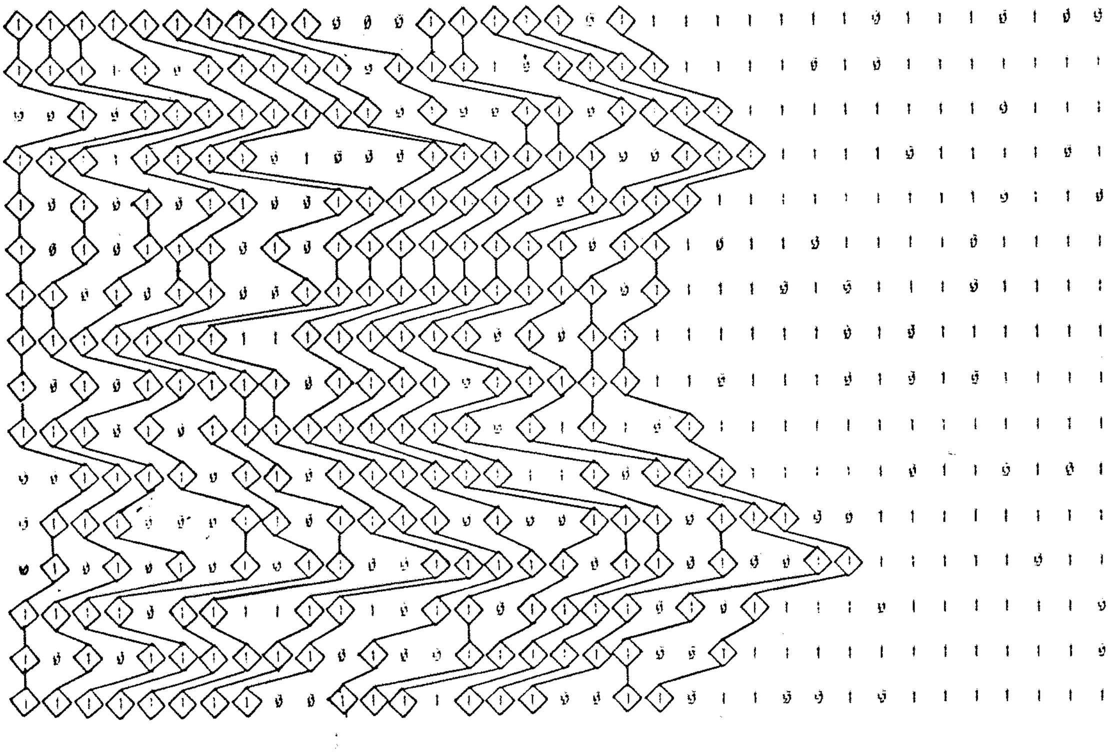


Figure I.4 Columns for 16X16 functional array configured using scheme C.

# Appendix II : Algorithm for the formation of cellular array with Y % cell yield and with defect clustering.

Required probability of good cell : Y/100
Required probability of bad cell : (100-Y)/100

First, an array with half the required defect probability is formed.
Probability of bad cell : B : (100-Y)/200
Probability of good cell : G : (100+Y)/200

Next, the defect probability is increased to (100-Y)/100 by changing some of the good cells to bad according to the number of bad neighbours.
Total probability of changing good cells to bad : B
Number of observed neighbours of each good cell : 8
Probability of a good cell having n bad neighbours and (8-n) good neighbours : $\frac{8!}{n!(8-n)!} B^n G^{8-n} G$

Probability of a good cell with n bad neighbours being change to bad : (n+1)X

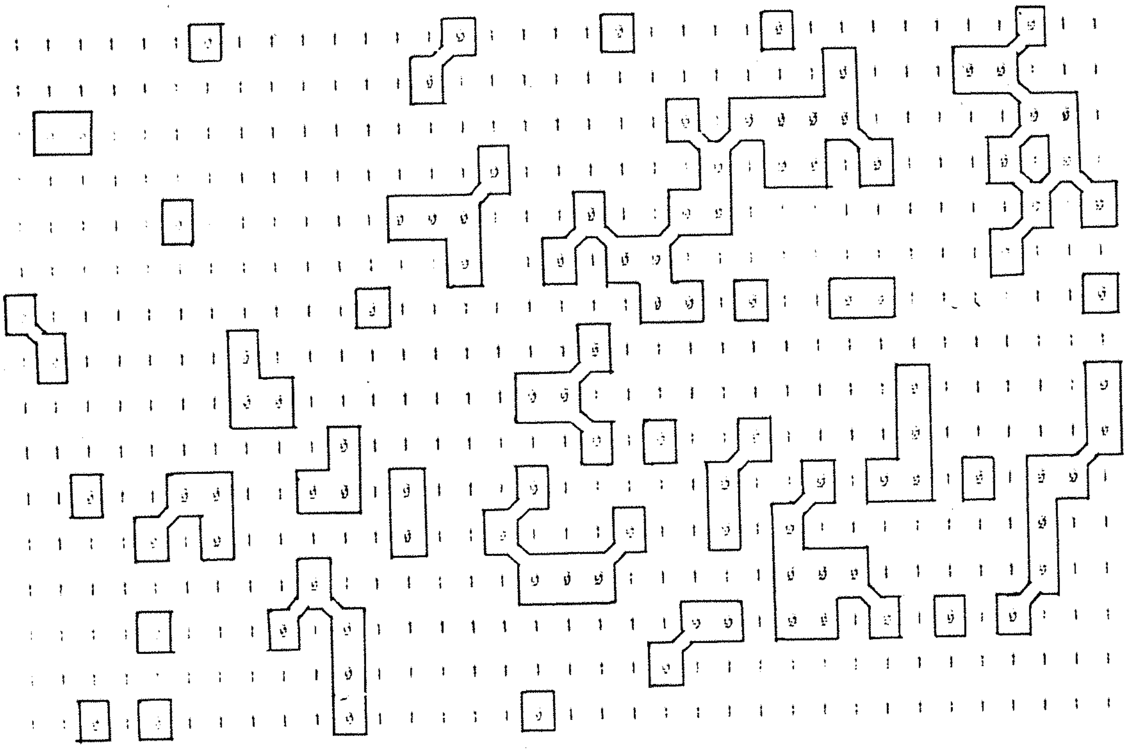
$\therefore$ Total probability of any good cell being change to bad :

$$\sum_{n=0}^{8} (n+1)X \frac{8!}{n!(8-n)!} B^n G^{9-n} = B$$

$$\sum_{n=1}^{9} n \frac{8!}{(n-1)!(9-n)!} (100-Y)^{n-2} (100+Y)^{10-n} = \frac{200^8}{X}$$

From the above equation, the value of X can then be found. For a good cell with n bad neighbours, it is changed to bad if a number generated for it is less than (n+1)X. Otherwise the cell will remain good.

```
      PROGRAM CLUSTER
C
C
C     This program produces an array with NROW rows and NCOL columns,
C     and containing good and bad cells. Some of the bad cells are
C     made to cluster together. The overall probability for a good
C     cell is CY %. G05CBF and G05DAF are pseudorandom number generators
C     in file NAGF1. G05DAF(A,B) produces a number between A and B.
C     This program is run on the University of Southampton ICL 2976.
C
C
      INTEGER J(16,35),JK(18,37),X,Y
      N0=0
      N1=0
      CALL G05CBF(0)
      NROW=16
      NCOL=35
      CY=80.0
      DO 20 X=1,(NROW+2),1
      DO 20 Y=1,(NCOL+2),1
      A=G05DAF(0.0,1.0)
      JK(X,Y)=0
      IF (A.LE.((100+CY)/200)) JK(X,Y)=1
   20 CONTINUE
      SUM=0.0
      FAC=1.0
      DO 10 I=1,9,1
      SUM=SUM+I*FAC*((100-CY)**(I-2))*((100+CY)**(10-I))
   10 FAC=FAC*(9-I)/I
      XP=2.56E18/SUM
      DO 50 X=1,NROW,1
      DO 30 Y=1,NCOL,1
      IF (JK((X+1),(Y+1)).NE.0) GOTO 23
      J(X,Y)=0
      GOTO 28
   23 NBAD=9
      DO 25 IX=X,(X+2),1
      DO 25 IY=Y,(Y+2),1
      NBAD=NBAD-JK(IX,IY)
   25 CONTINUE
      B=G05DAF(0.0,1.0)
      J(X,Y)=1
      IF (B.LT.((NBAD+1)*XP)) J(X,Y)=0
      IF (J(X,Y).EQ.1) N1=N1+1
   28 IF (J(X,Y).EQ.0) N0=N0+1
   30 CONTINUE
      PRINT 45,(J(X,K),K=1,NCOL)
   45 FORMAT(/35(I1,2X))
   50 CONTINUE
      PRINT 60,N1,N0
   60 FORMAT(//'NO. OF GOOD CELLS = ',I3,20X,'NO. OF BAD CELLS = ',I3)
      STOP
      END
```



```
NO. OF GOOD CELLS = 452                    NO. OF BAD CELLS = 108
```

Figure II.1 Physical array with defect clustering formed with 80 % cell yield.

*Microelectron. Reliab.*, Vol. 25, No. 2, pp. 291–294, 1985.
Printed in Great Britain.

0026-2714/85$3.00 + .00
© 1985 Pergamon Press Ltd.

# FAULT-TOLERANT COMMUNICATIONS FOR WAFER-SCALE INTEGRATION OF A PROCESSOR ARRAY

W. R. Moore and R. Mahat
Department of Electronics and Information Engineering, University of Southampton,
Southampton, SO9 5NH, U.K.

(*Received for publication* 4 *September* 1984)

**Abstract**—This paper describes schemes for introducing fault-tolerance into a two-dimensional orthogonal array of cells with nearest neighbour communication paths. The schemes are designed to tolerate a large number of faults and are therefore applicable to the yield-enhancement of large-area VLSI circuits. Simulation results are presented which show the superiority of the schemes over previous proposals and indicate that the nearest neighbour interconnections need not be a barrier to the desirable goal of integrating an array computer onto a whole-wafer circuit.

## INTRODUCTION

An increasingly popular architecture for high speed image processing and other computationally intensive problems is a rectangular array of single-bit processors with four nearest neighbour interconnections as illustrated in Fig. 1 [1–6]. Simple gating within the processors can couple pairs of interconnections together to give diagonal routing so that this network permits all processors simultaneously to access data from any one of their eight nearest neighbours [4]. At the present state of the art, VLSI circuits can be fabricated with around 32 or 64 processors per chip so that a typical application with perhaps a 16 × 16 or a 64 × 64 processor array would require the interconnection of between 4 and 128 identical chips. In addition each processor has its own memory and this may cause as much as a five-fold increase in the silicon area used.

The production of larger chips would have advantages in terms of higher speed and lower power consumption as well as requiring a physically smaller volume. In the limit it may be feasible to build a complete array computer on a whole-wafer circuit. The usual limit to chip size is the increasingly poor yield but this limit can be raised by the use of fault-tolerant design techniques.

Although it is possible to probe test every cell (one processor plus memory) and wire together the working ones this is a very expensive procedure even with the recent progress in laser fusing and welding [7]. A more likely route to economic wafer-scale integration would seem to be a logically controlled configuration, but simple approaches such as bypassing a faulty row or column [8] are not applicable to the relatively large number of defects expected. The problem is far from trivial and the first published study of it produced rather discouraging results [9]: Manning's approach requires that the yield of each cell exceed 97.5% in order that the average wafer with 25 × 25 cells can be configured into a 16 × 16 working array (a 144% overhead of redundant cells). The inefficiency of this approach reflects the use of a high proportion of cells solely as switches.

More recently [10] Hedlund examined the problem of mapping the CHiP reconfigurable computer onto a whole-wafer. The CHiP computer is essentially a two-dimensional grid but the inclusion of extensive switching between cells (Fig. 2a) allows it to be configured in a variety of architectures, Hedlund supplements these switches (Fig. 2b) to permit the formation of 2 × 2 sub-arrays of good cells in 3 × 4-cell blocks of the grid and places the blocks in a hierarchical framework. Hedlund predicts that a 65% cell yield is sufficient to allow a wafer of 9 × 9 blocks to be configured as a 16 × 16 working array with a 96.5% yield, but this works out at a staggering 280% overhead of redundant cells.

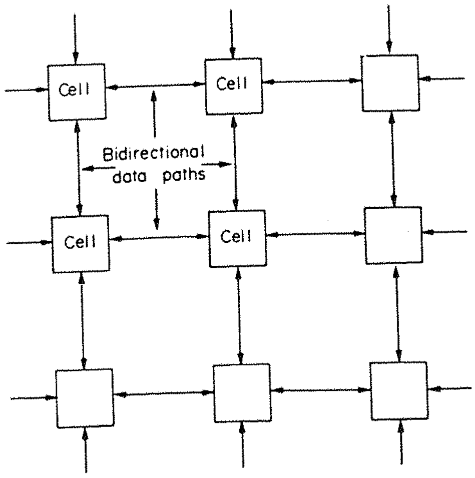


Fig. 1. An array of cells with four nearest neighbour interconnections.

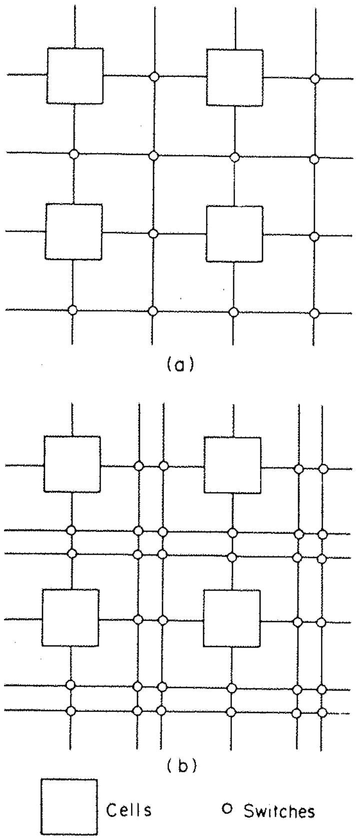


Fig. 2. Mapping the CHiP computer onto a defective wafer. (a) The CHiP array. (b) Supplemented grid.

This paper examines the problem of placing switches specifically to configure cells into a rectangular array of working cells and produces much more encouraging results. For example, with Scheme B, Fig. 11 shows that for a 95 % cell yield an overhead of just 15 % redundant cells is sufficient to permit the average wafer to be configured as a 16 × 16 array and Fig. 9 shows that a 65 % cell yield can lead to a 96.5 % wafer yield with an overhead of just 170 % redundant cells.

## CONFIGURATION SCHEMES

The configuration schemes investigated were chosen for their effectiveness and for their ease of use. The basic concept behind them is very simple: *firstly* a rectangular array of cells is laid out; *then* row configuration switches are placed to permit defective cells to be bypassed and the remaining cells in each row to be chained together; *finally* column configuration switches are placed between adjacent columns to permit the column communication paths to be shifted to take in a working cell from every row.

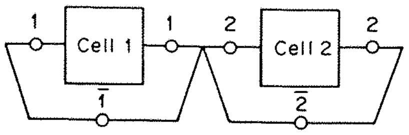


Fig. 3. Row configuration scheme.

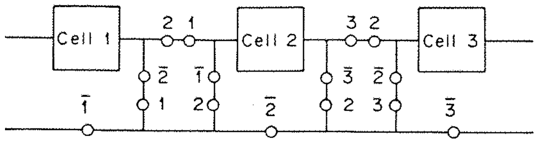


Fig. 4. Alternative row configuration scheme to avoid single point failure mode.

The basic row configuration circuit is shown in Fig. 3. Each cell has three gates, a working cell is connected into this chain by opening the two gates on either side and a defective cell is bypassed by opening the third gate in the bypass line instead. In this way adjacent cells can be connected via just two gates and each defective cell that is bypassed adds just one further gate to the communication path. Because the circuit is vulnerable to a single short circuit defect at the conjunction of every three gates, an alternative circuit has also been considered as shown in Fig. 4. This performs the same function as the simple circuit of Fig. 3 but avoids the vulnerability to single point defects. Predictably the switch area increases and hence also does the probability of *some* switching fault occurring. Nevertheless the switches are still easy to control and a maximum of two additional gates is added to any communication path.

Connecting together the rows to form a fully connected array is a slightly more difficult problem and we have considered three alternatives. The simplest circuit is shown as Scheme A in Fig. 5. Two working cells in the same column are connected by

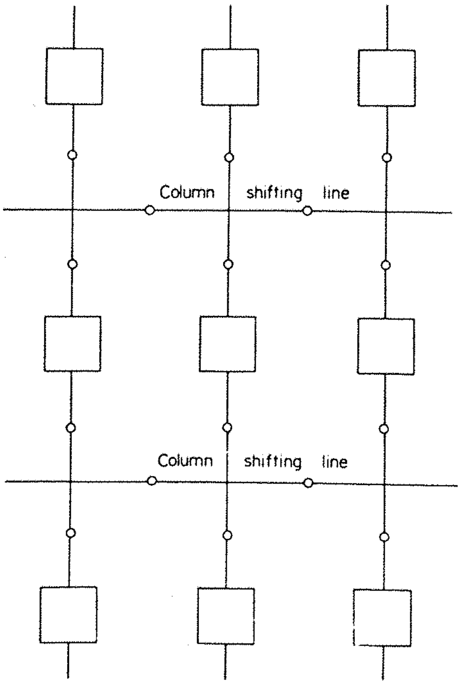


Fig. 5. Column configuration scheme A.

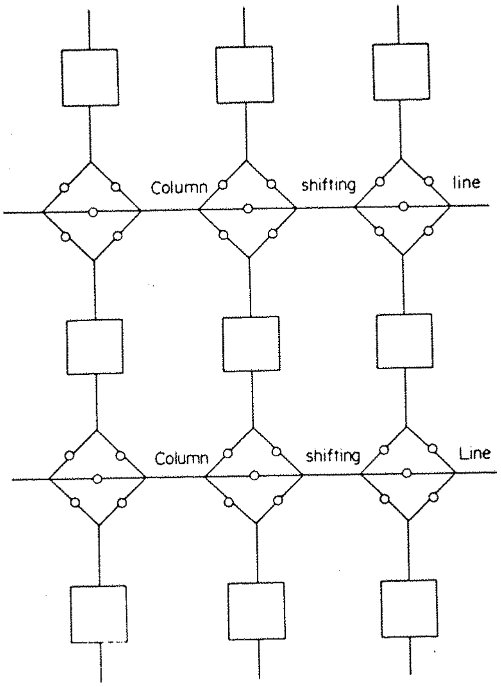


Fig. 6. Column configuration scheme B.

opening the two gates between them and closing the two adjacent gates on the column shifting line. The column can be shifted left or right by opening one of the gates on the column switching line and closing the gates to the cells to be bypassed. One disadvantage of this simple scheme is that whenever the column shifting line is used, two cells are discarded, one of which may be fault-free.

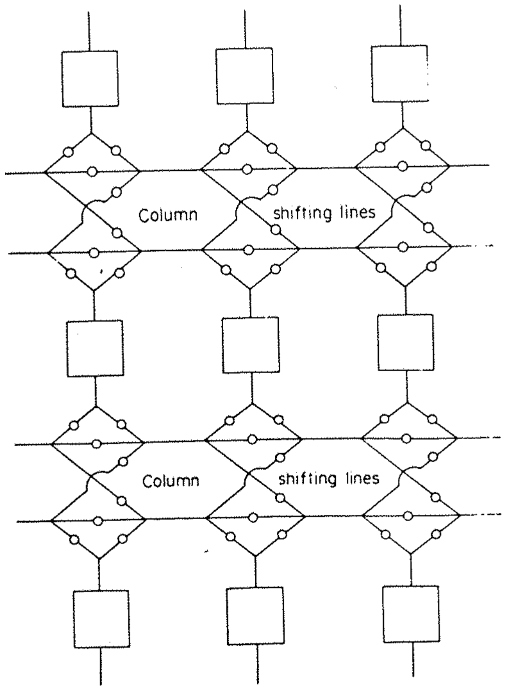


Fig. 7. Column configuration scheme C.

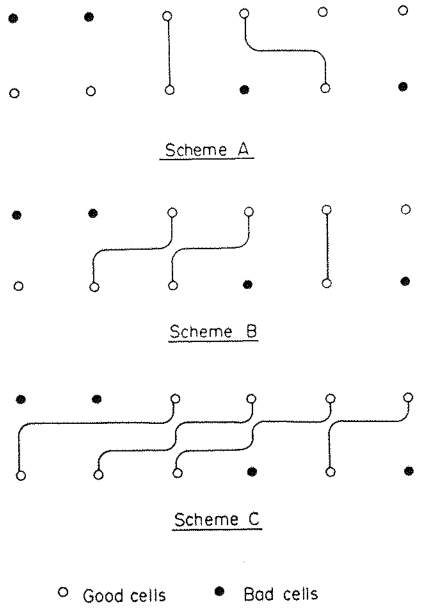


Fig. 8. Configuration of the same defective array using schemes A, B and C.

An alternative circuit to overcome this problem of losing good cells is shown as Scheme B in Fig. 6. With this circuit, it is still possible to use a cell even when the cell above it is connected to another column. To connect two cells in the same column, the two gates on either the right or the left are opened. To shift this column, only one of these gates is opened depending on the direction of shift. To perform a double column shift the center gate is opened and all the other four gates closed.

A further refinement is shown as Scheme C in Fig. 7. Here there are two lines for column shifting so that even double shifts can be made without the necessity of discarding cells.

Figure 8 shows how the three schemes configure the same defective array. For this particular array, Scheme A can utilize only two columns, but Scheme B three columns and Scheme C four columns.

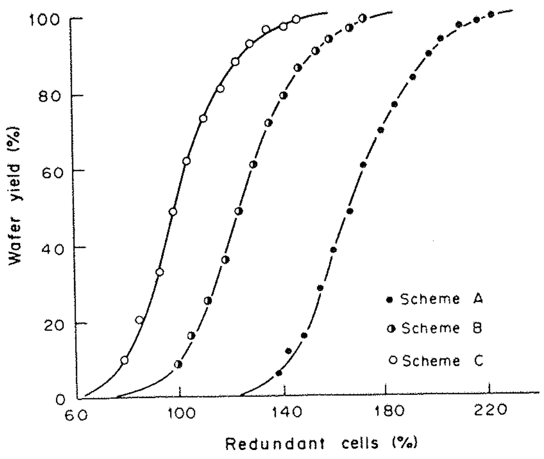


Fig. 9. Yield of wafer with individual cell yield of 65 % using the three column configuration schemes.

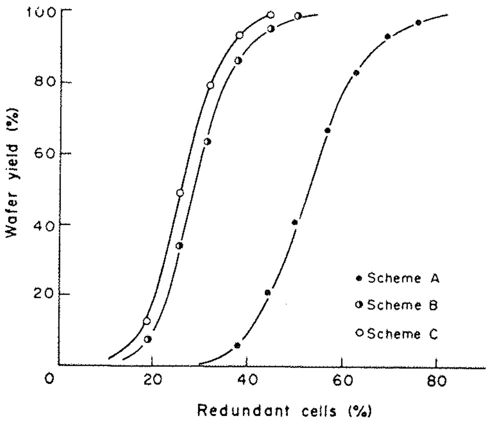


Fig. 10. Yield of wafer with individual cell yield of 90 % using the three column configuration schemes.

## SIMULATION RESULTS

These three schemes have been tested to evaluate their effectiveness in forming an array of 16 x 16 gooa cells. For simplicity a uniform random distribution of defects has been assumed. Monte Carlo simulation with 500 trials at cell yields of 65 %, 90 % and 95 % are given in Figs 9, 10 and 11 which illustrate the expected yield as a function of the overhead of redundant cells. All three schemes perform substantially better than previous approaches. On balance Scheme B appears to represent the best value performing significantly better than Scheme A in all cases. Scheme C offers a much lower return for its increase in complexity but begins to look more attractive at very low cell yields.

Typically, Scheme B produces about 95 % wafer yield with an overhead of just 43 % when the cell yield is 90 % or with an overhead of just 165 % when the cell yield is 65 %.

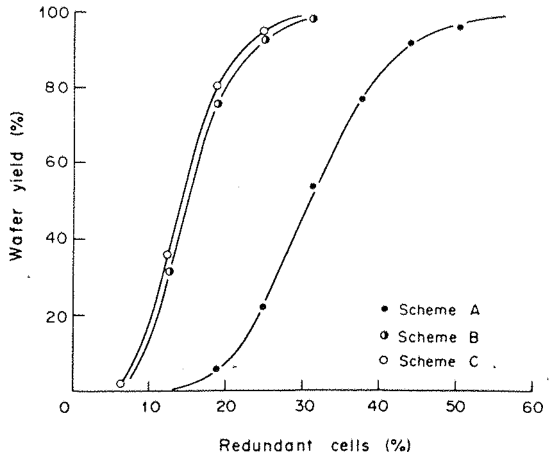


Fig. 11. Yield of wafer with individual cell yield of 95 % using the three column configuration schemes.

## DISCUSSION

We have presented results which we believe demonstrate that it is feasible to design fault-tolerance into a rectangular array of processors to permit larger than average VLSI chips to be produced economically. A number of important practical aspects have been ignored however and are the subject of our current studies.

The effects of defects in the configuration circuitry have not been thoroughly investigated. For a typical cell containing a single-bit processor and memory we expect the configuration circuitry (Scheme B) to occupy no more than 4 % of the cell area. It is then certainly necessary to avoid all single-point failure modes as discussed above (Fig. 4). The approach is still vulnerable to certain combinations of faults on the row switching circuit however, and to defects affecting large areas of a wafer. We do not believe that these represent a significant problem but possible improvements would be the use of relaxed design rules for the configuration circuits or else the use of a hierarchical configuration scheme.

We have assumed that the switches will be controlled by local latches and are currently investigating the best way of testing the wafer and of distributing the configuration control.

Any real system will have other interconnections between cells, at least power, ground and clock and probably some control lines too. These are essentially easier to deal with because they need not be specific to a particular cell, but are nevertheless not trivial problems in a whole-wafer design.

*Acknowledgements*—We are grateful to Neil Burgess and Phil Kent for their initial studies of this problem.

## REFERENCES

1. S. F. Reddaway, *Infotech State of the Art Report on Super Computers*, Vol. 2 (Edited by C. R. Jesshope and R. W. Hockney), p. 309 (1979).
2. M. J. B. Duff, *Electronics and Power* 888 (November 1980).
3. K. E. Batcher, *IEEE Trans. Comput.* **29**, 839 (1980).
4. I. N. Robinson and W. R. Moore, *IEEE Custom Integrated Circuits Conference*, Rochester, New York, p. 41 (1982).
5. C. Weems, S. Levitan and C. Foster, "Titanic, a VLSI based content addressable parallel array processor", *ICCC* 236–239 (September 1982).
6. T. Kondo, T. Nakashima, M. Aoki and T. Sudo, *IEEE J. Solid St. Circuits* **18**, 2, 147 (1983).
7. D. Russell and P. Varman, *9th Symposium on Computer Architectures*, p. 190. IEEE (1982).
8. W. R. Moore, *Proc. IEE*, Pt. E **129**, 229 (1982).
9. F. B. Manning, *IEEE Trans. Comput* **26**, 6, 537 (1977).
10. K. S. Hedlund and L. Snyder, *International Conference on Parallel Processing*, p. 262. IEEE (1982).