EXPLORING HIGH RESOLUTION TEST PATTERN TO IMPROVE THE CACHE FAILURE ANALYSIS

By

ONG CHEIN EE

A Dissertation submitted for partial fulfilment of the requirement for the degree of Master of Science (Microelectronic Engineering)

August 2017

Acknowledgement

First, I would like to take an opportunity to thank my dissertation project supervisor, Dr. Mohd Tafir bin Mustaffa, for giving valuable advice, suggestion and feedback on my dissertation project. His kindly guidance really help in adding more values in this dissertation project and make this project completed on time.

Moreover, this dissertation project would not have been possible without the support from many people, especially my family, friends and my working teammate. I would like to appreciate my manager Mrs. Joyce Ho Suan Hong for giving the opportunity to further study in USM and her guidance in confidential information matter when preparing this dissertation. At the same time, I would like to thank our technical expert Mr. Ahmad Afif bin Azmi and Mr. Khor Chuan Chong support and giving the technical advice throughout the whole dissertation project.

Finally, I would like to thank my beloved family, especially my parents Ong Bok Seng and Chong Meow Sean which always encourage and motivate me from the starting point toward the ending point of my master study in USM.

Table of Content

List of Tables

List of Figures

List of Abbreviations

Abbreviation	**Meaning**
ATE	Automated Test Equipment
BIST	Build in Self-Test
BL	Bit-Line
BSE	Back-Scattered Electrons
CTV	Capture Test Vector
DE	Design Engineer
DFT	Design for Test
DLS	Dynamic Laser Simulation
DRAM	Dynamic Random Access Memory
EFA	Electrical Failure Analysis
EOS	Electrical Overstress
ESD	Electro Static Discharge
FAE	Failure Analysis Engineer
FAFI	Failure Analysis & Failure Isolation
FIB	Focused Ion Bean
I/O	Input & Output
IC	Integrated Circuit
IP	Intellectual Property
IV	Current-Voltage
JTAG	Joint Test Action Group
MBIST	Memory Build in Self-Test
MOVI	Moving Inversion
PBIST	Programmable Build in Self-Test
PDE	Product Development Engineer
PEM	Photon Emission Microscope
PFA	Physical Failure Analysis
QE	Quality Ensure
QRE	Quality & Reliability Engineer
RF	Register File
ROI	Region of Interest
SEM	Scanning Electron Microscope
SOF	Stop on Fail
SRAM	Static Random Access Memory
TAP	Test Access Point
TDO	Test Data Output
TEM	Transmission Electron Microscopy
WL	Word-Line
WLE	Word-Line Enable

PENEROKAAN CORAK UJIAN RESOLUSI TINGGI DALAM PENAMBAHBAIKAN ANALISIS KEGAGALAN SUSUNATUR

Abstrak

Biasanya, hanya asas lulus/gagal algoritma ujian digunakan untuk membuat ujian terhadap susunatur dalam pemproses. Tetapi, asas lulus/gagal algoritma ujian tidak dapat mengenal pasti lokasi kegagalan susunatur apabila digunakan dalam analisis kegagalan (AK) untuk mencari punca kegagalan. Resolusi corak algoritma ujian memainkan peranan yang penting dalam AK supaya dapat mengenal pasti setiap bit yang gagal dalam susunatur. Tesis ini membincangkan konsep untuk membuat corak ujian resolusi tinggi melalui *Memory Build in Self-Test* (MBIST). Cara penggunaan MBIST ialah dengan memasukkan elemen *Capture Test Vector* (CTV) dalam corak ujian untuk menambahbaik resolusi corak ujian. Pada masa yang sama, kepentingan corak ujian resolusi tinggi telah ditunjukkan dalam kes kajian yang sebenar. Corak ujian tersebut telah digunakan di dalam *Automated Test Equipment* (ATE) untuk melakukan ujian terhadap pemproses. Satu pemproses telah menjalani proses *Focused Ion Beam* (FIB) bagi memusnahkan memori bit dalam lokasi susunatur untuk membuktikan corak ujian tersebut dapat berfungsi dengan betul. Akhir sekali, corak ujian resolusi tinggi tersebut digunakan dalam kes sebenar sebagai bukti corak ujian tersebut mempunyai kebolehan untuk meningkatkan kecekapan AK. Teknik AK dan aplikasi corak ujian resolusi tinggi dalam proses susunatur memperbaiki telah dibentangkan dari peringkat pengujian sehingga perigkat pemusnahan AK. Keputusan kes ini telah membuktikan konsep yang dicadangkan untuk membolehkan corak ujian resolusi tinggi adalah berkesan dan dapat mempertingkatkan kecekapan AK dengan secara tidak langsung mempertingkatkan kadar kejayaan untuk menemui punca kegagalan.

EXPLORING HIGH RESOLUTION TEST PATTERN TO IMPROVE THE CACHE FAILURE ANALYSIS

Abstract

Typically, only pass/fail basis test algorithm is being used to test the cache array in silicon devices. But the pass/fail basis test algorithm is insufficient to identify the failing characteristic of the cache array when it comes to the failure analysis (FA) and debug stage to find out the root cause of the failing mechanism. The resolution of test algorithm plays an important role in helping FA process to identify every single failing bits in cache array. In this dissertation, the concept of bringing up the Memory Build in Self-Test (MBIST) high resolution test pattern is discussed. The utilization of MBIST engine by insertion of the Capture Test Vector (CTV) element into the test algorithm is the main concept in increasing the resolution test pattern. At the same time, the importance of high resolution test pattern in FA process is being shown in a real case study. The generated high resolution test pattern is integrated for Automated Test Equipment (ATE) usage so that the test pattern can be applied in real silicon device testing. Then, a silicon device is edited using Focused Ion Beam (FIB) to destroy the memory bits in cache array for proving the test pattern is functioning properly. Finally, the high resolution test pattern is being used in real case application for proving the high resolution test pattern have the capability in improving FA efficiency in identifying the failing bits. The FA technique and application of using high resolution test pattern in debugging the cache failure are shown from the testing stage until the destructive FA stage. The finding in real case FA proved the concept of bring up the MBIST high resolution test pattern is working properly. It is able to increase the effectiveness of failure analysis and failure isolation process which indirectly increase the success rate for finding the root cause.

CHAPTER 1

INTRODUCTION

1.1 Background

In industry standard, good silicon has to pass through several quality and reliability testing after burn-in and stress test. Good silicon will fulfill the minimum requirement of quality and reliability in industry standard before the silicon can be manufactured in high volume and released in the market. The Design for Test (DFT) circuitry is constructed in processor for testability propose. DFT circuitry is able to identify the silicon health condition after go through the burn-in [1] and stress test. DFT circuitry is an access point for the Test Engineer to test and debug the silicon failure. The DFT concept created the observability and controllability of the internal circuit by controlling the external input and output (I/O) pins of the silicon devices.

The DFT test can be categorized into scan test, cache test and parametric test. Scan test is targeted on the logic failure of the transistor. Parametric test is targeted on the I/O failure such as Electro Static Discharge (ESD), Electrical Overstress (EOS), I/O leakage or open. Cache test is specially target on the cache memory failure while the Build in Self-Test (BIST) [2] is the DFT architecture for cache region. This dissertation project is targeted in failure analysis process which focuses on cache test methodology; failure analysis and failure isolation (FAFI) on cache region by enabling the high resolution test pattern capability on automated test equipment (ATE) defined as tester.

For having a better understanding on Failure Analysis Engineer (FAE) debug and perform FAFI on cache reject unit from production floor, quality and reliability department or customer. The General Flow Diagram of Cache Failure Analysis Process is introduced in Figure 1.1.

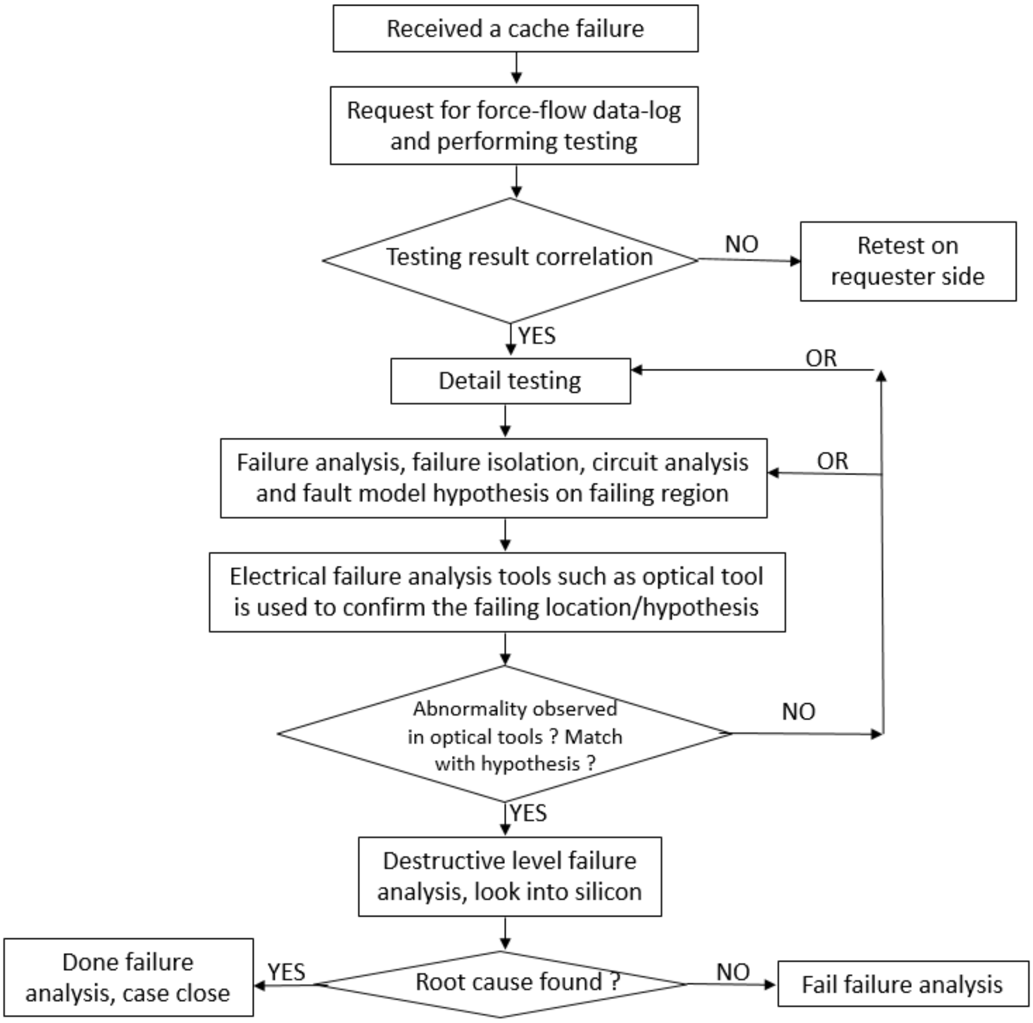

Received a cache failure
Request for force-flow data-log and performing testing
Testing result correlation
NO
Retest on requester side
YES
Detail testing
OR
Failure analysis, failure isolation, circuit analysis and fault model hypothesis on failing region
OR
Electrical failure analysis tools such as optical tool is used to confirm the failing location/hypothesis
Abnormality observed in optical tools ? Match with hypothesis ?
NO
YES
Destructive level failure analysis, look into silicon
Done failure analysis, case close
YES
Root cause found ?
NO
Fail failure analysis

Figure 1.1: General Flow Diagram of Cache Failure Analysis Process

A silicon device cache portion includes Static Random Access Memory (SRAM), Dynamic Random Access Memory (DRAM) and File Register (RF). The testing operation to identify the health condition on the cache array can be illustrated in Figure 1.2. For a good/healthy condition cache array, the data written into cache must be the same when the data is read out from the cache array. But for a bad cache array, the data write in and read out will be difference. As example in Figure 1.2, the word “Data” is written into cache array but read out appeared as “D0qa”.The test on the cache will be performed through read and write operation in different ways on the cache cell to identify the cache cell health condition. The cache defect can be either happening in write path, read path, write and read path or on the transistor itself. Different cache testing methodologies are needed in failure isolation process for the FAE to narrow down the failing suspect and form a precise hypothesis.

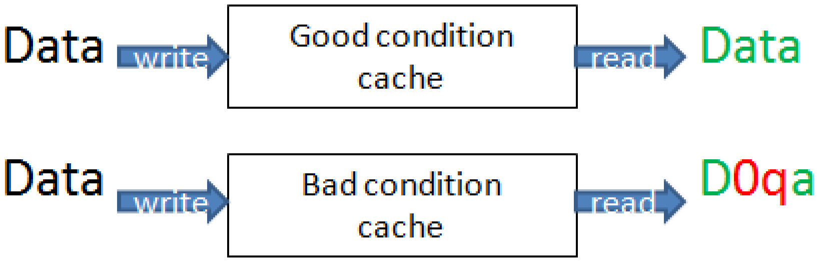


Figure 1.2: Cache Testing Concept

Sufficient data collection during testing is needed for FAE to carry out successful failure isolation. Typically, the HVM (high volume manufacturing) units from production floor will go through a simple testing which only indicate the pass or fail on the units due to massive number of unit need to test in short time. The failing unit will be sent to FAE for further investigation. The main purpose of further investigating is to root cause the failing units and improve the manufacturing yield. At the same time, the Quality and Reliability Engineer (QRE) will perform burn-in and stress test on the silicon device to make sure the silicon devices meet certain Quality Ensure (QE) qualification. After the QE stress testing, a portion of the good condition silicon may fall out as well, the FAE needs to investigate the reason of failing.

When the FAE received the failing unit from production floor or QRE, FAE only able to know the unit failing condition. The details information such as failing bits in array and bits failing signature are not provided. The data provided by production floor or QRE can be categorized as poor testing resolution data because the team only concern about pass or fail result in testing. Thus, in this dissertation the focus is on the importance of high resolution test pattern in term of carrying out a successful failure analysis.

The “detail testing” part in Figure 1.1 is referring to the details of data collection and more failing information extracted from the failing unit. The FAE will test the failing unit with extra test parameter/temperature surrounding. The testing detail level able to identify all the failing bits/signature in cache cell. Without the “detail testing” part in Figure 1.1, FAE unable to plan an effective failure analysis plan with information provided by production floor and QRE. Only the high resolution test pattern have the capability in identifying the failing characteristics in cache array. The high resolution data collection is important for the FAE to observe the failing signature and pin point the failing location in cache region for leading the FAE to perform successful failure analysis.

1.2 Problem Statement

In term of carrying out a successful failure analysis, FAE needs to obtain the details of failing information from the silicon device. Information such as stressing temperature, burn-in data, stress test pattern information and failing characteristic are needed from QRE when the silicon device failed QE stress test. For the silicon device which fail in production floor testing, only the test flow pass or fail info [3] is available for FAE. The Figure 1.3 illustrates the test flow that carry out in the production floor testing.

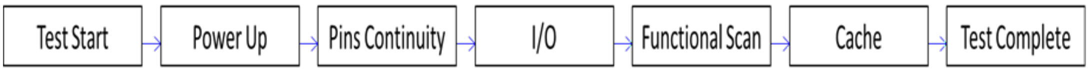


Figure 1.3: Silicon Device Test Flows in Production Floors

For cache type failure, FAE can use optical tools, in-chamber probing and circuit analysis technique to perform FAFI, but without the detail failing information provided, the probability for FAE to find out the defect signature and defect bits in cache array are nearly impossible [4]. The FAE able to estimate the big area of failing array with low resolution test pattern data and continue FAFI with the assisting of optical tools. The actual failing location can be possibly found but only limited to the solid failure mechanism [5]. Normally, the marginal failure and the defect that occurred on controller side on the array will be hard to detect by using optical tools. FAE needs to get an overall failure of cache array to isolate down the possible failure location.

Because of the advance of Nano-technology, the transistor size in the silicon devices are getting smaller. The marginal failure which depending on temperature, test voltage and test frequency is getting frequently. Without the high resolution test pattern to map out all the possible failing bits in cache array, FAFI analysis will be more difficult because the number of failing bits might be vary with the temperature surrounding, test voltage or test frequency. The high resolution test pattern can map out all the failing bits in different condition for FAE to carry out accurate circuit analysis and find out all the possible defect which will cause the related marginal failure.

For enabling the failure analysis capability of cache array, FAE needs to find out a test methodology which can increase the test resolution for obtaining more failing information from the cache array such as failing signature and failing location in cache array. FAE needs to generate or enable the high resolution test algorithm for obtaining failing signature and failing bits information to carry out the successful FAFI analysis.

The low-resolution test pattern used in production floor or used by QRE did not contain the Capture Test Vector (CTV) function. CTV function is used to enable the status register in BIST circuitry; every test access point in BIST will contain a status register for status bit data storage. The status bit information will be recorded in status register if the CTV function of BIST was enable. More time needed for recording down all the test vector information is the main reason of the low resolution test pattern is being used in production floor.

Thus, this dissertation will be focusing on the importance of high resolution test pattern in FAFI process. The resolution of cache region test pattern is increased by adding the CTV in test algorithm, and the high resolution test pattern will be implemented in ATE for testing. At the same time, the created high resolution test pattern will be validated with silicon device and applied in real FAFI cases.

1.3 Objective

The objectives of this project are:

i. To improve the failure analysis capability on cache region by increasing the test pattern resolution in Automated Test Equipment.

ii. To develop the effective technique in debugging the cache types failure.

iii. To validate the proposed high resolution test algorithm.

1.4 Scope

There are multiple methodologies in cache debugging, but this dissertation will mainly focus on increasing the test resolution on BIST architecture by editing the CTV in test algorithm which targets cache failure. The high resolution test algorithm will be implemented in ATE to carry out the real silicon device failure analysis.

The scope of dissertation project will cover the methodology of generating the high resolution test pattern from scratch. The generated high resolution test pattern will be integrated in ATE for real silicon device testing and the output result need to be correlated with low resolution test pattern. A real case study and a real sample validation will be shown at the end to prove the high resolution test pattern implementation working in ATE.

1.5 Contribution

The main contribution for this dissertation is the improvement of FAFI process on the cache type failure in semiconductor industry. With the present of high resolution test pattern in targeting cache failure, the FAFI process will be more efficient and able to detect the failing bits in cache array accurately.

1.6 Thesis Organization

This dissertation report consists of five chapters and all the content will be briefly described in every paragraph from the beginning of chapter to the end of chapter.

Chapter one describes the background of the dissertation project, a brief overview of FAE roles to ensure the product meet the QE requirement and the needs of high resolution test pattern in FAFI process. The problem statement described the problem raised for FAE during the FAFI process and the concept behind of the dissertation. While the objective and scope described the final outcome of the project and a certain level of achievement. Contribution section highlighted the needs of high resolution test pattern in semiconductor industry, and the lacking of failure analysis capability during product improvement stage in targeting cache failure.

Chapter two consists of the study of literature review from other related topics. The literature review includes the specific knowledge which required in understanding the dissertation project such as BIST architecture and cache architecture. At the same time, the literature on the technique used to carry out the dissertation project is reviewed as well. The different of failure analysis methodology used in targeting on cache failure as well as the effectiveness of the method is referenced in this section also.

Chapter three is the main part of the dissertation report. It contains the methodology and technique used in completing the dissertation project. The methodology will start from the scratch once the FAE contains the collateral file until the high resolution test pattern is produced and integrated into ATE. The steps of using the integrated high resolution test pattern in ATE will be explained in debugging a real case silicon failure and test for proving the produced high resolution test pattern is actually working well.

Chapter four is mainly discusses about the outcome of Chapter three. The outcome of Chapter three can be categorized into test validation outcome and real case silicon failure analysis outcome. The test validation is done by using self-induce silicon failure; the silicon is tested in ATE by using high resolution test pattern and expecting the output result is correlated with the self-induce failure. The high resolution test pattern which integrated in ATE is used to debug a real case silicon failure. The failure analysis result is shown in this section which proven the high resolution test pattern able to capture the failure which induced by other factor.

Chapter five concludes the dissertation project and discussion on the further improvement plan on this dissertation project.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The resolution of test pattern plays an important role in FAFI for root cause the failure in silicon device and improves the manufacturing yield and quality of silicon device in semiconductor industry [4]. The FAFI process become more challenging because of an advance fabrication technology which lead to the increasing of transistor count, smaller geometry size and more interconnection layer in a silicon device. With the advancement of fabrication technology, the fault model happened in cache region are getting less visibility and the major fault model is not a solid or single bit failure in cache array. The test pattern resolution become more important in identifying the failing pattern/signature in cache array for FAE to make a smart guess to isolate down the ROI (Region of Interest).

This chapter contains the needed knowledge, information and background for reader to understand the concept behind this dissertation project. The methodologies in using the BIST for cache array debugging together with the concept of using memory BIST to produce the different resolution of test result is also discussed in this chapter. Furthermore, the related case studies in cache debugging are included in this chapter as well for the reader to understand the important of high resolution test pattern in cache debugging.

2.2 Failure Analysis and Fault Isolation (FAFI)

Failure analysis can be defined as a process in finding and determining the root cause of physical defect in silicon device. It is nearly impossible to overcome the reliability and production yield issues without understanding the root cause of these problems. Therefore failure analysis becomes a critical element in semiconductor industry for ensuring the product meet the quality, reliability and high yield expectation by studying the failure occurred in silicon devices. For instance, better design solution was proposed to improve the reliability, performance and the production yield as a good failure analysis result [6].

Now a days, the failure analysis jobs task are getting more difficult because of the complexity of the IC design that is getting higher over the years. From the IC technology development trend, the expectation on the number of transistor integrated in an IC devices will be dramatically increase. This situation may lead the changes in silicon device such as increased I/O pin counts, additional interconnection level, larger die size, higher integrated level and mixed analog and digital circuitry [7].

The increasing of circuit complexity and the changes in silicon device will result in the high density transistor inside a silicon device and causing the defects size of transistor in the layout getting smaller and hard to locate the fault. This situation is shown in Figure 2.1 [4]. The FAE required more advance failure analysis technique and laboratories tools to find out the defect in the silicon device.

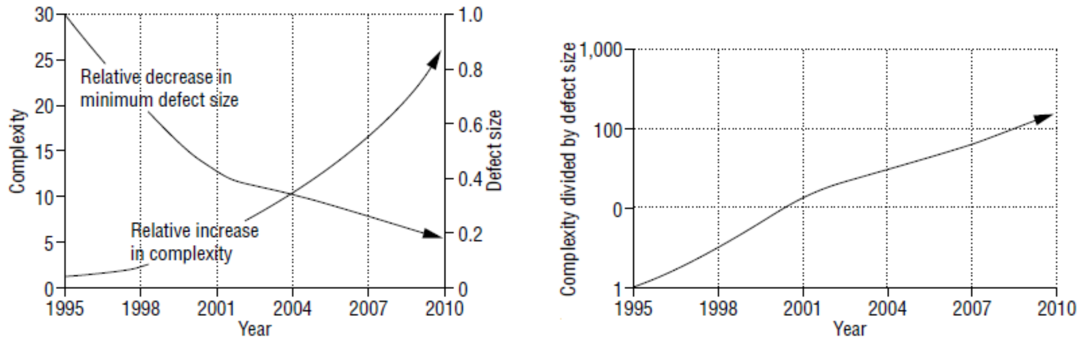


Figure 2.1: Complexity versus Minimum Defect Size and Relative Increase in Localization Difficulty [4]

The test and diagnosis tools play an important role for the FAE to carry out fault localization and fault isolation from a million of transistor [4]. Fault isolation and fault localization is the most critical part in the failure analysis process because it will affect the finding result and the decision of technique used to find out the physical defect in the silicon device. An inaccurate failure isolation and fault location will cause the waste of resources and silicon devices cannot be tested any more after go through the destruction FA.

Before a complex IC starts to build, the DE and FAE already start to think about the testability [7] of the product at the back end before sale into the market. The DFT circuitry in the silicon device adds the value for PDE, QRE and FAE to overcome the quality and reliability issues at the back-end process. Normally, the PDE and FAE will fully utilizing the DFT circuitry to perform testing on the product and extract the details information from the silicon device.

FAE required high controllability and observability when running the test/diagnosis tools in order to understand more about the failing characteristic and signature. The high controllability and observability diagnosis result may help FAE to perform fault localization accurately and estimate the fault model of the defect. In cache testing case, the high resolution test pattern means high observability, the test pattern able to pull out the register status, failing address (row & column) and failing bits information.

2.3 Cache Architecture

Cache is similar to RAM (Random Access Memory) which is a part of data storage in the computer which store the frequently used data and can be accessed in short time [8]. But, the cache memory is the fastest and nearest to CPU compared with RAM, because the cache is integrated into the processor which can define as nearest to CPU and the access time is shorter compare with RAM. Cache is different from other types of memory such as hard disk, pen-drive or computer internal storage that able to store the data permanently, due to the data stored in cache will be lost once the power supply is removed.

A cache cell can be categorized into Static Random Access Memory (SRAM) and Register File (RF) in the industry standard, while it has own architecture to perform write and read data operation from the cache cell. Cache component includes array, decoder, timer and I/O (Input/Output) shown in Figure 2.2.

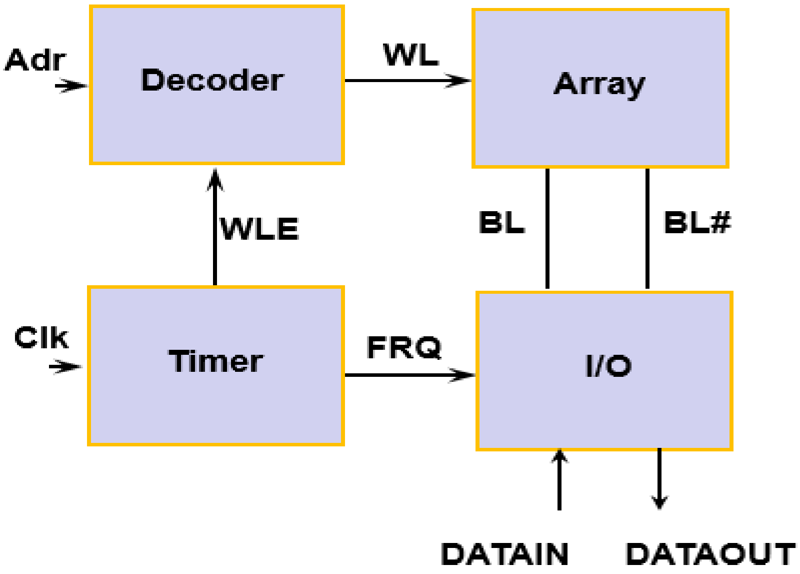


Figure 2.2: Generic Cache Array Component

The decoder will decode the address provided by the user for accessing the data in the desired location, the WLE (Word Line Enable) of the decoder enable the row address component in cache cell to perform write/read operation at the desired location. The timer is for data sampling propose, the timer controls the frequency of the data flows in and out while the I/O act as an interface circuitry for external and internal data flows in and out. The array part is the main component of the cache component which all the data will be stored in the array part.

The array is made out of multiple cache cell [9] shown in Figure 2.3. The row of array is controlled by the word-line (WL) while the column of array is controlled by bit-line/bit-line bar (BL/BL#). In term for the data write into the array cell, the WL must be enable first then the BL/BL# will be enable next for the data to write into array cell according to address provided.

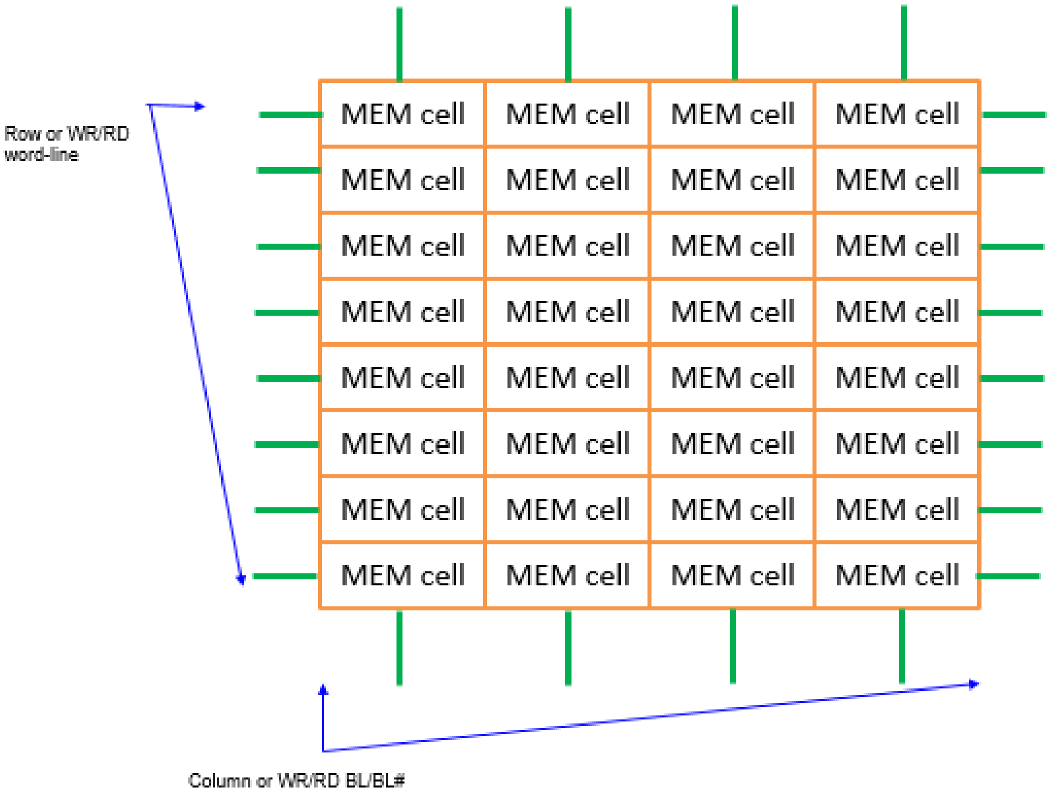

Row or WR/RD word-line
MEM cell MEM cell MEM cell MEM cell
MEM cell MEM cell MEM cell MEM cell
MEM cell MEM cell MEM cell MEM cell
MEM cell MEM cell MEM cell MEM cell
MEM cell MEM cell MEM cell MEM cell
MEM cell MEM cell MEM cell MEM cell
MEM cell MEM cell MEM cell MEM cell
MEM cell MEM cell MEM cell MEM cell
Column or WR/RD BL/BL#

Figure 2.3: Array in Cache Component

2.3.1 SRAM Architecture

Typical CMOS Static Random Access Memory (SRAM) cell are made by 6 transistors, shown in Figure 2.4. A SRAM is formed when 2 NOT gate is connected each other. In SRAM circuitry, the output of M1 and M2 are connected with the input gate of M3 and M4 while the output of M3 and M4 are connected with the input gate of M1 and M3 which form the simplest 1 bit SRAM cell.

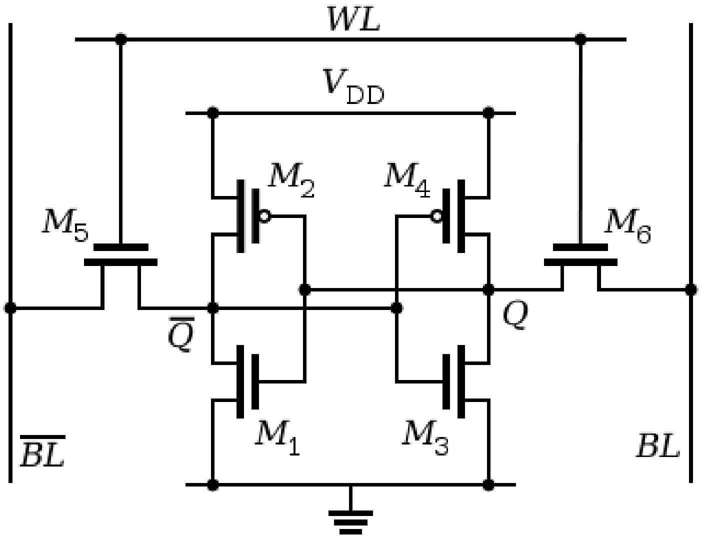


Figure 2.4: 6 Transistors CMOS SRAM Cell [10]

M5 and M6 act as control transistor that allows the data to write in and read out from the 4 transistor inside. M1, M2, M3 and M4 combine together can form two CMOS inverter which connect with each other. The data flow in the two inverter loop to ensure the written data get amplify when pass through the inverter; if not the written data amplitude signal will become weaker over time.

In write operation, the input data will flow through write driver which connected to bit-line (BL) and bit-line-bar ($\overline{BL}$) . When the data is ready for write operation, the word-line (WL) will be enable for the data flow into the 2 CMOS inverter for data storing purpose.

In read operation, the BL and $\overline{BL}$ have to be pre-charged to ensure the NMOS transistor in the CMOS inverter will create a potential different with M5 or M6 transistor for the data to flow out the array. The flow out data in BL and $\overline{BL}$ will have a small voltage difference, a sense amplifier circuitry will sense which line has the higher voltage and thus determine whether there was 1 or 0 is stored in the SRAM.

2.3.2 RF Architecture

Register File (RF) cell is the enchantment of SRAM cell which consist of multiple read and write port [11]. Unlike SRAM cell, the operation of read and write are controlled by similar port/controller transistor M5 and M6 shown in Figure 2.4. The size of RF memory cell is bigger compare with SRAM cell which made by more transistors. The data access time of RF cell is also faster compare with SRAM cell, because the *WL* and *BL* of the RF cell do not need to undergo pre-charge stage to write in and read out the data.

The single-ended Read/Write RF cell is shown in Figure 2.5 which is made out of two read ports and one write port. The write in operation will be slower compare with the read out operation. When the write operation is performed, the *wrEN* transistor will be turned on for the data flow into the node “A”, and then the I1 and I2 loop inverter will hold the data which same as SRAM to store the data. During the read out operation, the *wordRD1* and *wordRD2* need to turn on the N3 and N4 transistor in term for the data flow through the N3 and N4 to *bitRD1* and *bitRD2* port. The data flows in *bitRD1* and *bitRD2* will be decoded at the end-stage of read operation while the I3 inverter just acts as a buffer component for the storage node [11].

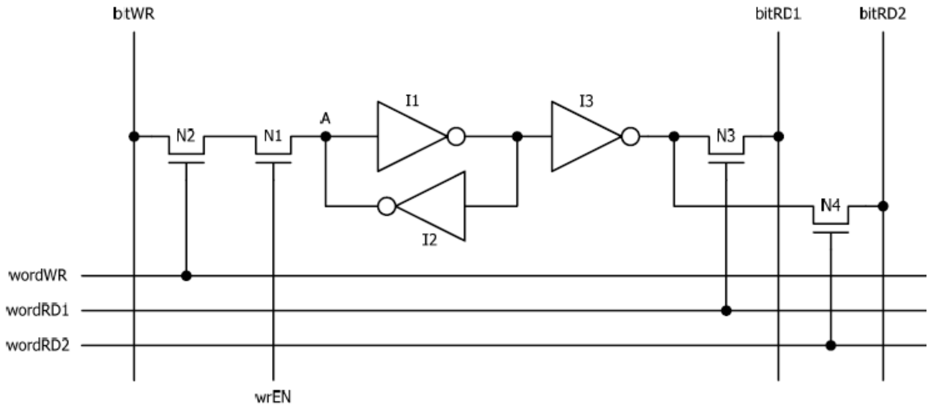


Figure 2.5: Single-ended Read/Write (RF) Cell [11]

2.4 Cache Array Fault Model

The cache array failure can be categorized into few fault models, such as stuck at fault, address decoder fault, coupling fault, transition fault, and data retention fault.

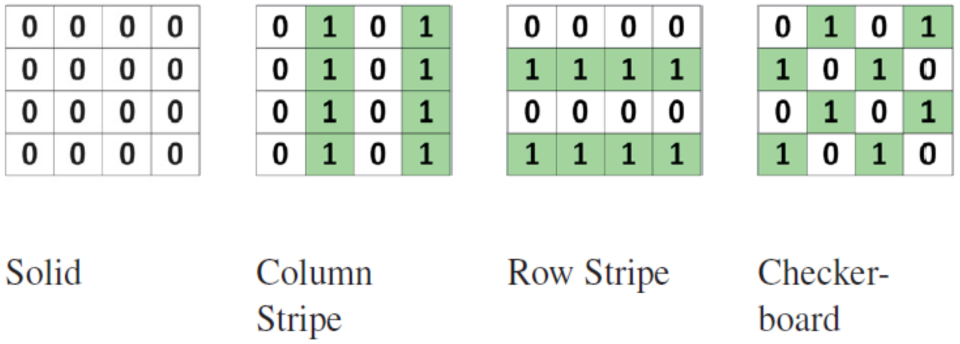
For stuck at fault, the memory cell itself can be stuck at 0 or stuck at 1, it normally happened when one of the signal shorts to the VCC or VSS [12]. Address decoder fault can be defined as the failure which happened in address decoder parts. This types of failure may cause the wrong address accessing, no address accessing, or inaccurate multiple address accessing. This types of cache array failing signature will cause the multiple column, row, and random bits failure [12]. The coupling fault is the failure memory cell that cause a void failure to the neighbor cell and cause it to go into erroneous state. It may cause the void failure isolation because the root cause cannot found in the neighbor cell [12]. In transition fault, the fault will cause the memory cell state go into single way only (single time access only). For example, if the cell contains 1 or 0 has been written to it, then it cannot perform transition back to 1 or vice-versa. This kind of failure might happen in word line or bit line of the cache array [12]. At last, the data retention fault is the fault which cause the data logic stored in the memory cell will lose/change after a period of time, un-match with the data write and data read out after a period of time. It may be happened when the resistive open occurs between two inverter [13].

Based on the mentioned fault models, it is very rare to observe only a single fault model found in cache array now a days. These fault models can occur at the same time in the cache array or the fault model are linked with each other [14]. Although different types of test algorithm can be used to differentiate the fault model happened in a cache array, but more complex test algorithm is needed to detect the fault model which linked with each other and some complicated fault model such as Data Retention fault, Coupling fault and Transition fault. The brief explanation on the test algorithm will be discussed in next section.

2.5 Test Pattern in Targeting Cache Failure

Test pattern or test algorithm for cache can be defined as the test methodology to identify the health condition of the cache array. The test background, and test addressing scheme were included in a test pattern to test the cache array effectively.

The test background is the data "0" or "1" written in each of the cache array. The test background can be all "0", all "1", or "1" and "0" complement in different combination ways, as shown in Figure 2.6.



Solid

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Column Stripe

0	1	0	1
0	1	0	1
0	1	0	1
0	1	0	1

Row Stripe

0	0	0	0
1	1	1	1
0	0	0	0
1	1	1	1

Checker-board

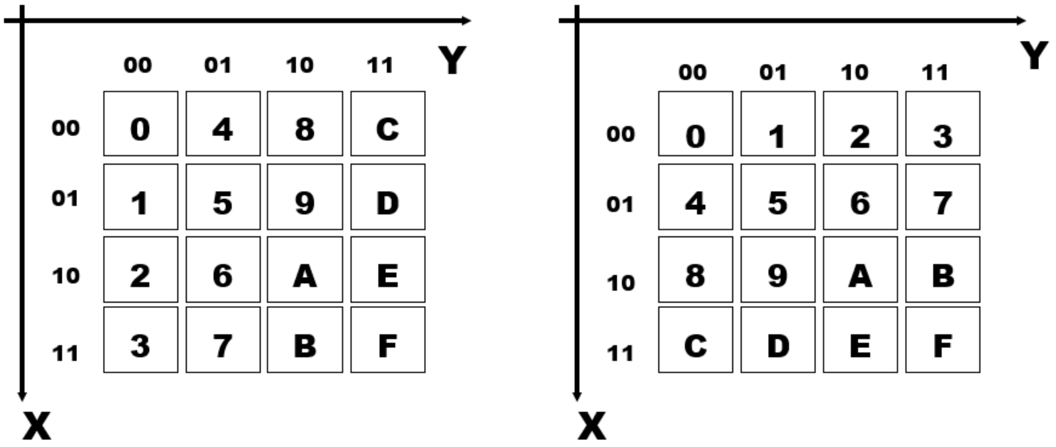
0	1	0	1
1	0	1	0
0	1	0	1
1	0	1	0

Figure 2.6: Test Background [15]

The addressing scheme is the order of the address for the data to write/read in the cache array during the test application. The test addressing scheme can be described as Fast-row, Fast-column, Gray Code and Address Complement [15]. The types of test addressing scheme are described in the Table 2.1 and illustrated in Figure 2.7, Figure 2.8 and Figure 2.9 respectively.

Table 2.1: Types of Test Addressing Scheme

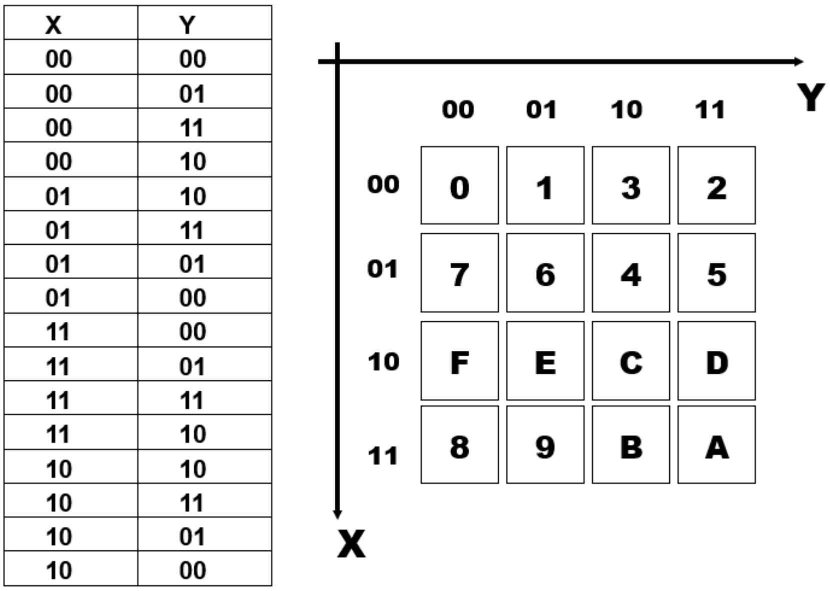
Test Addressing Scheme	Description
Fast-Row	Address WL/row increment before BL/column addressed
Fast-Column	Address BL/column increment before WL/row addressed
Gray Code	address sequence increments or decrements which only one address bit change each time
Address Complement	address sequence increments or decrements which maximum address bit change each time



	00	01	10	11
00	0	4	8	C
01	1	5	9	D
10	2	6	A	E
11	3	7	B	F

	00	01	10	11
00	0	1	2	3
01	4	5	6	7
10	8	9	A	B
11	C	D	E	F

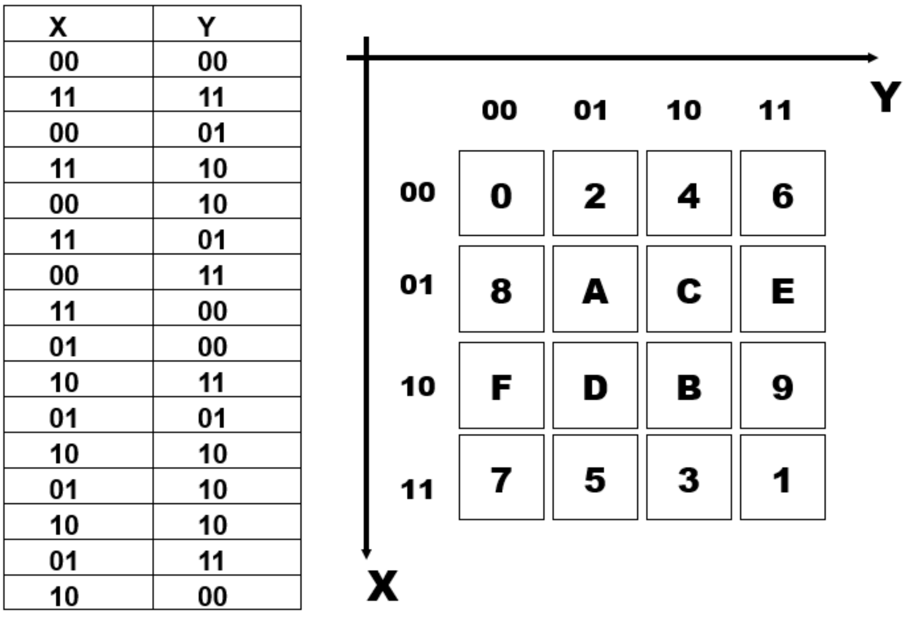
Figure 2.7: Fast Row and Fast Column Test Addressing Scheme



X	Y
00	00
00	01
00	11
00	10
01	10
01	11
01	01
01	00
11	00
11	01
11	11
11	10
10	10
10	11
10	01
10	00

	00	01	10	11
00	0	1	3	2
01	7	6	4	5
10	F	E	C	D
11	8	9	B	A

Figure 2.8: Gray Code Test Addressing Scheme



X	Y
00	00
11	11
00	01
11	10
00	10
11	01
00	11
11	00
01	00
10	11
01	01
10	10
01	10
10	10
01	11
10	00

	00	01	10	11
00	0	2	4	6
01	8	A	C	E
10	F	D	B	9
11	7	5	3	1

Figure 2.9: Address Complement Test Addressing Scheme

The test pattern methodology can be represented in notation form, shown in Table 2.2. The notation represent the sequence and the ways of the testing in cache array.

Table 2.2: Test Pattern Notation

Notation	Description
^	Address increment from address 0 to address n-1
v	Address decrement from address n-1 to address 0
( )	Perform all the operation in the "( )" before moving to next address
,	Operation Separator
w0/w1	Write data "0" / write data "1"
r0/r1	Write data "0" / write data "1"

Different types of test pattern have different specific function to detect the types of fault model in cache array, not every fault model in cache array able to detect by same test pattern [16]. Many types of test pattern were developed to cover most of the fault model in cache array such as MSCAN, MATS, MATS+, MATS++, March X, March Y, March C-, Butterfly and many more can be referred in [17].

In this dissertation project, the MOVI (Moving Inversion) test algorithm, an industry standard test pattern [15], is integrated to high resolution test pattern because this test algorithm can detect most of the common fault model in cache array such as Stuck-At Fault, Address Decoder Fault, Transition Fault and most of the Coupling Fault.

2.5.1 Moving Inversion (MOVI) Test Algorithm

MOVI (Moving Inversion) test algorithm consists of 5 March element. The test algorithm short notation start with addressing order operation (^ or v) and followed by a sequence list of operation in cache array. The MOVI algorithm short nation is shown in Equation 2.1 [16]:

v(w0); ^(r0,w1,r1); ^(r1,w0,r0); v(r0,w1,r1); v(r1,w0,r0)   (2.1)

The test algorithm start with writing in all the data value “0” from address N-1 to address 0 in cache array, the subsequence test algorithm operation will be read out the data which write in previously followed by write in the compliment data value into the cache array. The operation starts from the address “0” to address N-1 and then repeat the same operation from address N-1 to address “0” again.

By performing read and write into the cache array, the failing bits can be detected during the read operation. The failing bits only can be detected during the read operation only. Failing bits can be recognized by comparing the data value write-in and data value read-out just as shown in Figure 1.2. For example, the stuck at one failure mechanism can be detected when the test algorithm trying to write data value “0” into the array bits but the read out data value is “1” which is not expected. The unequal data value write-in and read-out will be categorized as failing bits. The combination read write operation in MOVI test algorithm able to detect the different types of fault mechanism explained in Section 2.4.

The illustration of the MOVI test algorithm operation is shown in Figure 2.10. The data background is solid while the addressing scheme is fast-row.

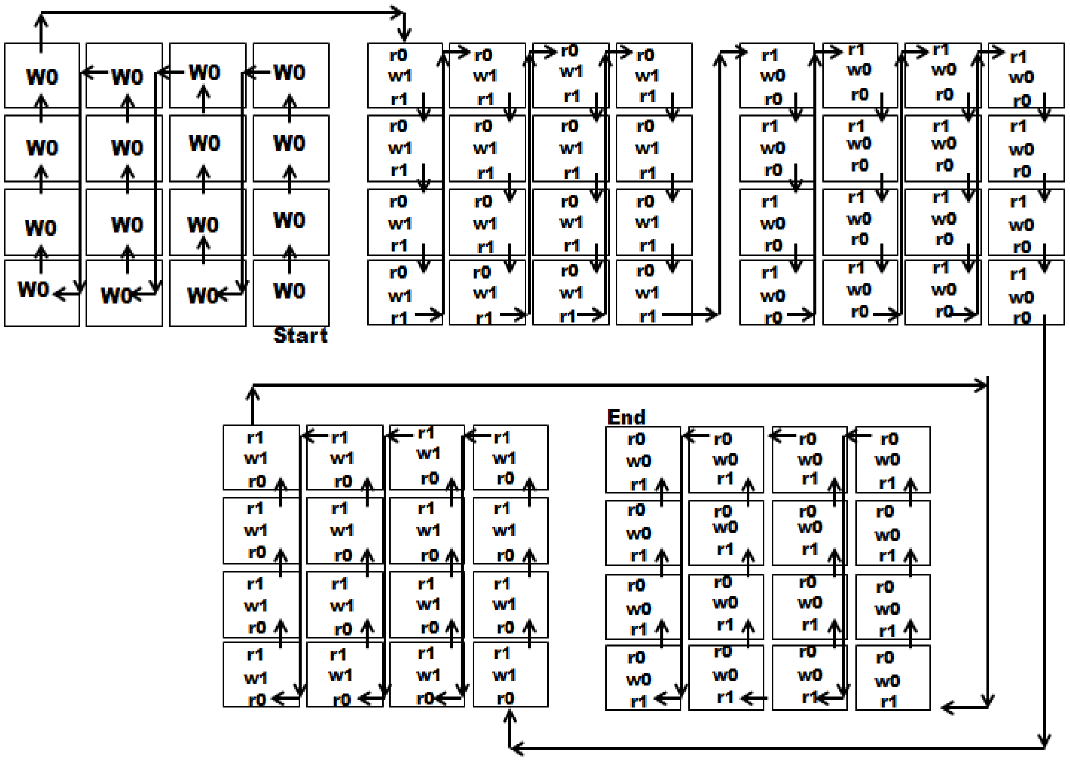


Figure 2.10: The Solid Background, Fast Row Addressing Scheme MOVI Test Algorithm Operation in Cache Array