Low Power Circuits for Multiple Match Resolution and Detection in Ternary CAMs

by

Wilson W. Fung

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada 2004

I hereby declare that I am the sole author of this thesis.

I authorize the University of Waterloo to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Wilson W. Fung

I authorize the University of Waterloo to reproduce this thesis by photocopying or other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Wilson W. Fung

Acknowledgements

First I would like to express my gratitude to my supervisor Professor Manoj Sachdev at the University of Waterloo. Thank you for your supervision, support, kindness, and guidance over the past two years. I would also like to thank Professor Sherman Shen and Professor Catherine Gebotys. Thank you for your positive and valuable comments and suggestions on my thesis.

There are a number of research mates I would like to acknowledge. Definitely the first one goes to Nitin Mohan. He served as a second mentor to me over the past two years. In addition, special thanks to Bhaskar Chatterjee, Andrei Pavlov, Dave Rennie, Nelson Lam, William Chu, and Igor Lovich. Thank you for checking me out for coffee everyday after 3pm. Without you the friends, my life at Waterloo would be dry with no color. I would like to also thank Phil Regier for offering great support on VLSI tools and helping me (and the team) to meet the IC submission deadlines.

My very special acknowledgement is to my parents, my sister, and Clara. Thank you for your persistent love, encouragement, and understanding about my decision on pursuing my Masters at Waterloo, which meant many days and weeks away from home.

The last but definitely not the least, I would like to thank MOSAID Technologies Inc. for funding the initial phase of this project, and Micronet R&D for continuous funding and support. It gives me the opportunity to explore the wonders of low-power circuits for Ternary Content Addressable Memories.

Wilson W. Fung
November 2004
Waterloo, Canada

Abstract

Ternary Content Addressable Memory (TCAM) is a type of associative memory that offers ternary storage and supports partial data-matching. Each ternary bit can be either a "0", a "1", or a "don't care" state. It is a key technology to enable the true power of the next-generation networking equipment and many lookup-intensive applications. Depending on the storage contents, a TCAM search can lead to multiple "matches". A special logic unit, named Multiple Match Resolver (MMR), is required to resolve the best candidate if more than one words indicate a "match". In the early development of TCAM, the capacity was small, with only a few hundred to several thousand words. The design of MMR was relatively easy, and could be realized using static digital logics. Today, the TCAMs for backbone network routers can have up to 512k words. This directly translates to a Multiple Match Resolver and Detector with 512k inputs if the resolution is down to word-level. This definitely makes the design a non-trivial task. In addition, the increasing demands on higher search speed, lower power consumption, tighter memory pitch, multiple match detection, and flexible multiple match readout, are putting more challenges to the design of TCAM. The focus of this thesis is not on the TCAM memory cell design, but rather, it is on the low-power circuit techniques for multiple match resolution and detection in TCAM. Both digital techniques and mixed-signal techniques are presented and analyzed in details.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

With the increasing breakthroughs in fiber optics technology, the wire speed is no longer the bottleneck of a communication system. Instead, the data processing speed is the bottleneck, because these optical signals still have to be converted to electrical signals for routing to their destinations. In the core computer network, each packet of data must be classified and forwarded from one physical link to another within nanoseconds. The recent policy-based routing and Quality of Service (QoS) requirements further increase the number of table lookups needed per packet [1]. It is clear that the conventional software approach based on hash tables is no longer sufficient. One solution is to employ Ternary Content Addressable Memories (TCAMs) for parallel and high-speed data lookup in hardware.

Although TCAMs can offer high-speed lookups (over 100 million searches per second) for next-generation networking equipment, it is not widely employed in today's market [2]. The major hurdles are high power consumption, due to the nature of parallel lookups, and high cost, due to large cell size and large peripheral circuit overheads. Recent publications on TCAM mainly focus on TCAM cell design and pipelining architectures, while little attention is made towards the TCAM-specific peripheral circuitry. Examples of TCAM-specific peripheral blocks include Multiple Match Resolver (MMR), Match Address Encoder (MAE), and Multiple Match Detector (MMD).

The aim of this research is to explore the circuit techniques and architectural techniques for reducing power consumption in MMR, MAE, and MMD. A number of low-power techniques, in both circuit-level and architectural-level, are proposed and presented in this thesis. These circuits have been designed and implemented on silicon using TSMC 0.18 μm CMOS technology.

1.2 Significance of This Work

Some of the materials in this thesis are disclosed in patent documents only. However, their writing styles and presentation styles are sometimes hard-to-follow, with only little comparison or numerical numbers to support their claims. Although most of the prior works on multiple match resolution and detection evolved from the same circuit ideas, nobody has ever tried to generalize or categorize these invented schemes. Note that the design of MMR and MMD circuits are not "standardized" like DRAM or SRAM cell circuits. In addition, there are a number of TCAM vendors on the market, however they all keep the design information as trade secrets and rarely disclosed to the public. The information and analysis in this thesis is the first complete reference in this nature. It contains the design knowledge and results from the author's perspective over 2 years of research at the University of Waterloo.

1.3 Thesis Organization

This thesis is organized in a flow that matches the flow of a TCAM search. Each chapter is written in a self-contained format. Chapter 2 provides background information on TCAM architectures and a high-level description of the Multiple Match Resolver (MMR), Match Address Encoder (MAE), and Multiple Match Detector (MMD). Chapter 3 highlights the logic equations and logic-level optimization techniques for the MMR block. Chapter 4 reviews the prior schemes in cell-based MMR design, and presents the design of a novel 256-bit MMR. Chapter 5 offers comprehensive analysis on different styles and topologies of MAEs. Chapter 6 presents the design of MMDs. Both digital and mixed-signal multiple match detection schemes are explored. Chapter 7 provides the architectural level techniques for sequentially reading out multiple match addresses from the TCAM. Conclusions and Recommendations will be given in Chapter 8 at the end.

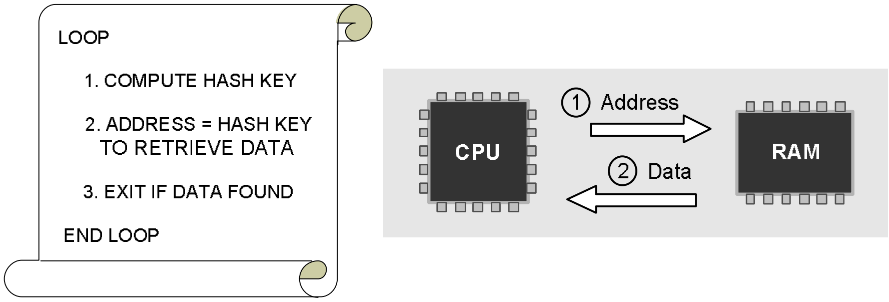
Chapter 2

Ternary Content Addressable Memory (TCAM)

2.1 What is Content-Addressable Memory (CAM)?

Content Addressable Memory (CAM) is a type of associative memory that outgrows from the existing RAM technology. It provides the same basic features such as “read” and “write”, and offers an on-chip parallel search capability against the data contents. Hence, it is appropriate to refer CAM as a hardware “search engine”. In fact, the core of CAM can be built on many types of RAM technologies, including SRAM, DRAM, or even the emerging nonvolatile alternatives. However, SRAM-based CAM (and Ternary CAM) is the leading candidate in the market today, because SRAM offers low leakage, high performance, and a compatible manufacturing process for the CAM-specific peripheral logic circuitry.

In order to understand the CAM search operation, it helps to contrast it with a RAM-based search operation. Figure 2.1 illustrates the high-level flow of an associative search using RAM. In the microprocessor, a software algorithm is running and responsible for finding the result of a given search key. This algorithm has to rely on successive approximations, such as multi-level hashing or binary-search, before hitting the “best match” in the lookup table (in RAM). This iterative and successive process is time-consuming. The worst-case search-time is dependent on the number of entries in the table.



```
LOOP
    1. COMPUTE HASH KEY
    2. ADDRESS = HASH KEY
       TO RETRIEVE DATA
    3. EXIT IF DATA FOUND
END LOOP
```

Figure 2.1: The flow of an Associative Search using RAM

Instead of relying on successive approximations, a CAM search is straight forward with a worst-case search-time independent from the table size. An associative lookup using CAM can be completed virtually within a single clock cycle. This concept is illustrated in Figure 2.2. A local "index" table is stored inside the CAM for parallel and fast indexing. To initiate a search, the microprocessor needs to specifies only the search key. The CAM will compare all its data contents in parallel against the key, and generate an address, associated with the best "match", for reading data in RAM. The retrieved data will be forwarded by the CAM back to the microprocessor. Note that all these intermediate steps are transparent to the microprocessor.

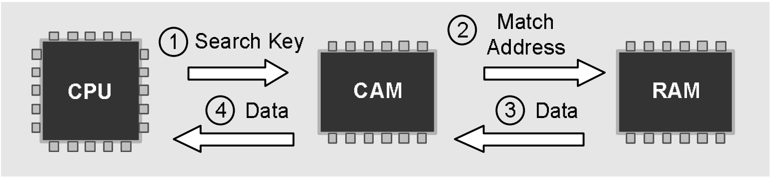


Figure 2.2: The flow of an Associative Search using CAM with Automatic Forwarding [3]

2.2 TCAM Fundamentals

There are two types of CAMs: Binary CAM and Ternary CAM. For Binary CAM, each storage unit is a binary bit, in either logic "0" or "1". For Ternary CAM (TCAM), each storage unit can have 3 states, either a "0", a "1", or a "X" (usually called the "don't care" state or the "masked" bit). While the binary storage units are only capable of performing exact data matching, the additional "don't care" state allows TCAM to offer partial data matching. In data communication systems and robotic systems, partial data matching is required intensively. This attractive feature is the main driver for the boom of the TCAM market. We will focus on static-based TCAMs in the coming sections, because it is the flagship variation of CAM today.

Figure 2.3 shows the circuit schematic of a conventional SRAM-based 16T TCAM cell. A ternary bit is emulated by the combination of 2 binary bits. Thus, this TCAM cell can have a value of either "00", "01", "10", and "11". However, for proper operations, only three of them are used in TCAM applications. Table 2.1 shows the logic representations.

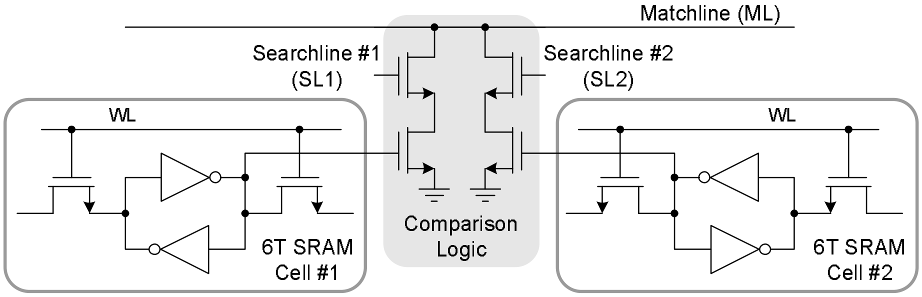


Figure 2.3: A 16T Conventional SRAM-based TCAM Cell [4]

TCAM Cell Value	Logic Representation
00	"X" (Don't Care)
01	"0"
10	"1"
11	Error! (Not Used)

Table 2.1: TCAM Cell Values and Logic Representations

These logic representations are defined to facilitate the TCAM "search" operation. Before the discussions of this operation, let's first take a look at the "Comparison Logic" in Figure 2.3. The "Comparison Logic" consists of two discharging paths. Each path is gated by two NMOS transistors in series. It is conducting only if both gates are "1"s. Prior to a search operation, the Matchline (ML) is pre-charged to a "1". At the on-set of the evaluation phase, each bit of the search key is applied on the Searchlines (SL1 and SL2). If no discharging path is conducting, the ML will remain at a "1". This indicates that this TCAM word is a perfect match to the search key. On the other hand, if the ML is discharged to a "0", the corresponding TCAM word is not identical to the search key. The logic is all that simple. Hence, the "don't care" state is emulated by "00" so that both discharging paths in the comparison logic do not conduct during the evaluation.

Figure 2.4 shows a simple 2 × 2 TCAM array with a Matchline Sense Amplifer (MLSA) connected to each ML. Here, there are two TCAM words in this array, each is 2-bit length. If the word is a perfect match to the search key, none of the comparison logic would be conducting. In this way, the search key can be compared against every single word in the entire TCAM array in parallel. The output of the MLSA, denoted by MLSO, is a "1" if the word is a match to the search key, or a "0" otherwise.

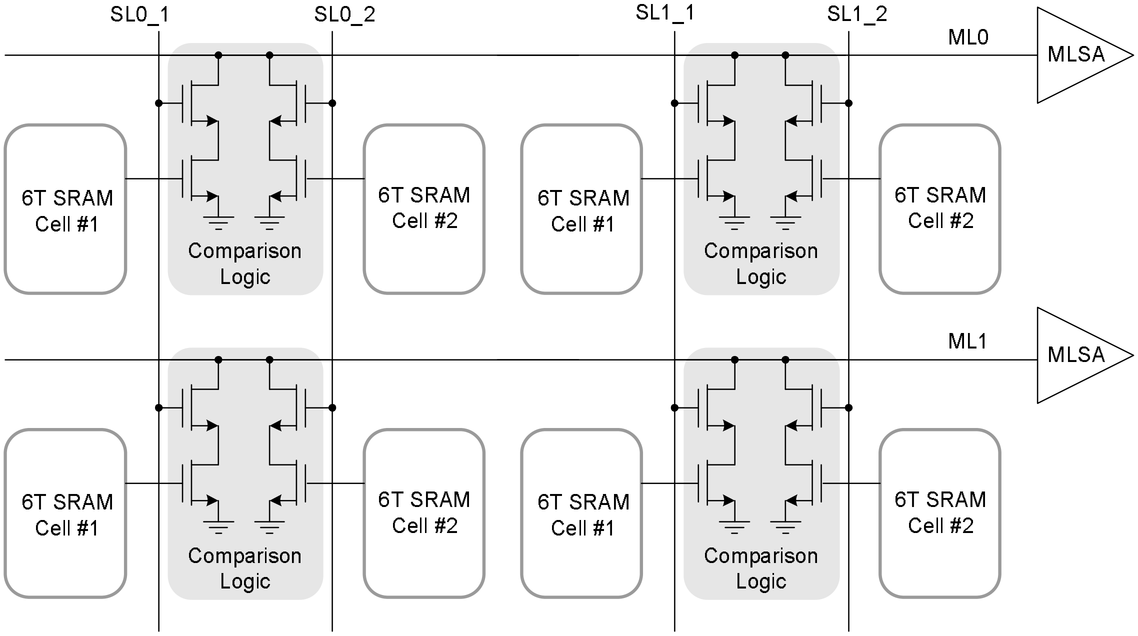


Figure 2.4: The Structure of a 2 × 2 TCAM

Although the example in Figure 2.4 is simple. The same concepts are applicable to a high-density TCAM array with 64k × 144-bit words. There are many different ways in Matchline Sensing and comparison logic design. However, the focus of this thesis is not in these cell array components. The purpose of this section is to set up the ground work for descriptions of the Multiple Match Resolver (MMR) and Multiple Match Detector (MMD) in the coming section. Readers with interest in the design of TCAM core components are welcome to look into [5] and [6] for more details.

2.3 The Flow of a TCAM Search

In Section 2.2, we studied that a TCAM search is initiated by applying the search key on the searchlines. The parallel comparisons are then activated to determine if any word in the array is a perfect match to the search key. The comparison results will be presented at the output of the MLSAs. Definitely, the flow of a TCAM search is not complete up to this stage. The design of the circuits for the remaining stages are the focus of this thesis. Figure 2.5 shows the complete flow in the high level.

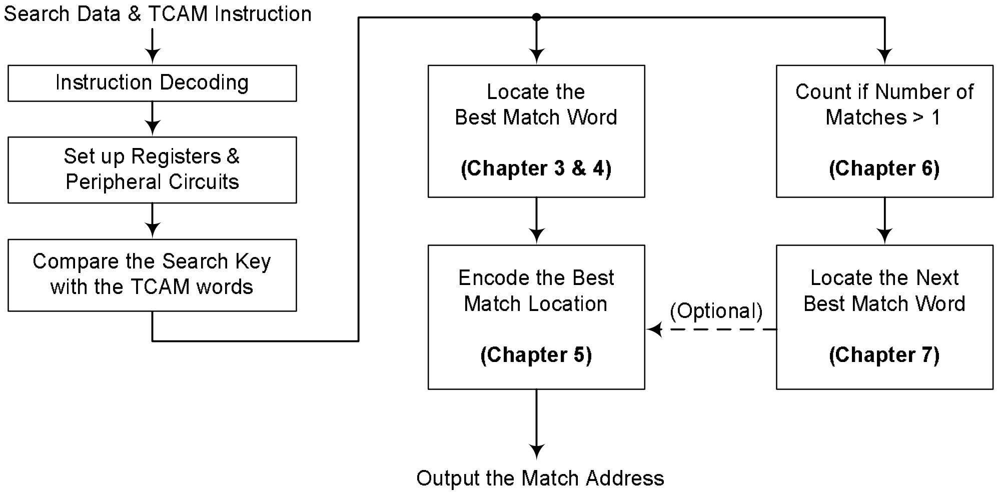

Search Data & TCAM Instruction
Instruction Decoding
Set up Registers & Peripheral Circuits
Compare the Search Key with the TCAM words
Locate the Best Match Word
(Chapter 3 & 4)
Count if Number of Matches > 1
(Chapter 6)
Encode the Best Match Location
(Chapter 5)
(Optional)
Locate the Next Best Match Word
(Chapter 7)
Output the Match Address

Figure 2.5: The Internal Flow of a TCAM Search

Similar to any other parallel operations, a TCAM lookup can lead to resource conflicts due to the possibility of multiple matches. Hence, the next step is to determine the "best match" in a TCAM search. The logics and circuit techniques for multiple match resolution will be the topics of Chapter 3 and Chapter 4. The step following the resolution stage is to encode the best match location into binary format.

Many lookup applications require not only the best match in a search, but the second best and so on. In order to satisfy such demand, a stage called "Multiple Match Detection" is performed in parallel to count if the number of matches is greater than one. This provides the option for the external processor to retrieve the next-best match in a search if required. Chapter 6 and 7 will discuss how multiple matches are detected in a high-density TCAM.

2.4 TCAM Architecture

Figure 2.6 shows a conventional architecture of a high-density TCAM. The TCAM arrays are divided into many small blocks in a hierarchical topology. In order to adapt to the same hierarchy, the Multiple Match Resolver (MMR), Match Address Encoder (MAE), and Multiple Match Detector (MMD) are also divided into small blocks and distributed all over the chip. The local MMR, MAE, and MMD are responsible for the intra-block affairs, while the second level MMR, MAE, and MMD are responsible for the inter-block issues.

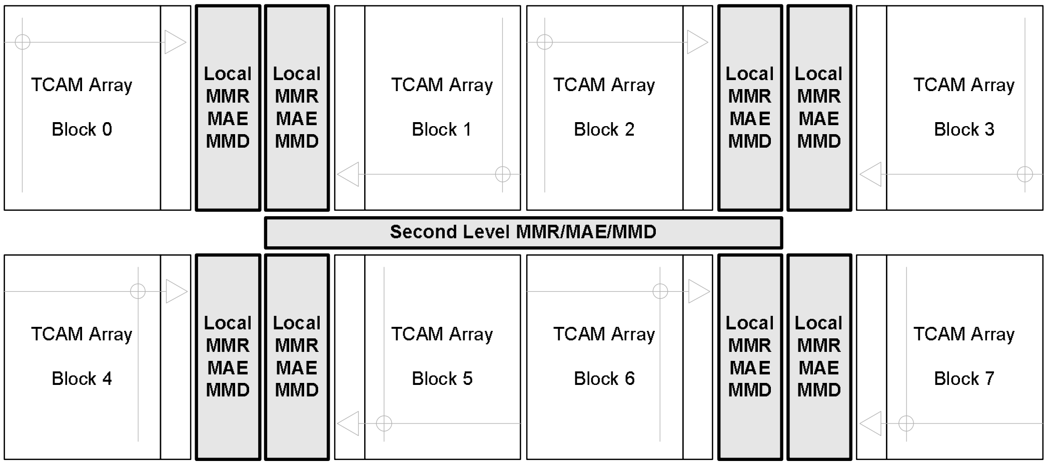


Figure 2.6: The Conventional Architecture of a High-Density TCAM

There are some other TCAM architectures proposed in the literatures, such as the ones in [7], [8], and [9]. However, in this thesis, we assume that a high-density TCAM is structured using the floorplan as shown in Figure 2.6.

Chapter 3

Multiple Match Resolution Basics

The flow of a Ternary CAM (TCAM) search operation has been introduced in the last chapter. Once the search key is compared with all TCAM words, the results must be processed for locating the best "match". In this chapter, we will try to study the logics and science of resolving the best match in a TCAM search. This important step is highlighted in Figure 3.1.

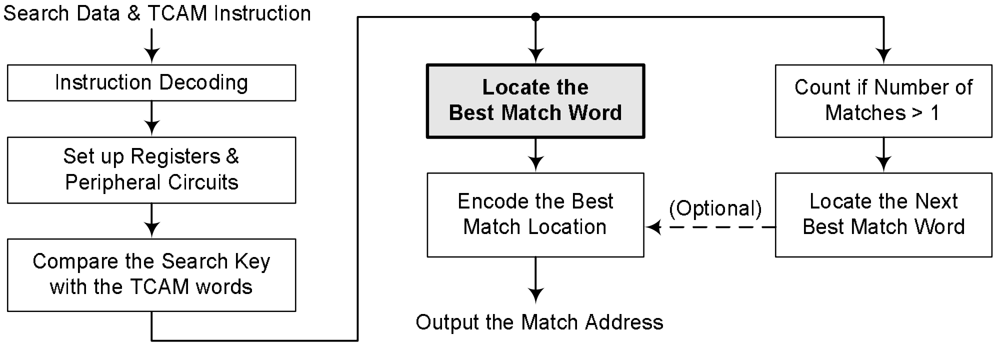

Search Data & TCAM Instruction
Instruction Decoding
Set up Registers & Peripheral Circuits
Compare the Search Key with the TCAM words
Locate the Best Match Word
Encode the Best Match Location
Output the Match Address
Count if Number of Matches > 1
Locate the Next Best Match Word
(Optional)

Figure 3.1: The Role of Locating the Best Match in a Ternary CAM Search

The main focus of this chapter is to provide the fundamentals of Multiple Match Resolution. They include the problem definition, the logic equations, design issues, and architectural optimization techniques. Most of these techniques are independent from the underlying circuit style. Chapter 4 will deal with the circuit-level issues, including the design and analysis of cell-based Multiple Match Resolvers.

3.1 Problem Definition

3.1.1 Direct Interfacing MLSAs to a Simple Encoder

Consider the block diagram shown in Figure 3.2, where the outputs of Matchline Sense Amplifier (MLSA) directly connect to the inputs of a simple digital encoder. It is a well-known principle in digital design that an encoder is functional with at most one input in "active" state [10]. Otherwise the encoder output would be just the bit-wise OR-ed result of all the individually-encoded values. In the case of TCAM, each word can be a match or partial match to a search key. This implies that more than one MLSA outputs can be "active" at a time. Such behavior may violate the rule of encoding, and result to have corrupted match address, as shown in Figure 3.2(b).

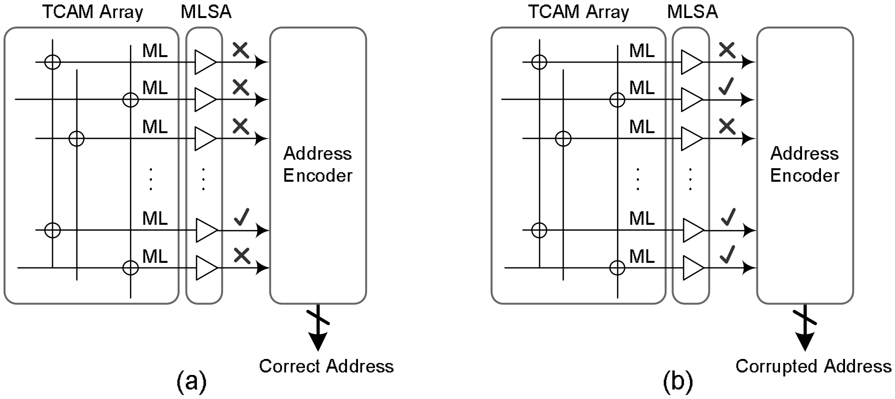


Figure 3.2: Direct Interfacing MLSA to Address Encoder when (a) 1 Match or (b) ≥2 Matches

This undesired behavior urges a need to post-process the MLSA outputs, so that only the best "match" signal can reach the inputs of the encoder. One solution is to employ a priority encoder (PE) for replacing the simple encoder. In brief, each input of a PE has a unique priority value. The priority assignment can be either ascending or descending. When more than one inputs are active, the encoded address refers to the highest priority active input. The design of PE is also a well-known art in digital design, however existing PE implementations are usually rendered based on truth tables. Their resolutions are limited to 8 to 32 inputs only. They are designed for general-purpose applications such as resource arbitration [11, 12].

3.1.2 Dividing a Priority Encoder into Two Blocks

A typical state-of-the-art TCAM ICs can have up to 256k or even 512k words [13]. This translates to 512k MLSA outputs and the need of having a PE with 512k inputs if the resolution is down to word-level. As previously discussed in Section 2.4, such large number of inputs can only be realized through multiple levels of resolution. Even so, each level still needs to resolve 256 or 512 inputs [13]. In order to handle this large number of inputs, the PE is usually split into two blocks: Multiple Match Resolver (MMR) and Match Address Encoder (MAE). Another reason for splitting the PE into two blocks is to facilitate “Sequential Next-Best Match Resolution”, which will be the topic of Chapter 7. Figure 3.3 in the next section illustrates the role of MMR. We will focus on the logic-level optimization techniques for MMR in this chapter.

3.2 The Logics of Multiple Match Resolution

3.2.1 The Conventions and Logic Equations

A Multiple Match Resolver (MMR) is an N-bit input, N-bit output datapath circuit. Its design is very similar to that of a high-speed adder in a microprocessor. Figure 3.3 shows the physical placement of a MMR in a typical TCAM block.

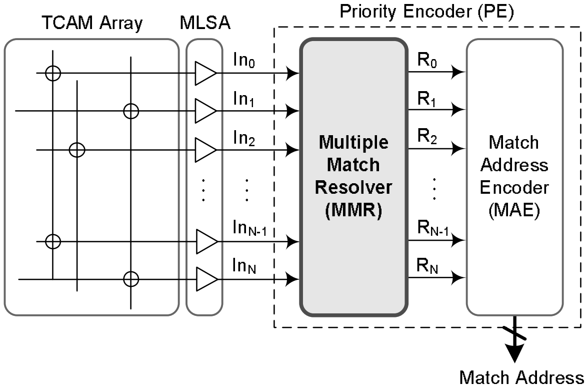


Figure 3.3: Definition of MMR

Each TCAM word is prioritized, and the priority is determined by its physical address. As a convention, the lowest-address TCAM word has the highest priority. It is the responsibility of the application software to store data into the “right” TCAM memory address, so that later on, the

MMR can accurately determine the best match in a TCAM lookup.

From this section forward, we will follow the active-high logic convention. That is: a logic “1” indicates a “match” condition, and a logic “0” represents a “no match” or “mismatch” condition. The resolved output bit, denoted by R, is a “1” if (i) the corresponding input bit is signaling a “1”, and (ii) all higher priority input bits are zeroes. Only the highest priority “1” will be copied to its corresponding output bit. The outputs of a MMR can be described using the following logic expressions [11, 14].

$$\begin{aligned}
R_0 &= In_0 \\
R_1 &= In_1 \cdot \overline{In_0} \\
R_2 &= In_2 \cdot \overline{In_1} \cdot \overline{In_0} \\
&\vdots \\
R_N &= In_N \cdot \overline{In_{N-1}} \cdot \ldots \cdot \overline{In_1} \cdot \overline{In_0}
\end{aligned}$$

They can be generalized using Equation (3.1), where i ϵ {0, 1, ... N}. N is the total number of MMR outputs.

$$R_i = In_i \cdot \overline{In_{i-1}} \cdot \overline{In_{i-2}} \cdot \ldots \cdot \overline{In_1} \cdot \overline{In_0} \tag{3.1}$$

3.2.2 Static Logic Implementation

Early works on MMRs were direct translations of the above equations into complementary CMOS circuits. However, when N is large (for example, N = 256), a static gate will reach its intrinsic performance limit. A number of reasons are given below.

1. The propagation delay of a static CMOS gate deteriorates rapidly as a function of fan-in. The larger number of transistors rapidly increases the capacitance at the output node and at the internal nodes. An approximation of how the fan-in (FI) and fan-out (FO) influence the propagation delay of a complementary CMOS gate can be approximated using Equation (3.2) below.

$$t_p = \alpha_1 FI + \alpha_2 FI^2 + \alpha_3 FO \tag{3.2}$$

where FI = N and the constants α_1, α_2, and α_3 are weighting factors, which are dependent on the CMOS technology [15]. Such quadratic dependence on fan-in significantly degrades the performance of the wide-input AND gate when N is large.

2. The capacitive loadings on the preceding stage (ex. MLSA) are highly unbalanced. While $MLSA_0$ drives a fan-out of N, $MLSA_N$ drives a fan-out of 1 only. This imply that the MLSA output buffer must be sized to drive N fan-out load in the worst case if the MLSA cell is replicated. *(Note: For now, assume MLSA is directly interfacing with MMR. Although this is not the case, the same argument applies to the sizing of the buffer following the MLSA)*

3. The MMR layout would be highly irregular. Pitch-matching these large fan-in static gates to the MLSA outputs is also very challenging. The design will be limited by the complexity of inter-connections when N is large.

As a common practice, Equation (3.1) can be divided into a tree of smaller AND/OR logics over a number of stages. However, the layout is still highly regular. These static circuits are definitely not suitable for fine-pitch and high-density TCAMs. Modern MMRs are all implemented using dynamic circuits, with pass transistor chains and wired-OR logics for ease of pitch-matching to TCAM array.

3.3 Techniques for Datapath Logic Optimization

As described in Section 3.2.1, a MMR is a datapath circuit similar to circuits like adder, multiplier, and shifter in the arithmetic logic unit. Intuitively, we can apply similar datapath optimization techniques to reduce the worst-delay of a wide-input MMR. The conventional techniques include "bypassing", "fixed-size lookahead", and "progressive-size lookahead". Although most of them are well-known concepts from traditional logic design, the emphasis here is to study how they are employed in the context of multiple match resolution. A modified version of lookahead technique, named "folding", will be introduced in Section 3.3.3. These logic optimization techniques are generic and not limited to any specific circuit-level implementation. They are the foundations in the design of high-speed MMRs.

3.3.1 Lookahead and Bypassing

Unlike the case in adder circuits, "lookahead" and "bypassing" in the context of multiple match resolution are somewhat overlapping. In brief, the bypassing in adder circuits employs the "Propagate" signals only, while the lookahead scheme utilizes both "Propagate" and "Generate" bits [15]. However, for multiple match resolution, the resolved output bit, R, depends only on the "input" bits. Hence, these two concepts are generally inter-changeable.

Single-Level Lookahead

In Section 3.2.1, we have studied that the MMR outputs are represented by Equation (3.1). The AND operation implies "transistors connected in series". The OR operation implies "transistors connected in parallel". According to the De Morgan's Law [10], we can group a number of AND operations and translate them into OR-type 'lookahead" signals. A simple 4-bit MMR with lookahead is illustrated in Figure 3.4(b).

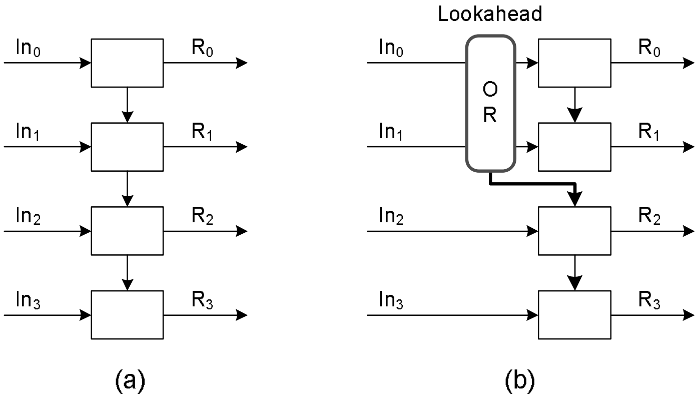


Figure 3.4: Logic Optimization: (a) Linear Ripple (b) With Simple Lookahead

Assume that each block in the diagram consumes 1 unit delay, the introduction of the lookahead signal reduces the worst-delay from 4 units to 3 units in this example. Below shows the corresponding logic equations where $LA_{i-0} = In_i + In_{i-1} + \ldots + In_0$.

$$\begin{aligned}
R_0 &= In_0 & & R_0 = In_0 \\
R_1 &= In_1 \cdot \overline{In_0} & & R_1 = In_1 \cdot \overline{In_0} \\
R_2 &= In_2 \cdot \overline{In_1} \cdot \overline{In_0} & \Longrightarrow \quad & R_2 = In_2 \cdot \overline{LA_{1-0}} \\
R_3 &= In_3 \cdot \overline{In_2} \cdot \overline{In_1} \cdot \overline{In_0} & & R_3 = In_3 \cdot \overline{In_2} \cdot \overline{LA_{1-0}}
\end{aligned}$$

In order to further reduce the worst-case delay of the circuit, one can introduce the lookahead signals in the topologies shown in Figure 3.5.

$$\begin{aligned} R_0 &= In_0 & \qquad R_0 &= In_0 \\ R_1 &= In_1 \cdot \overline{In_0} & \qquad R_1 &= In_1 \cdot \overline{In_0} \\ R_2 &= In_2 \cdot \overline{LA_{1-0}} & \qquad R_2 &= In_2 \cdot \overline{LA_{1-0}} \\ R_3 &= In_3 \cdot \overline{LA_{2-0}} & \qquad R_3 &= In_3 \cdot \overline{In_2} \cdot \overline{LA_{1-0}} \\ R_4 &= In_4 \cdot \overline{LA_{3-0}} & \qquad R_4 &= In_4 \cdot \overline{LA_{3-2}} \cdot \overline{LA_{1-0}} \end{aligned}$$

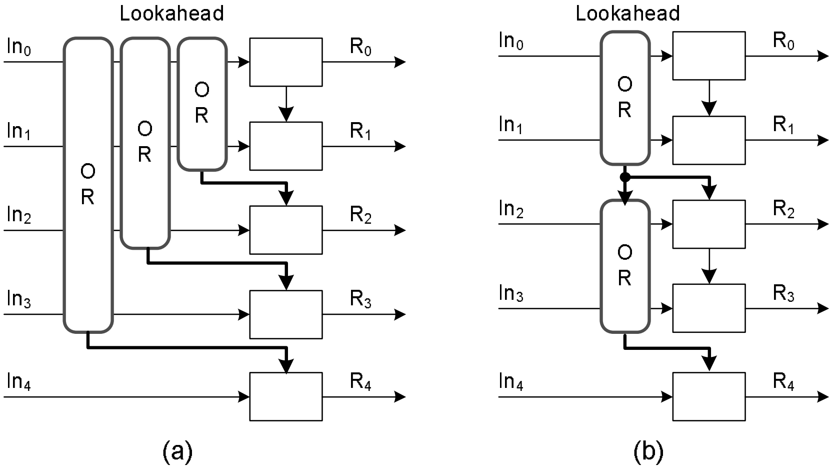


Figure 3.5: Single-Level Lookahead: (a) Ideal Case (b) In Practice

The topology in Figure 3.5(a) shows the ideal case, where a unique lookahead signal for each bit is available. In reality, this is impossible. The reasons are similar to the deficits of having large fan-in static gates as described in Section 3.2.2. Hence, the lookahead signals are usually propagated through the "lookahead level" as shown in Figure 3.5(b). However, this ripple "lookahead" chain will become the performance bottleneck as well when N is large. The worst case delay is still O(N).

Multi-Level Lookahead

If single-level lookahead is not sufficient, how about 2-level, or even 3-level lookahead? This is exactly the way and the only way to proceed for dealing with very wide-input MMRs. For clarification, a simple 2-level lookahead scheme is illustrated in Figure 3.6.

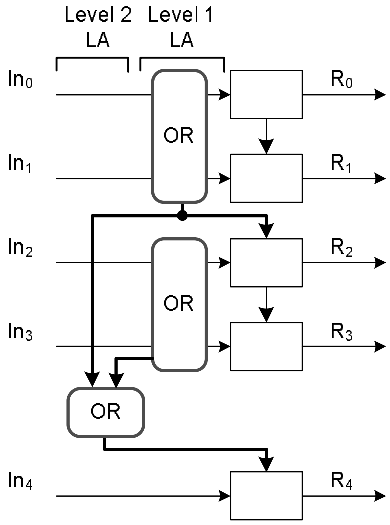


Figure 3.6: Multi-Level Lookahead in MMR

There was a long history in using multi-level lookahead signals to speed up wide-input MMRs. The previous works of note include [7], [11], [14], and [16]. Most of them are similar in nature, with only differences in circuit techniques. Figure 3.7 shows a 256-bit MMR with two levels of priority lookahead. The design was proposed by Yamagata in [7]. It is implemented completely in static CMOS logics.

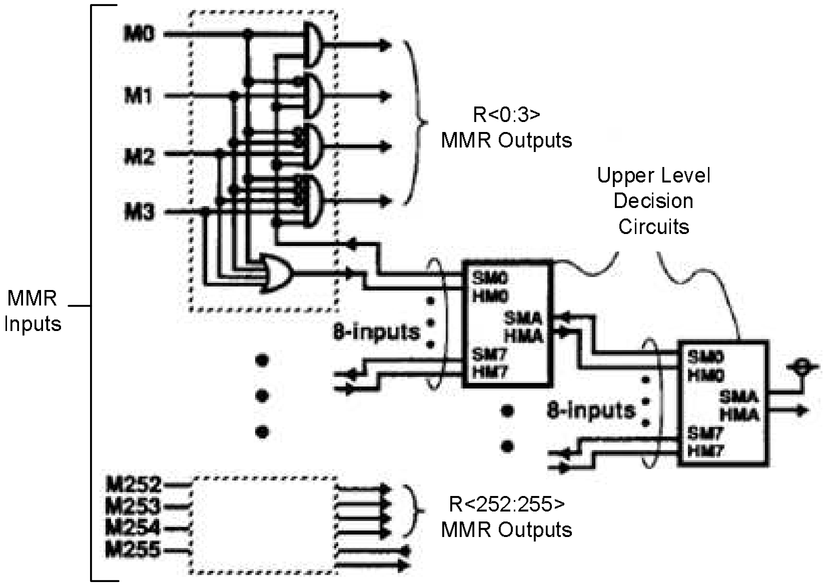


Figure 3.7: A 256-bit MMR with 2 Levels of Priority Lookahead (adapted from [7])

Note that both the MMR cells and the lookahead circuits must be physically laid out in a single column, with inputs on one side, and outputs on the other. One tradeoff of having a large number of lookahead stages is the difficulty in pitch-matching the MMR inputs and outputs to the neighboring circuits (ex. MLSAs, scan-chains, Match Address Encoder etc). In addition, all interconnections must be fit over the MMR cells along the same column of silicon area. A large number of lookahead stages do not always offer a positive gain in performance and circuit efficiency.

3.3.2 Progressive Lookahead

For the topologies described in the last section, the size of each lookahead circuit within the same level is identical. This unfortunately does not lead to the optimal reduction in worst-case delay. The fixed "block size" approach is not taking the ripple delay in the lookahead level into design considerations. Hence, to achieve the optimal and equal delay among all paths in the circuit, one can size the blocks progressively, as depicted in Figure 3.8. Such progressive sizing can "even out" the delay on each individual path. This is analogous to the "square-root" configuration in Carry Select adder design [15].

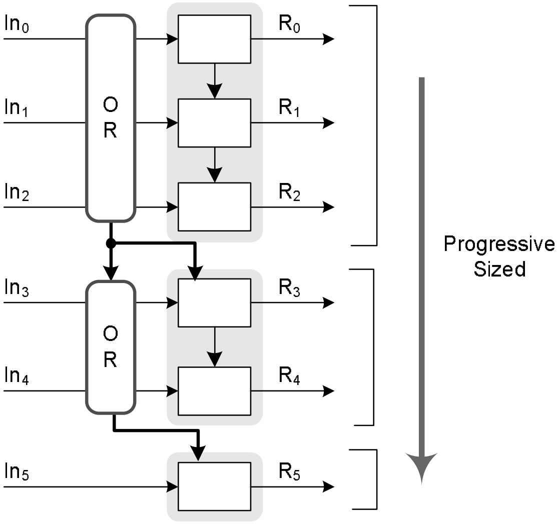


Figure 3.8: Progressive Sizing of Lookahead Circuits

In theory, this simple trick can offer a small amount of delay improvement over the fixed-size lookahead scheme. The improvement is even more dramatic when N is large. The delay of progressive lookahead is O($\sqrt{N}$), while the conventional approach (fixed-size) is O(N) [15]. However, this is only in theory. The slight improvement in speed is offset by two drawbacks, as described in the following.

1. The idea of progressive sizing suggests that each block must be custom-designed. This include custom transistor sizing, custom circuit layout, and custom routing over the MMR cells. Such custom-designed also implies that pitch-matching to the MLSAs and TCAM array would be an issue. In addition, this progressive sizing approach cannot be employed by automated CAM compilers. It also makes design migration difficult from technology to technology.

2. The O($\sqrt{N}$) delay is only true under the assumption that all lookahead circuits (in different sizes) exhibit the same delay.

In conclusion, the progressive lookahead scheme rarely comes into practice in the design of MMR in high-density Ternary CAMs.

3.3.3 Multi-Level Folding

Figure 3.9 illustrates a technique named “Folding” for reducing the worst-case delay of MMR. It was proposed by Huang in [17].

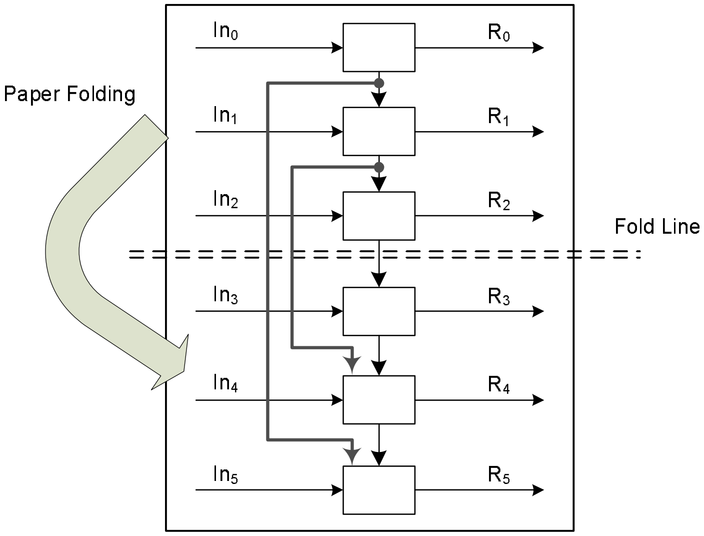


Figure 3.9: The Concept of Paper Folding on MMR Logic Optimization

According to Equation (3.1) previously defined in Section 3.2.1, the worst-case delay is the time for the highest-priority input (In_0) to inhibit the lowest priority input (In_N) if both of them are active. Hence, it is logical to connect the lookahead signal from the highest priority block to the lowest priority block, and the second highest to the second lowest, and so on. This approach is slightly different from the conventional lookahead schemes defined in the previous sections, where the lookahead signals are propagating in ascending order. The folding technique can be extended to multiple levels. The idea is like recursively folding a piece of paper. Figure 3.10 shows the logic design of a 128-bit MMR with 8-bit macro-blocks and 3 levels of folding.

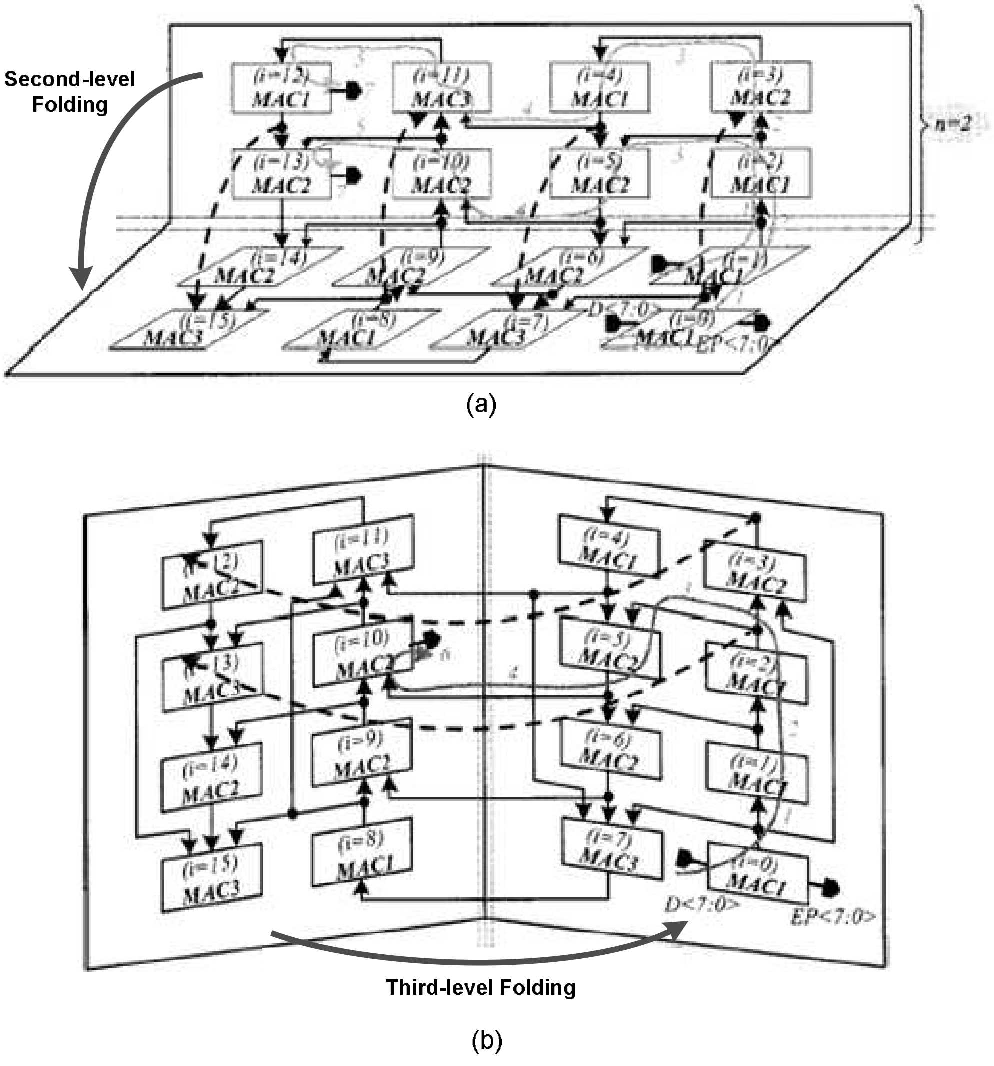


Figure 3.10: A 128-bit MMR with 8-bit Macro-blocks and 3-Level Folding (adapted from [17])

Similar to the progressive lookahead scheme, the multi-level folding technique is also impractical for integration with other blocks in TCAM. Although Huang in [17] reported significant speed improvement with silicon results, the numbers are extremely misleading. In his design, (i) the MMR cells are placed in "folded and circular topology", and (ii) the MMR is completely isolated with no interaction with other blocks on his test chip. In reality, the MMR cells must be laid out in a single-column fashion, for perfect pitch-matching with MLSAs and Match Address Encoder. This completes the review of the optimization techniques for MMR. In the next section, we will start looking into the CMOS circuit realizations.

3.4 Concepts of Cell-based MMRs

Previously in Section 3.2 and 3.3, we have explained the drawbacks of a static logic-based MMR. They are bulky and irregular in shape. Likewise, Domino logic-based MMRs, as proposed in [14], exhibits the same pitfalls. They do not meet the fine-pitch requirements in TCAMs.

3.4.1 Pass Transistor as a Switch

In order to offer friendly pitch-matching to the TCAM array, the preference is to design the MMR in a "cell-based" architecture. This is analogous to the memory core in TCAM, SRAM, DRAM, or Flash. However, in our case, the cells are tiled in one dimension only. In the ideal case, we want a cell that can be replicated as many times as required, and has no significant performance degradation even when N is large.

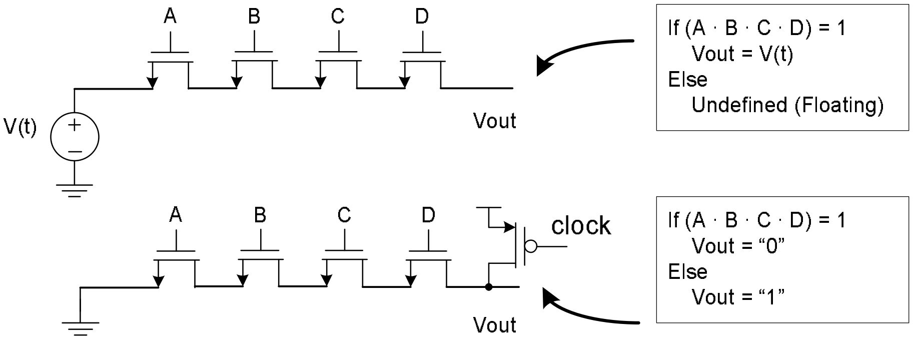


Figure 3.11: Using Pass Transistors as Switches

Figure 3.11 shows a simple NMOS pass transistor chain. The output voltage is a function of V(t) and the gate voltages A, B, C, and D. In order to avoid a floating output when $(A \cdot B \cdot C \cdot D) = 0$, a PMOS is present to pre-charge the output to "1". The output value remains at "1" unless $(A \cdot B \cdot C \cdot D) = 1$. This pass transistor chain can be employed in the design of MMR. The concept is to connect the MMR inputs (In_i) to gate of the MOS transistors. Each intermediate node of the chain can be a MMR output (R_i). This method can realize the cell-based implementation, such that each cell contains a pass-transistor for passing a signal.

Note that the pass transistor chain can be modeled by a simple RC network, as shown in Figure 3.12. Assume that V(t) in the diagram is the highest priority bit in the MMR, and the end of the chain V_N is the lowest priority bit. Each MOS transistor is modeled as a resistor, and the junction capacitance and wire parasitic capacitance are lumped into a simple capacitor C.

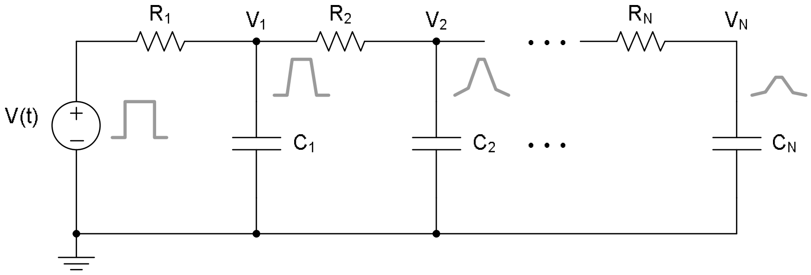


Figure 3.12: Distributed RC Ladder as a Model for a Pass Transistor Chain

An estimate of the worst-case time constant for such RC network is given by (3.3) [16].

$$\tau = RC \times \left(\frac{N^2}{2}\right) \tag{3.3}$$

Equation (3.3) suggests that the performance of the MMR would be limited by the speed of the pass-transistor chain when N is large. Hence, multi-level lookahead techniques are still required in this cell-based approach. For instance, a 256-bit cell-based MMR can be divided into 16 macro blocks with one level of lookahead. Each macro block has 16 pass-transistors in series.

3.4.2 Inhibit Chain vs. Match Token

In general, a cell-based MMR can be designed based on either (i) an "Inhibit Chain" method, or (ii) a "Match Token" method. The concepts are illustrated in Figure 3.13.

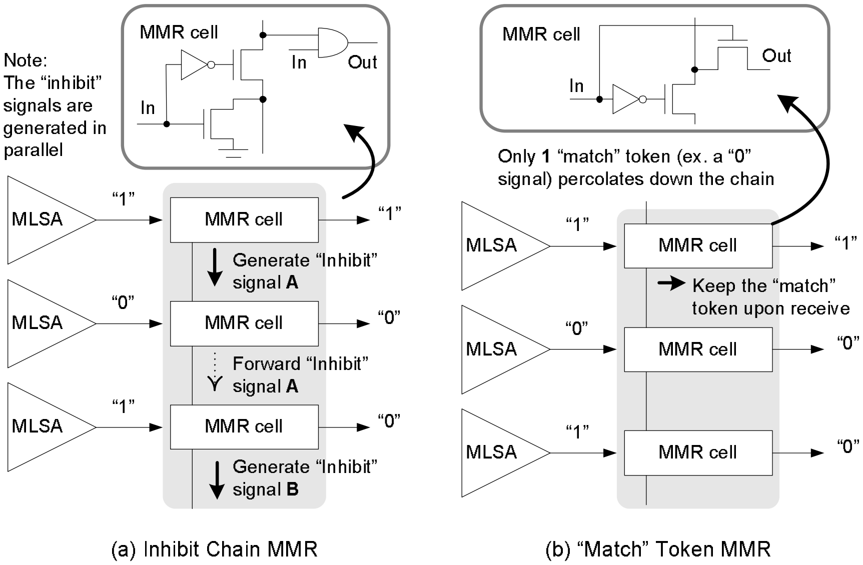

MMR cell
In
Out
In
Note:
The "inhibit" signals are generated in parallel
MLSA
"1"
MMR cell
"1"
Generate "Inhibit" signal **A**
MLSA
"0"
MMR cell
"0"
Forward "Inhibit" signal **A**
MLSA
"1"
MMR cell
"0"
Generate "Inhibit" signal **B**
(a) Inhibit Chain MMR
MMR cell
In
Out
Only **1** "match" token (ex. a "0" signal) percolates down the chain
MLSA
"1"
MMR cell
"1"
Keep the "match" token upon receive
MLSA
"0"
MMR cell
"0"
MLSA
"1"
MMR cell
"0"
(b) "Match" Token MMR

Figure 3.13: Inhibit Chain vs. "Match" Token based MMR (adapted from [5])

Inhibit-based Method

If an input bit is signaling a "match", the MMR cell assumes that it is already the highest priority match by setting the corresponding output bit to a "1". At the same time, it generates an "inhibit" signal. This "inhibit" signal is percolated down the pass-transistor chain to reset all the lower priority output bits to a "0". The output bit that survives until the end of the evaluation process represents the highest priority match.

The worst-case delay is the time to pass the inhibit signal from the highest priority cell to the lowest priority cell. This scheme is fast but the "broadcast" property is very energy-consuming, due to the high switching activities at the internal nodes and the output nodes. We will study some prior arts of Inhibit-based MMR in Section 4.1.

Token-based Method

Unlike the “Inhibit” method, the “Match Token” method does not suggest to raise the MMR cell output to a “1” right after the input bit is signaling a “match”. There is a global signal (a Match Token) percolating down the pass transistor chain from the highest priority bit to the lowest priority bit. If an input bit is signaling a “match”, the MMR cell keeps the “token” upon its arrival. Otherwise, it will forward the token to the lower priority bit. The first bit that receives the token represents the highest priority match.

The worst-case delay is the time to pass the token from the highest priority cell to the lowest priority cell. This delay is identical to the “Inhibit” method. However, it is much more power efficient due to low switching activities at the internal nodes and at the output nodes. We will study the circuits of Token-based MMR in Section 4.2. and a novel 12T Token-based MMR in Section 4.3.

Chapter 4

MMR Cell Design and Analysis

The main focus of this chapter is to explore the circuit techniques for designing a MMR cell for low-power and high-density TCAM applications.

4.1 Inhibit-based MMR Cell Designs

There were many different Inhibit-based MMR designs proposed over the past 20 years in numerous major journals, conference proceedings, and patent documents. They include [11], [16], [18], [19], and [20]. However, many of them were based on similar circuit principles. The claims in these proposed schemes differ only in one of the following.

- Using a V_{ss} or a V_{dd} as the "Inhibit" signal
- Using a NOR to replace a NAND as the output driver
- Whether the input is active-high or active-low

For completeness and review purposes, several inhibit-based MMR circuits are presented here in brief. Most of the circuit diagrams in the original references were illustrated in a complicated way with poor readability. The circuit diagrams in the following sections are re-drawn and simplified to emphasize the key points.

4.1.1 A 11T Cell with TG for Inhibit Signal Propagation

Figure 4.1(a) shows an inhibit-based MMR cell proposed by Bergh in [20]. Similar designs were also proposed in [16] and [19]. The cell consists of 11 MOS transistors, with active-low input, and active-high output. During pre-charge, all MMR inputs are inactive (at logic "1"). Hence, all transmission gates along the chain are "ON", and the intermediate nodes are discharged to V_{ss}. At evaluation, as shown in Figure 4.1(b), if an input is signaling a "match", In_i is pulled to a "0". This switches off the transmission gate, and sets the corresponding MMR output R_i to a "1" if the block is already enabled. At the same time, this input signal turns on the PMOS transistor, which charges the lower priority nodes to V_{dd}. In other words, the PMOS is generating an "inhibit" signal to invalidate all lower priority matches. The Block Enabled (BE) signal is also active-low. It is used to facilitate multi-level lookahead. If there is a match in a higher priority block, the current BE signal is held at inactive state. Otherwise, it will become active to raise the output of the highest priority bit in the current block to "1".

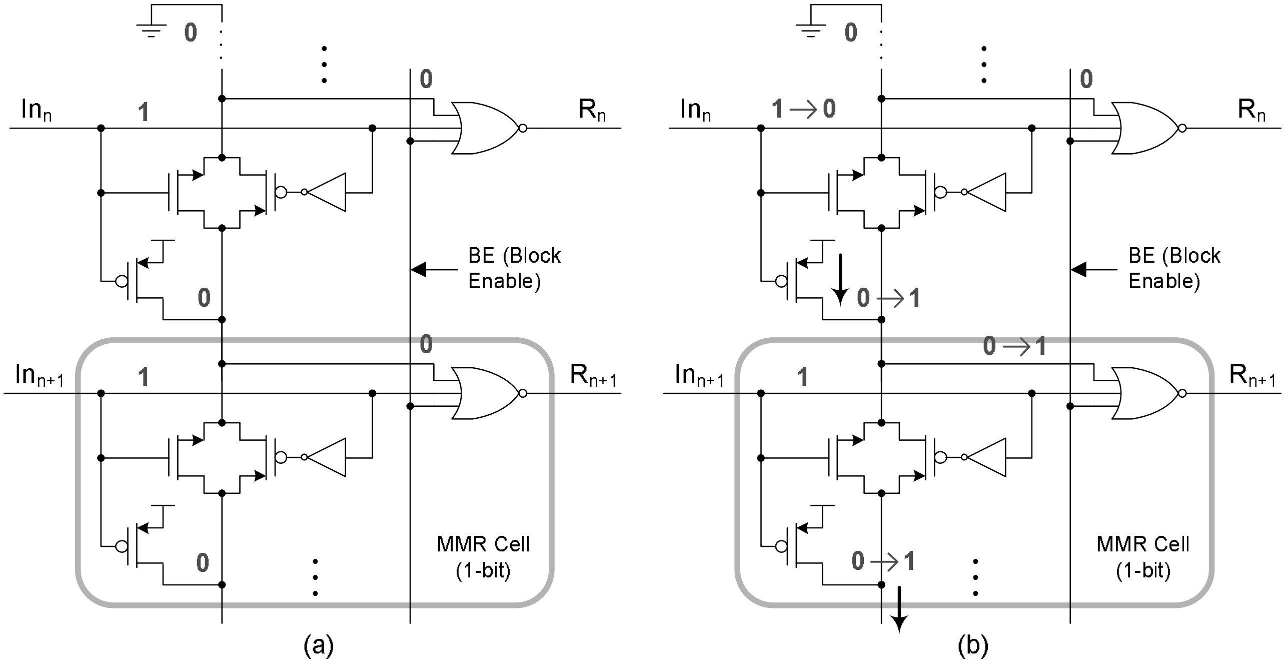


Figure 4.1: A 11T Cell with TG for Inhibit Signal Propagation (a) Pre-charge (b) Evaluation

The Transmission-Gate (TG) chain is offering relatively good noise margins at the internal nodes. However, there are a number of shortcomings in this design.

1. The transmission gate requires complementary enable signals

2. The critical delay depends on how fast the PMOS can charge all internal nodes to V_{dd}. Unless the PMOS is huge, the delay is much longer in compared to an NMOS pull-down.

3. The 3-input NOR gates are causing a huge capacitive load on the Block Enable (BE) signal. This imposes a limit to the maximum number of bits per macro-block.

4.1.2 A 9T Cell with NMOS for Inhibit Signal Propagation

Figure 4.2(a) shows an inhibit-based MMR cell proposed by Delgado-Frias in [11]. It consists of only 9 MOS transistors, with active-high input, and active-high output. This design employs NMOS pass-transistors to replace the transmission gates in the former example. During pre-charge, all MMR inputs are inactive. Hence, the NMOS pass-transistors are "ON", and the intermediate nodes are charged to V_{dd}. At evaluation, as shown in Figure 4.2(b), a "0 → 1" transition at the input closes the NMOS pass transistor, and sets the corresponding MMR output to a "1". An inhibit signal is generated by the NMOS pull-down transistor to invalidate all lower priority matches.

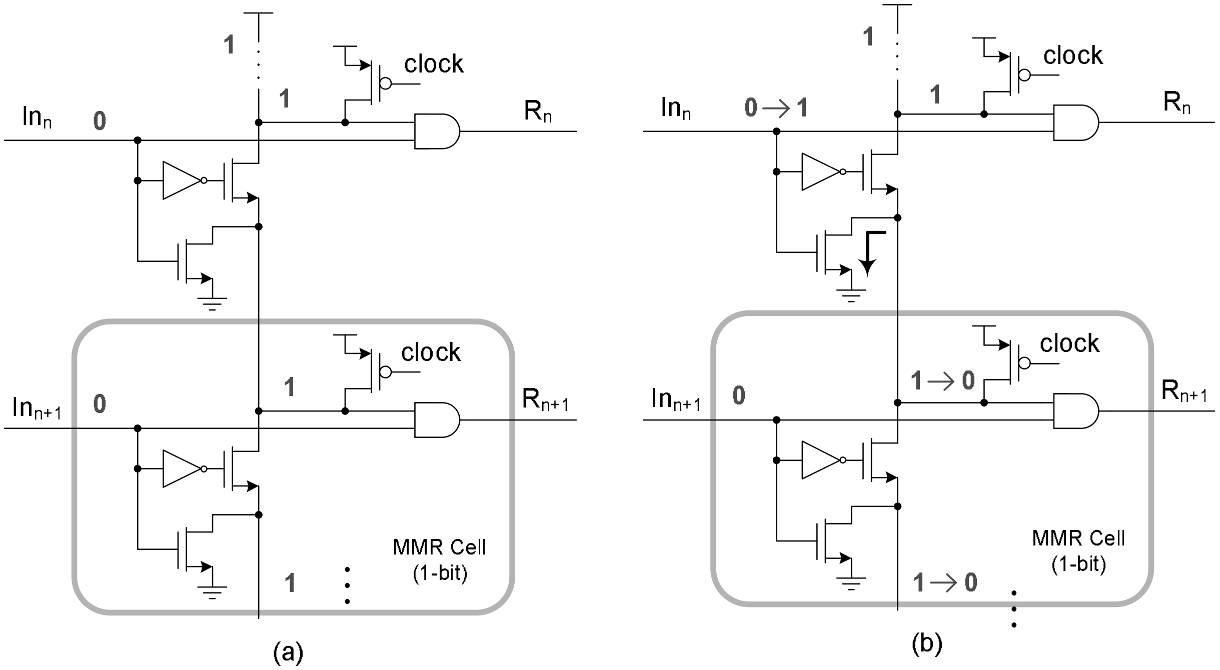


Figure 4.2: A 9T Cell with NMOS for Inhibit Signal Propagation (a) Pre-charge (b) Evaluation

The operation of this MMR is actually a dual of the former example. In the circuit-level, however, there are two key improvements. First, the design employs NMOS pass transistors for evaluation. Second, the transistor that generates the "inhibit" signal is an NMOS, which offers better driving capability in compared to a PMOS [15]. Another idea, proposed by Delgado-Frias, is to connect the lookahead signals to the internal nodes instead of connecting them to the output drivers, as depicted in Figure 4.3.

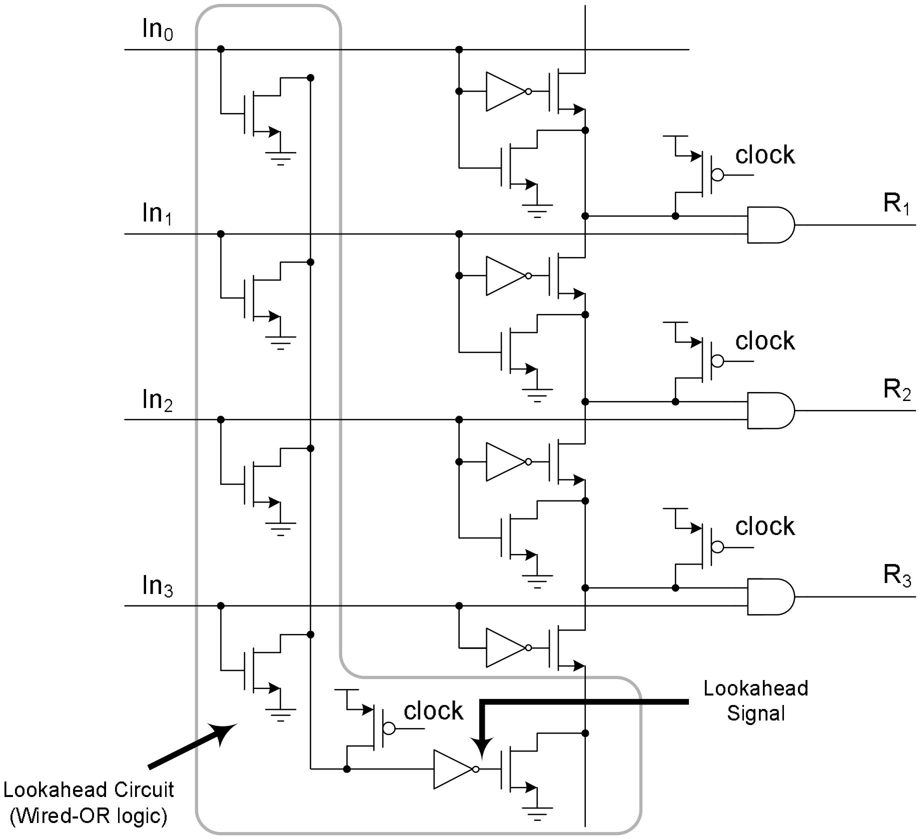


Figure 4.3: Embedded Lookahead Structure

This lookahead, or bypassing, structure is simple. However, such design cannot be scaled to handle a large number of inputs without the multi-level block enabling. In addition, the MMR outputs must be latched to avoid the transient during evaluation. Like many other inhibit-based MMRs, this design is consuming high power because almost all internal nodes are toggling even only one or two inputs are active.

4.1.3 A 14T Cell with Low-V_t Pass Transistor

Figure 4.4 shows a MMR proposed by Miwa in [18]. It has been employed in the design of a 1 Mb non-volatile CAM based on the Flash memory technology. This MMR cell is nearly identical to the one previously shown in Figure 4.2(a), except the slight modification in the output driver, and the employment of low threshold voltage (low-V_t) NMOS transistor along the pass transistor chain.

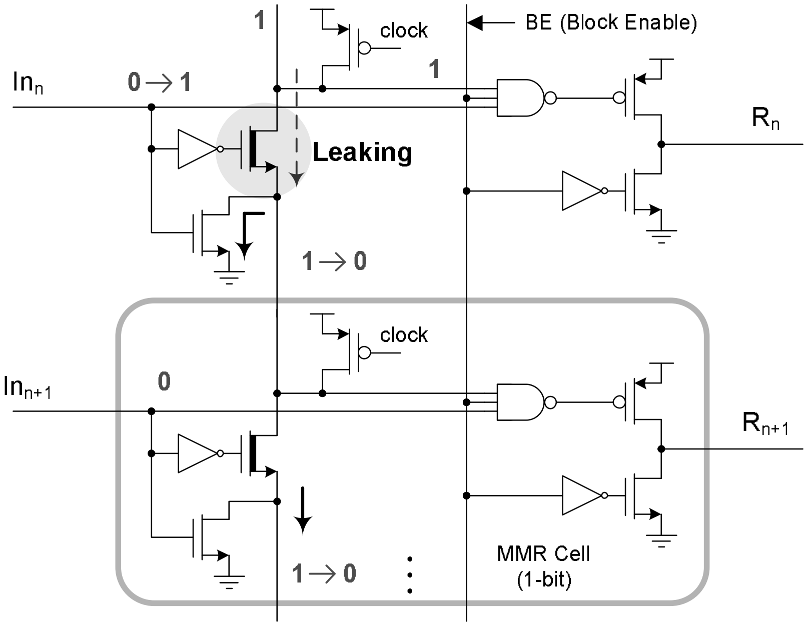


Figure 4.4: A 14T Cell with Low-V_t Pass Transistor

Based on Equation (3.3) previously studied in Section 3.4.1, the worst-cast delay of a distributed pass transistor network is proportional to the NMOS channel resistance. The channel resistance of an NMOS transistor is non-linear, however it can be estimated using Equation 4.1 [15].

$$r_{on} = \frac{1}{\partial I_d / \partial V_{ds}} \approx \frac{L}{K'W(V_{gs} - V_t - V_{ds})} \tag{4.1}$$

This equation shows that the channel resistance is inversely proportional to (V_{gs} - V_t - V_{ds}). Hence, a low-V_t NMOS can help to reduce the worst-case delay in the pass-transistor chain (for both "Inhibit" method and the "Match Token" method). As a consequence, the low-V_t property also implies that the transistor is extremely “leaky”. With a wide range of process variation, the leaking can be large enough to cause a false discharge on the highest priority bit. This can lead to a situation where the supposedly resolved highest priority match never appear at the MMR output. Although adding relatively large PMOS keepers to the intermediate nodes can help fighting the leakages, this strategy is not reliable because the leakage of a low-V_t device is more sensitive to process variations. Furthermore, large keeper transistors have negative impacts on the performance of the pass-transistor chain.

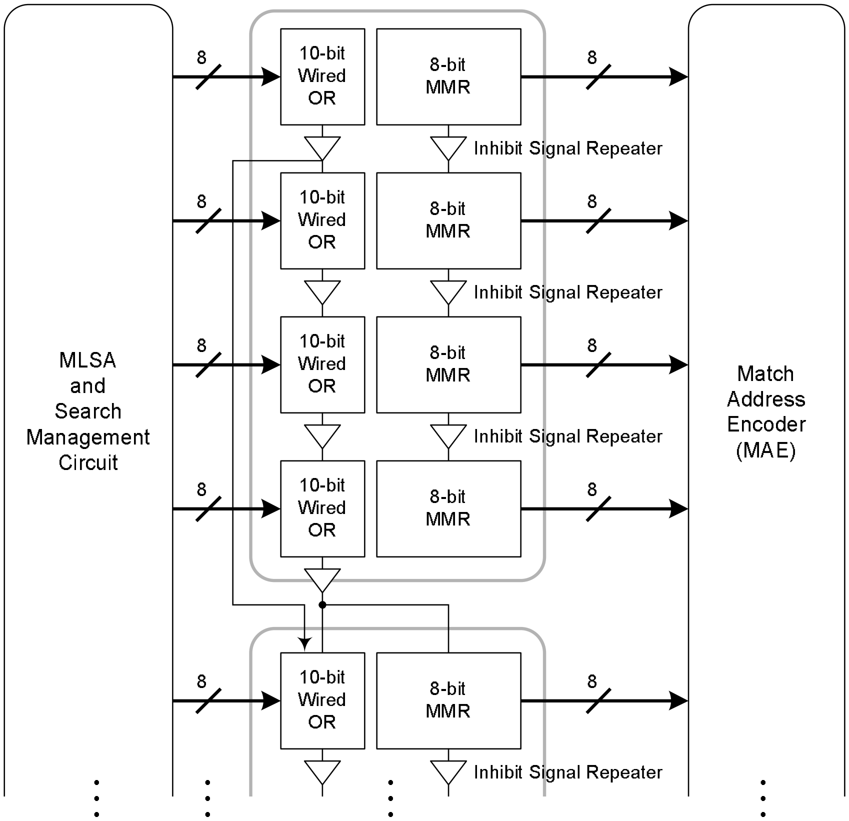

MLSA and Search Management Circuit
8
10-bit Wired OR
8-bit MMR
Inhibit Signal Repeater
Match Address Encoder (MAE)

Figure 4.5: Architecture of a 256-bit MMR with Low-V_t Inhibit Chain and Lookahead

Figure 4.5 shows the architecture of a 256-bit MMR with 2-level lookahead and low-V_t “Inhibit” chains. The inhibit signals in the first level are amplified to full-swing for a distance over every 8 bits. Bypassing paths and lookahead paths are also present to speed up the second level inhibit signal propagation. This architecture is similar to the modern MMRs used in Ternary CAMs.

4.2 Token-based MMR Cell Designs

As previously introduced in Section 3.4.2, a "Token" based MMR does not raise the MMR output to a "1" right after the input bit is signaling a "match". There is a global signal (a Match Token) percolating down the pass transistor chain from the highest priority bit to the lowest priority bit. If an input bit is signaling a "match", the MMR cell keeps the "token". Otherwise, it will forward the token to the lower priority bit. A MMR output is switching to a "1" only if it is holding the "Match Token". The first bit that receives the token represents the highest priority match.

4.2.1 A 12T Cell based on Token-Passing

Figure 4.6 shows a 8-bit MMR macro-block with wired-OR lookahead. Each MMR cell consists of 12 MOS transistors. This circuit was proposed by Foss in [21].

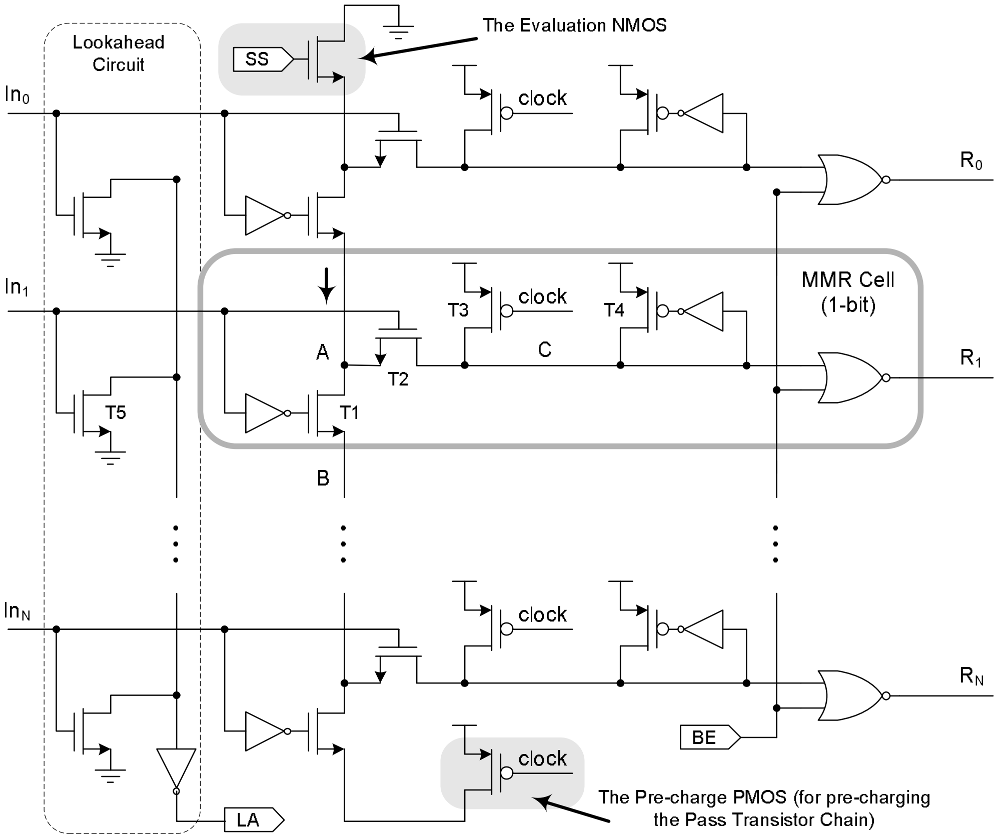


Figure 4.6: A 8-bit MMR Macro-block based on Match-Token Concepts

The circuit was designed based on the "Match-Token" concepts, with active-high inputs and outputs. Notice that the MMR cell does not generate any inhibit signal to invalidate the lower priority cells. It is just a passive element to either receive or forward the "Match Token".

During the pre-charge phase, both the input signals (In_i) and the clock signal are at "0" state. The pass transistor T1 is turned on and T2 is switched off. This isolates the internal transistors in the MMR cell (T3, T4, the keeper, and the NOR gate) from the pass-transistor chain. The pre-charging at node C resets the output node (R_i) to "0". Note that the intermediate nodes of the pass transistor chain (ex. node A and B) are being charged to $V_{dd} - V_{tn}$ instead of V_{dd}. This is because an NMOS transistor can only transmit a weak "1" [15]. In addition, the $V_{dd} - V_{tn}$ value is only true if the pre-charge period is sufficiently long (at t→ ∞). In practice, the intermediate node voltages are always slightly below $V_{dd} - V_{tn}$. To clarify the description, a timing diagram is shown in Figure 4.7.

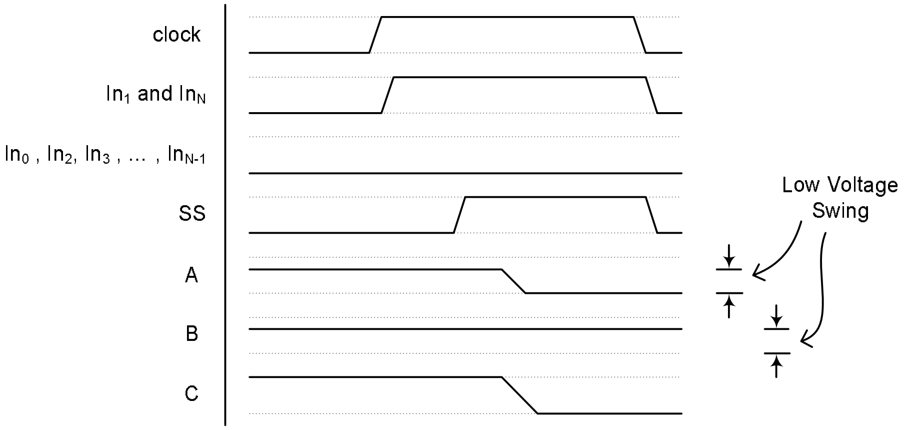


Figure 4.7: Timing Diagram for the Token-based Scheme by [21]

Assume that there are 2 matches in the TCAM array, they are located in word 1 and word N. Hence, In_1 and In_N are raised to V_{dd} at the on-set of the evaluation phase. The rest of the input bits remain at "0". The switching at In_0 turns off T1 and switches on T2. After a certain delay that guarantees the stability of the pass-transistor chain, the "SS" signal (Strobe Signal) is switched to "1". This allows the discharging of the entire pass-transistor chain up the highest priority bit. Such discharging is analogous to percolating a ground signal down the pass transistor network, so the name of "Match Token". Notice that node B and all the lower priority bits will be isolated from the V_{ss} signal and never receives the Match Token. Upon the arrival of the Block Enable (BE) signal, the output R_1 will be switched to a "1" to indicate that word 1 is the highest priority match. With careful observation, this design is actually a modified "Compound Multiple-Output Domino Logic" circuit. The only difference is the introduction of the Strobe signal to gate the evaluation NMOS, instead of gated by the clock signal. Detail description of "Compound Multiple-Output Domino Logic" is not given here, interested readers can look into [22] for more information.

Figure 4.8(a) shows a 64-bit MMR. The first-level is divided into eight macro-blocks, where each macro-block is the circuit previously shown in Figure 4.6. The lookahead signals are then processed by a second-level MMR, to determine the block that contains the highest priority match. The resolved second-level signals are therefore the "Block Enable" (BE) signals for the first-level MMRs. In order to layout both levels of MMRs into one column, the MMR cells in the second-level are distributed between the first-level blocks. This is illustrated in Figure 4.8(b).

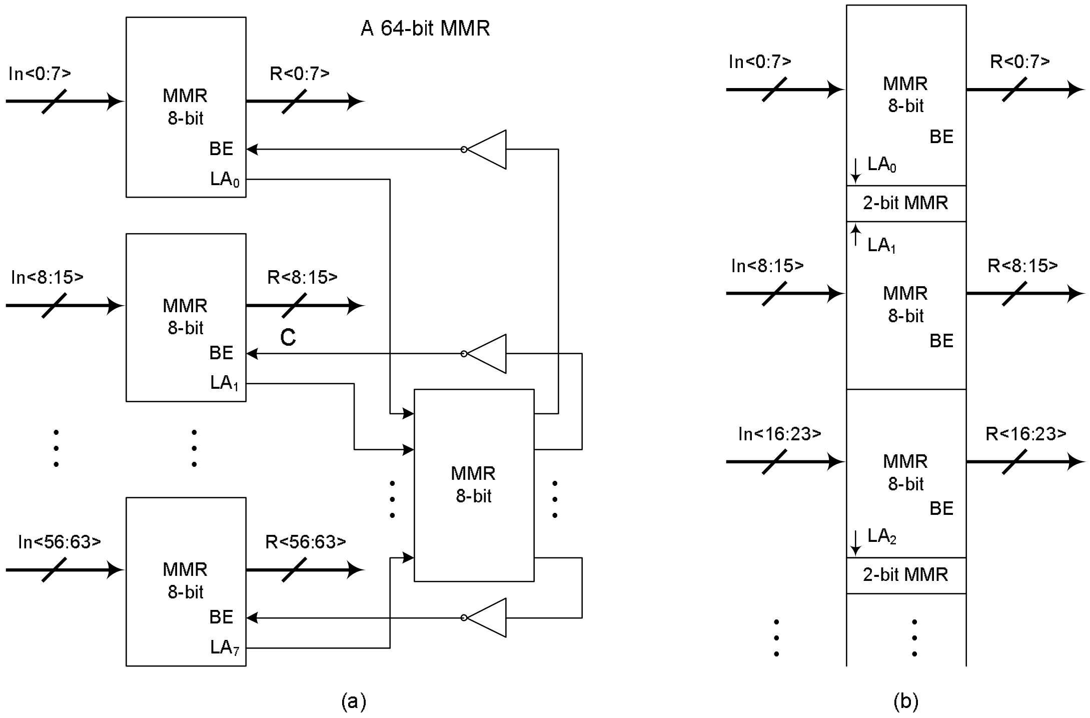


Figure 4.8: A 64-bit Token-based MMR using the Cell Proposed by [21]

The 12T MMR cell is a good design in general. However, it is definitely not the best in its class. There are a lot of rooms to grow and improve. The shortcomings in this design are listed in the following. They are good guidelines to make the circuit more suitable for low-power TCAMs.

1. The output driver of this MMR cell is a NOR-gate. Since the MMR output is active-high, the pull-up capability of the NOR gate directly influences the critical path delay. Notice that the Block Enable (BE) line is also part of the critical path, however it is connected to the NOR-gate of every cell in the macro-block. The total gate capacitance (C_g) due to these NOR-gates can be huge. The insertion of additional buffers at the MMR outputs does not mitigate the problem. Even these NOR-gates are in "minimum-size", the total C_g is still very large. This imposes a limitation on the maximum size of the macro-block. In other words, the NOR-gate in the MMR cell is limiting the scalability of this design.

2. Due to its "Domino Circuit" nature, this design creates a large load on the clock drivers. Even if none of the MMR inputs are active, the system is consuming power because the clock drivers are charging and discharging these pre-charge/evaluation MOS transistors every clock cycle (they are dummy loads in this case). The use of "clock gating" in the clock drivers will save power, but the re-buffering of the clock signal is adding more delay and skew to the circuit.

3. The synthesis of the "SS" signal is not given in [21]. If the NMOS evaluation transistor is activated every clock cycle, all internal nodes along the pass-transistor chain would be charged and discharged entirely. This unnecessary operation is wasting a lot of power. If this is the case, it is even more power consuming than the inhibit-based MMR circuits.

4. The PMOS keeper T4 is originally not in [21]. During the evaluation phase, if the input is a "0", T2 is off and node C is basically floating. If the input is a "1", T2 is on and node C is susceptible to any small noise on the pass-transistor chain. Hence, an inverter and T4 are added into the circuit for reliability.

In the next section, we will look at a novel MMR design. It is an improved implementation of this 12T Match-Token based design.

4.3 Design of a Novel MMR Cell

A novel 12T MMR cell based on the Match-Token concepts are disclosed in this section. There are five novel circuit ideas in this new design. For a quick preview, they have been labeled on Figure 4.9 and Figure 4.11.

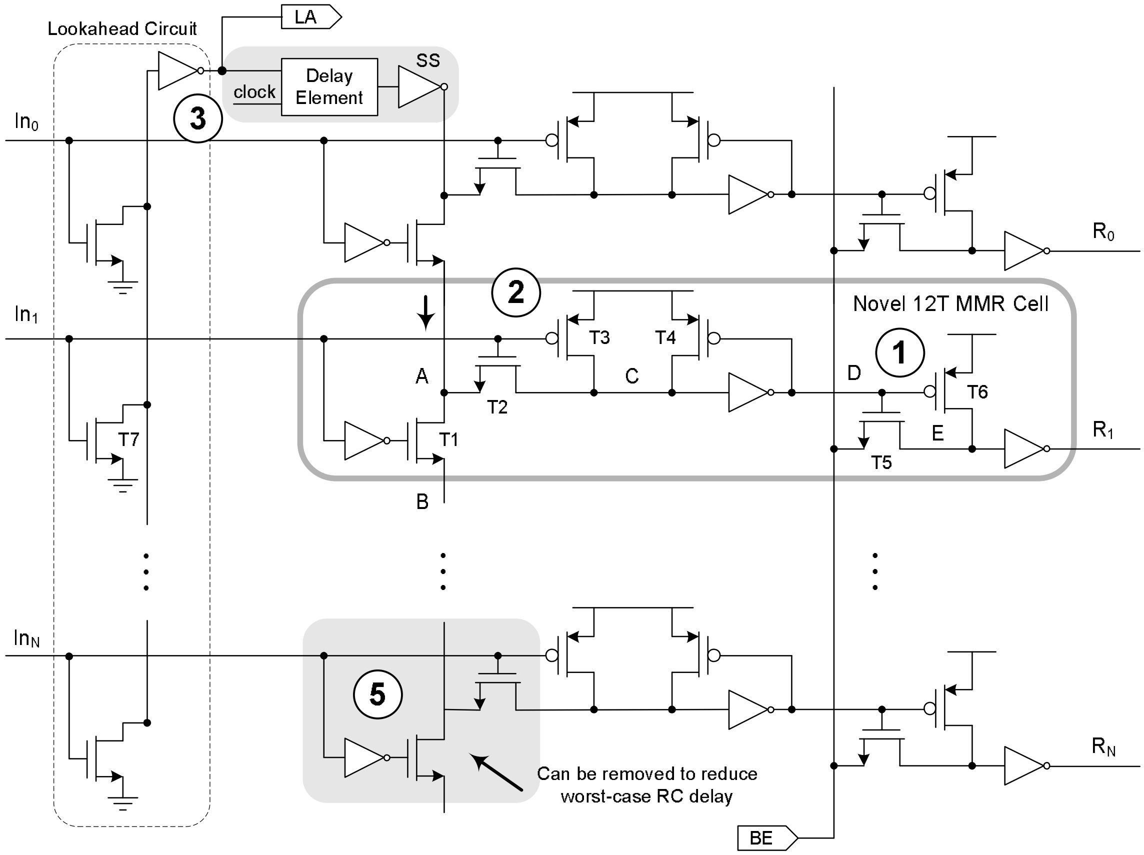


Figure 4.9: A 12T novel MMR cell in a 8-bit Macro-block

4.3.1 Timing and Circuit Operation

At the on-set of the pre-charge phase, the clock signal undergoes a transition of “1 → 0”. The pre-charging at node C relies on the “1 → 0” transition at the input bit (In_i). Once the input switches back to “0”, the pass transistor T1 is turned on and T2 is switched off. This isolates the internal transistors of the MMR cell (T3 - T6, and the two inverters) from the pass-transistor chain.

As a consequence, node C is charged to V_{dd} and node D to “0”, which in turn switches off T5 and pre-charge node E to V_{dd}. Note that the intermediate nodes of the pass transistor chain (ex. node A and B) are being charged to $V_{dd} - V_{tn}$ instead of V_{dd}. This is because an NMOS transistor can only transmit a weak “1” [15]. In addition, the $V_{dd} - V_{tn}$ value is only true if the pre-charge period is sufficiently long (at t$\rightarrow \infty$). In practice, the intermediate node voltages are always slightly below $V_{dd} - V_{tn}$. A timing diagram is shown in Figure 4.10 for visual interpretation.

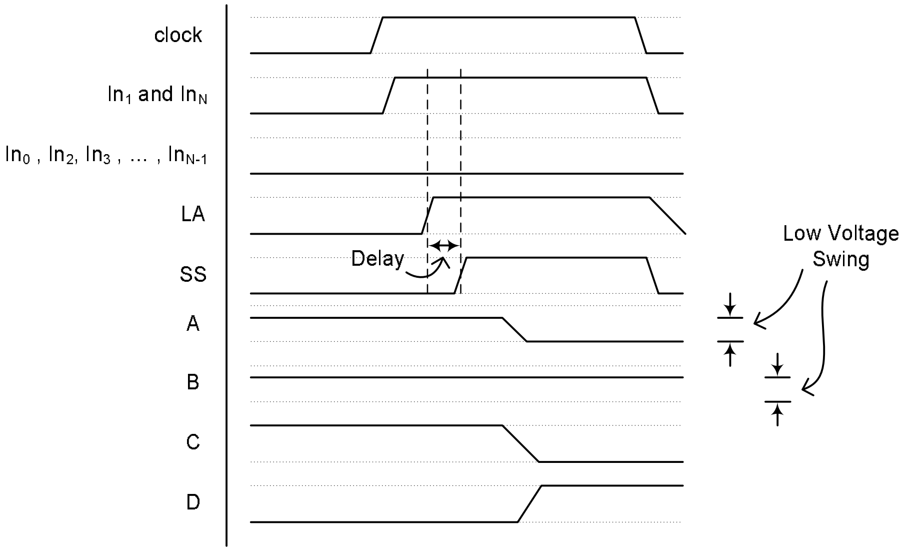


Figure 4.10: Timing Diagram for a Macro-block using the New Cells

Assume that there are 2 matches in the TCAM array, they are located in word 1 and word N. Hence, In_1 and In_N are raised to V_{dd} at the beginning of the evaluation phase. The rest of the input bits remain at “0”. The switching at In_0 turns off T1 and switches on T2. A wired-OR circuit is built into the macro-block for sensing if at least one match exists at the inputs. The output of this wired-OR gate is a lookahead signal for interfacing with the second-level MMR. This lookahead signal, denoted by LA, is applied to the input of a delay element for generating the “SS” (Strobe) signal. This delay is intentional, because the pass-transistor chain in the first-level is not the critical path of a multi-level MMR. The purpose of the delay element is to reduce as much capacitance as possible at the LA node.

Switching “SS” from “1 $\rightarrow$ 0” allows the discharging of the entire pass-transistor chain down to the highest priority “match” in the local macro-block. Such discharging is analogous to percolating a ground signal down the pass transistor network, so the name of “Match Token”. The internal nodes C and D of the highest priority cell will be inverted, so that T5 is “on” to connect the gate of the output inverter to the Block Enable (BE) line. Notice that node B and all the lower priority bits will be isolated from the V_{ss} signal and never receives the Match Token. Upon the arrival of the Block Enable (BE) signal, node E will be discharged to “0”, which in turn switches the output R_1 to 1 to indicate that word 1 is the highest priority match.

4.3.2 The Novelties in The Proposed Scheme

Item #1: A More Scalable Output Circuit

In order to minimize the capacitance on the Block Enable (BE) line, the static 2-input NAND/NOR output driver in the prior designs is replaced by a RAM-like circuit as shown in Figure 4.9. The circuit consists of 4 transistors: the CMOS inverter, T5, and T6. The reason for doing this is to hide the gate capacitance of the output drivers from the BE line. Notice that at most only one cell (the local highest priority “match”) is expecting the BE signal. The rest of the output drivers do not participate in the process. They are only capacitive loads on the critical path. With the proposed circuit, transistor T5 shields the internal gate capacitance and drain capacitance of the output drivers from the BE line. Only the local highest priority cell has its T5 conducting for receiving the BE signal. This is analogous to the writing process in the memory array.

Table 4.1 shows the total capacitance (excluding the inter-wire and parasitic capacitance) on the BE line for a 16-bit MMR macro-block. The “actual capacitance” in the table is based on sizing the MMR output drivers for 8fF output load at R_i. The values are simulation results using TSMC 0.18 μm CMOS model.

Output Driver Type	2-input NAND	2-input NOR	The Proposed Circuit
Symbolic Equation	16 $C_{g,NAND}$	16 $C_{g,NOR}$	18 C_d + 1 $C_{g,INV}$
Actual Capacitance	≈ 60 fF	≈ 80 fF	≈ 13 fF

Table 4.1: Total Capacitance on BE Line vs. MMR Output Driver Type

Notice that the size of a macro-block is limited by (i) the capacitance on the BE line, and (ii) the RC delay in the pass-transistor chain. The former limitation can be successfully tackled by the proposed output circuit. The later one will be discussed in Item #4 in the following. With these techniques, the macro-block size can be expanded from 8-bit to 16-bit, or even beyond.

Item #2: Data-Dependent Clocking

As previously studied in Section 4.2, the MMR cell proposed by Foss has a PMOS pre-charge transistor (T3) located at node C. The pre-charging of this node is only applicable if the cell is the highest priority "match" in the present cycle. Otherwise, there is no need to clock T3 for pre-charging node C. Node C is already at V_{dd} in the usual case. The presence of this clocked PMOS transistor in every MMR cell is putting a huge capacitive load on the clock driver.

In order to address this problem, we can employ a "pseudo-static" strategy to charge node C based on the input data. This is highlighted in Figure 4.9. If In_i is a "0", the PMOS transistor T3 remains "on" even during the evaluation phase. This is not a concern because node C is isolated from the pass-transistor chain. In fact, this conducting PMOS (T3) helps to fight the switching noise during evaluation phase. On the other hand, if In_i is a "1", the input bit enables T2 and disables T3, which is exactly identical to the operation of the clocked scheme. Such data-dependent clocking strategy is very effective in reducing the clock power.

Item #3: Conditional Generation of "Match Token"

In the prior token-based scheme, a "Match Token" is generated and percolates down the pass-transistor chain every clock cycle. This action is regardless to whether a match exists or does not exist in the macro-block. A better and smarter approach is to gate the generation of this token by the output of the wired-OR gate (the LA signal), as shown in Figure 4.9. Notice that the LA signal is applied to the input of a delay element for generating the "SS" (Strobe) signal. This delay is intentional, because the pass-transistor chain in the first-level is not the critical path of a multi-level MMR. The purpose of the delay element is to reduce as much capacitance as possible at the LA node.

Item #4: Embedded Bypassing Paths

The RC delay of the pass-transistor chain is the second barrier that limits the size of the macro-block (level 1 MMR) to at most 8-bit, as reported in [21] and [18]. The concerning factor here is not the speed but the functionality. This is because a long chain of NMOS in series causes the discharging current at node C of the cell even weaker than the charging current delivered by the PMOS keeper (T3). One solution is to introduce bypassing paths internally to reduce the number of NMOS in series in the worst-case. A 16-bit MMR macro-block can be achieved by dividing the inputs into 4 mini-blocks, each contains 4 bits of inputs. Figure 4.11 shows the proposed bypassing architecture. Although the proposed circuit looks simple, it is not that simple for the memory environment. Any additional datapath logic can destroy the regularity of the MMR structure, which makes the circuit not pitch-matching to the MLSAs. The solution of this problem will be revealed in the discussion of Item #5 below.

Item #5: Removal of the Redundant Pass Transistors

Based on Item #3, the match token is generated only if there is at least one match in the macro-block. This implies that there must be at least one "receivers" along the pass-transistor chain. When the lowest priority cell receives the token, this cell must be the highest priority "match" in the macro-block, unless the MMR does not function correctly. If this is true, the transistors T1 and T2 and the inverter driving T1, at the lowest priority bit are all redundant. It is obvious that they can be removed to improve the worst-case signal strength.

The absence of these transistors requires a special MMR cell dedicated to only the last bit of a macro-block. In the first sense, this violates the regularity of the memory structure. However, their absence can create silicon area to place the control circuity for realizing Item #3 and Item #4. Those two items were not proposed in the past because other researchers might have problems to find silicon spaces to fit these logics into the TCAM environment. With this innovation, we can have embedded bypassing paths, and conditional request of "Match Token" features for great power savings. These circuit techniques make low-power MMR for high-density TCAM achievable.

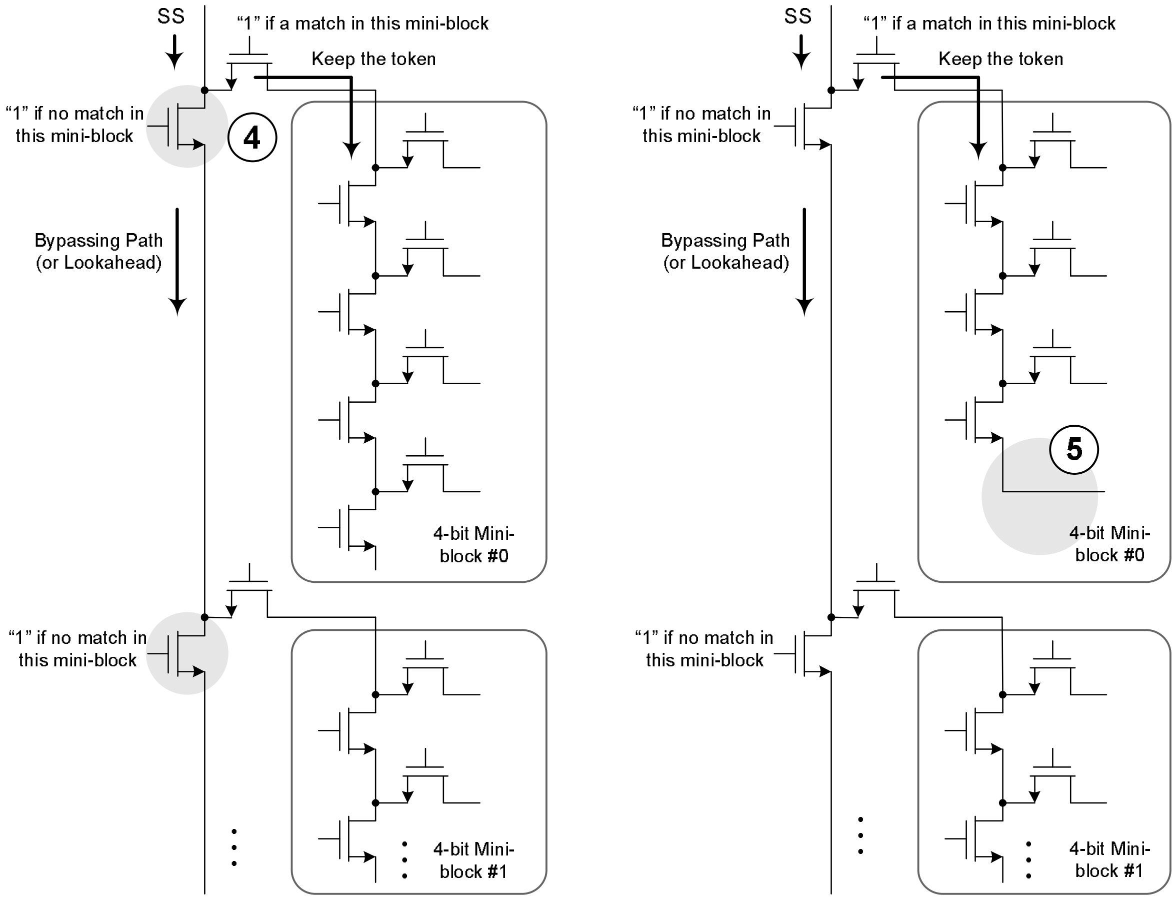

SS
"1" if a match in this mini-block
Keep the token
"1" if no match in this mini-block
4
Bypassing Path (or Lookahead)
4-bit Mini-block #0
"1" if no match in this mini-block
4-bit Mini-block #1
SS
"1" if a match in this mini-block
Keep the token
"1" if no match in this mini-block
Bypassing Path (or Lookahead)
5
4-bit Mini-block #0
"1" if no match in this mini-block
4-bit Mini-block #1

Figure 4.11: A 16-bit MMR Macro-block with Novel Bypassing Architecture

4.3.3 Parametric Analysis and Simulation Results

The proposed MMR circuits have been simulated using Spectre and HSPICE with the BSIM3 model for TSMC 0.18 μm CMOS technology. Inter-wire and parasitic capacitances are extracted using DivaEXT from Cadence. The delay and energy consumption of the novel scheme is compared against two previous works. They are the token-based scheme by Foss in [21], and the inhibit-based scheme with multi-level folding proposed by Huang in [17]. For fair comparison, all three MMRs are 64-bit wide and simulated using the same testbench. They are all divided into 2 hierarchical levels, with 8-bit macro-block in the first level, and another 8-bit in the second level for resolving the highest priority block. Although the novel scheme is scalable to achieve 256-bit resolution, the same is not true for the other two. Hence, 64-bit is chosen as the right size for a fair comparison in this context.

Figure 4.12 shows an "Energy vs. Delay" curve for the two token-based schemes. The data points are obtained by varying the size of the transistors along the critical paths in each design. Both circuits have a minimum Energy-Delay-Product (EDP) when the transistors are sized to achieve a worst-case delay of 610 ps. For the same worst-case delay, the novel circuit consumes only $(0.87pJ/1.72pJ) \approx 50.58\%$ of the energy required by the old scheme.

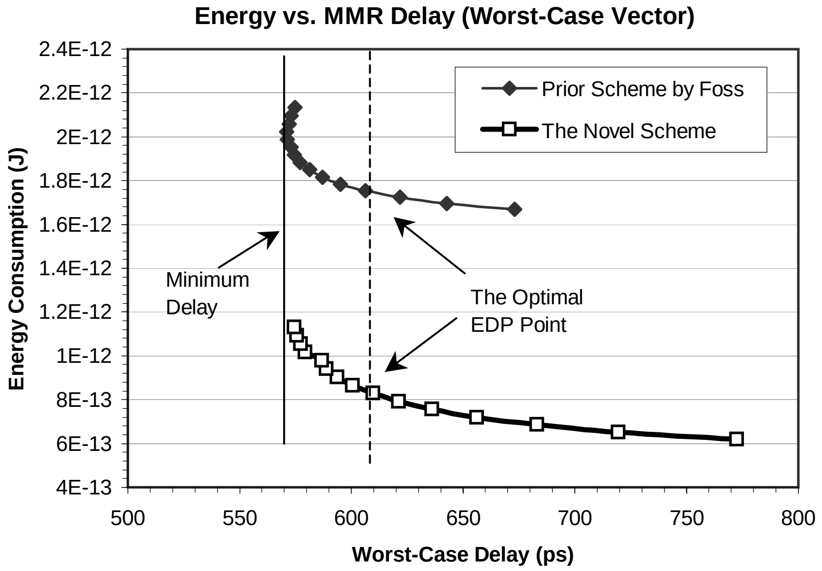


Figure 4.12: Energy-Delay Curve for the Two Token-based Schemes

The large savings in energy consumption is mainly due to (i) the reduction of BE line capacitance (Item #1 in Section 4.3.2), and (ii) a smaller short-circuit current at internal node during evaluation (Item #5 in Section 4.3.2). In addition, both schemes have a minimum delay of 572 ps for 64-bit resolution (at 27°C, typical process corner).

Figure 4.13 shows the "Energy vs. Delay" curves for all three schemes, with and without the consideration of clock power. The curves in grey color represent the total energy consumption including the energy for MMR operation, clock driver, and input drivers etc. The reduction in energy is even more dramatic when the clock energy is taken into consideration. The new scheme saves clock energy by removing the clock-dependent pre-charge transistors (Item #2 in Section 4.3.2).

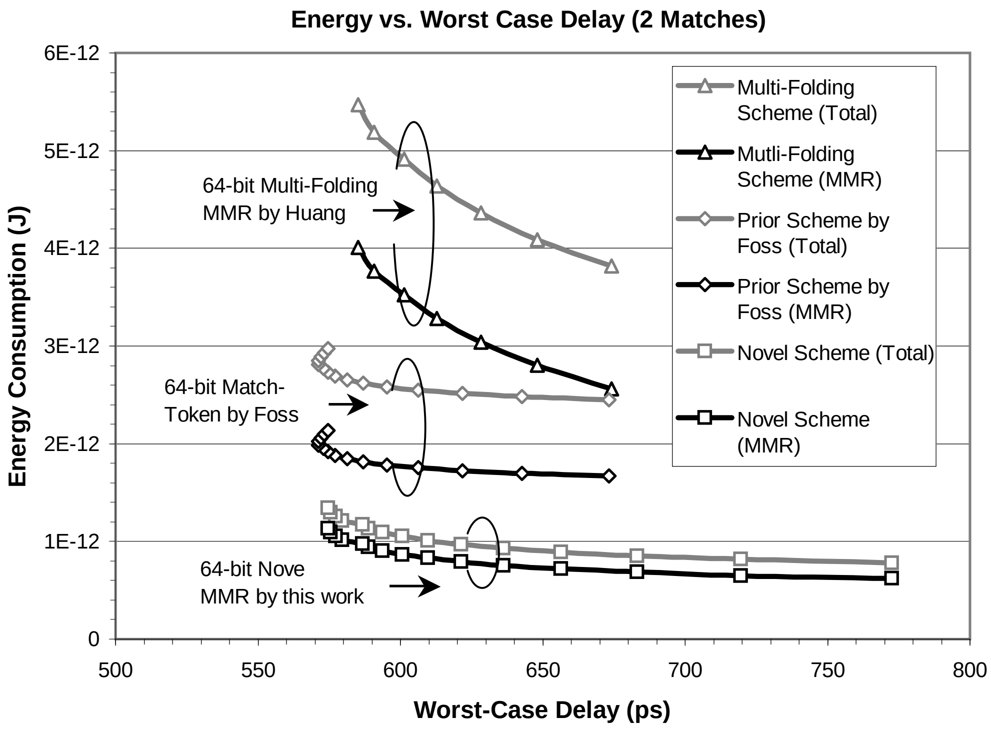


Figure 4.13: Energy-Delay Curve for All Three Schemes with and without Clock Power

It is clear that the inhibit-based Multi-Folding scheme is extremely power hungry, although it can be sized to outperform the novel scheme. For low power Ternary CAMs, the novel MMR is energy-efficient, and yet able to deliver the right speed for high-performance multiple match resolution.

4.3.4 Post-Layout Simulation Results

A 256-bit MMR based on the novel circuit techniques have been designed and fabricated in TSMC 0.18 μm CMOS technology. The 256-bit MMR is realized in two levels. The first level is divided into 16 macro-blocks. Each macro-block has 16-bit resolution. The layout plot is shown in Figure 4.14. This 256-bit MMR occupies 15 μm x 1100 μm $\approx$ 16.5nm^2. It is implemented using two 128-bit MMRs interleaved together due to limited silicon area.

The chip has been simulated using Spectre with external bondwire parasitics and package parasitics. Table 4.2 shows the expected worst-case results in the physical measurement.

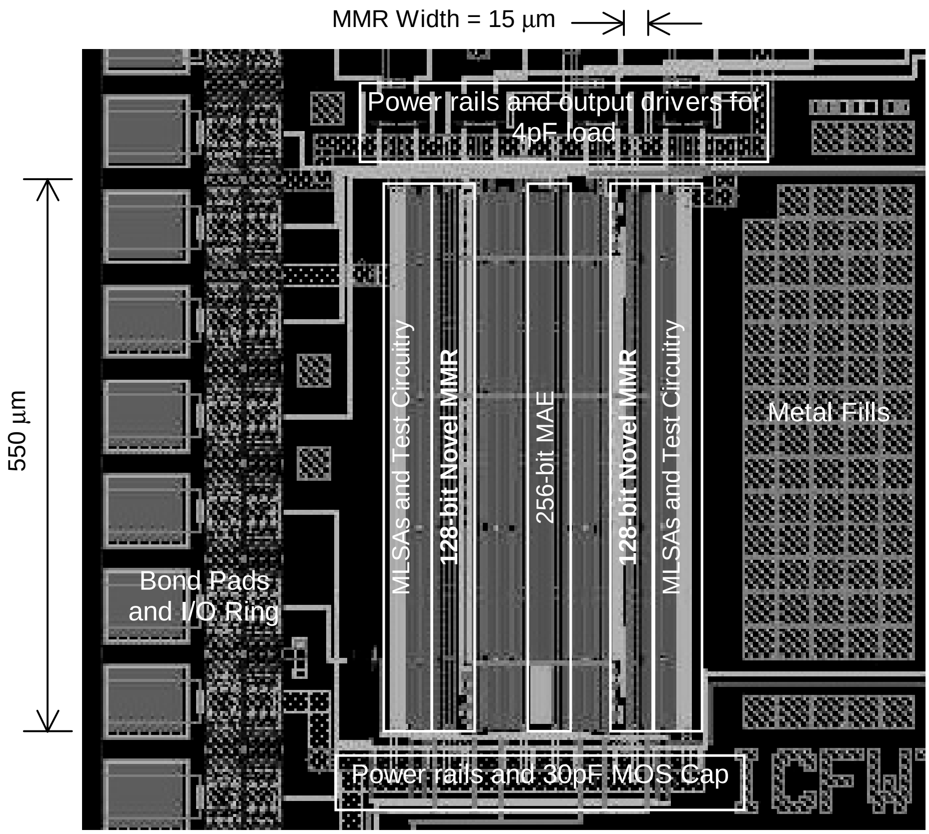


Figure 4.14: Layout Plot of a 256-bit MMR based on the Novel Schemes

	Delay (in ps)	Energy Consumption (Freq = 125 MHz)
Worst-Case 1 Match	894.4	39.8 pJ / cycle
Worst-Case 2 Matches	863.2	51.1 pJ / cycle

Table 4.2: Post-Layout Simulation Results of a Novel 256-bit MMR

Chapter 5

Match Address Encoding

As previously described in the Chapter 3, at most only one input of the Match Address Encoder (MAE) would be "active" after the multiple match resolution. This active MAE input, if any, represents the location of the best match of a Ternary CAM (TCAM) search. The next step, as illustrated in the flow diagram in Figure 5.1, is to encode this match location into binary format. This binary address is used to retrieve external data in off-chip SRAM.

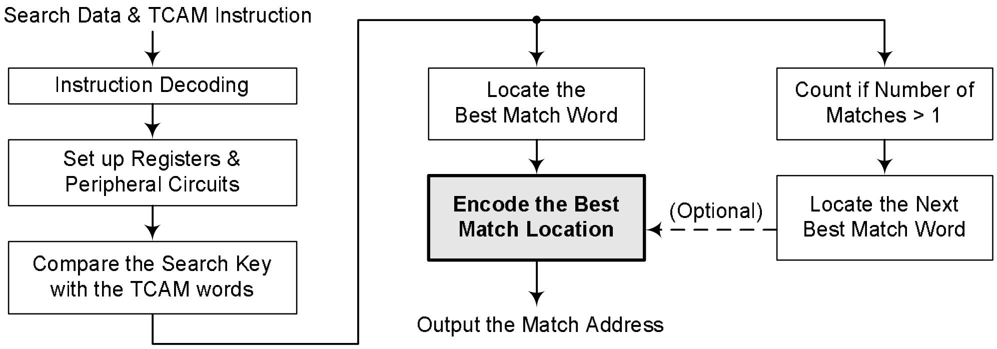

Search Data & TCAM Instruction
Instruction Decoding
Set up Registers & Peripheral Circuits
Compare the Search Key with the TCAM words
Locate the Best Match Word
Encode the Best Match Location
Output the Match Address
Count if Number of Matches > 1
Locate the Next Best Match Word
(Optional)

Figure 5.1: The Role of Encoding the Match Address in a Ternary CAM Search

Some issues in the design of MAE were usually overlooked in the past publications on TCAM. They assumed that the MAE can be in any ROM (Read-Only-Memory) encoding structure. However, due to the interfacing with the multiple match resolver (MMR), different types of ROM-like encoder may have a different power consumption. The purpose of this chapter is not to serve as a reference for ROM circuit design. Instead, it is only a brief chapter summarizing the keys in choosing the right ROM encoder as the MAE for low-power TCAMs.

5.1 The Need of Encoding the Address into Binary Format

A state-of-the-art TCAM IC can have up to 512k words [13]. This translates to a 19-bit address space, and the requirement of a 512k-to-19 binary encoder. Definitely this is impossible to be done in a single stage. The MAE for high-density TCAMs are performed in multiple stages, similar to the ways in multi-level MMRs. These multi-level encoding and match resolution stages can significantly increase the overall latency of a TCAM search.

In fact, if there are on-chip SRAM blocks coupled to the TCAM arrays, the MMR outputs can directly serve as the SRAM word line drivers. The highest priority "Match" signal can serve as an index to retrieve the search results. This way the match address encoding and the decoding in the off-chip SRAM can be omitted. This embedded SRAM scheme has been studied in [8] and [23]. However, modern TCAMs usually omit the on-chip SRAM because its absence offers a higher effective TCAM capacity, and many lookup applications require a non 1-to-1 correspondence between TCAM and RAM [24]. The associated data is typically stored in off-chip SRAM, in a location specified by the CAM match address encoded in binary form. This justifies the need of having match address encoders after the multiple match resolution stage.

5.2 Basics of a ROM Encoder

Figure 5.2 shows a simple dynamic CMOS ROM-like encoder for match address encoding. It is a NOR-type encoder because the transistors are connected in parallel like a wired-OR gate. The operation is extremely simple. All the bitlines (BLs) are pre-charged to V_{dd}. The absence of a NMOS indicates a "0", while a "1" is indicated by connecting a NMOS with its drain on the BL and its gate on the wordline (WL). Here, the wordline is denoted by R_i because the inputs of the MAE are the outputs of MMR in this context. If R_0 is the highest priority "Match", the resulting match address would be "000...000". Likewise, if R_1 is the highest priority "Match", the address is "000...001".

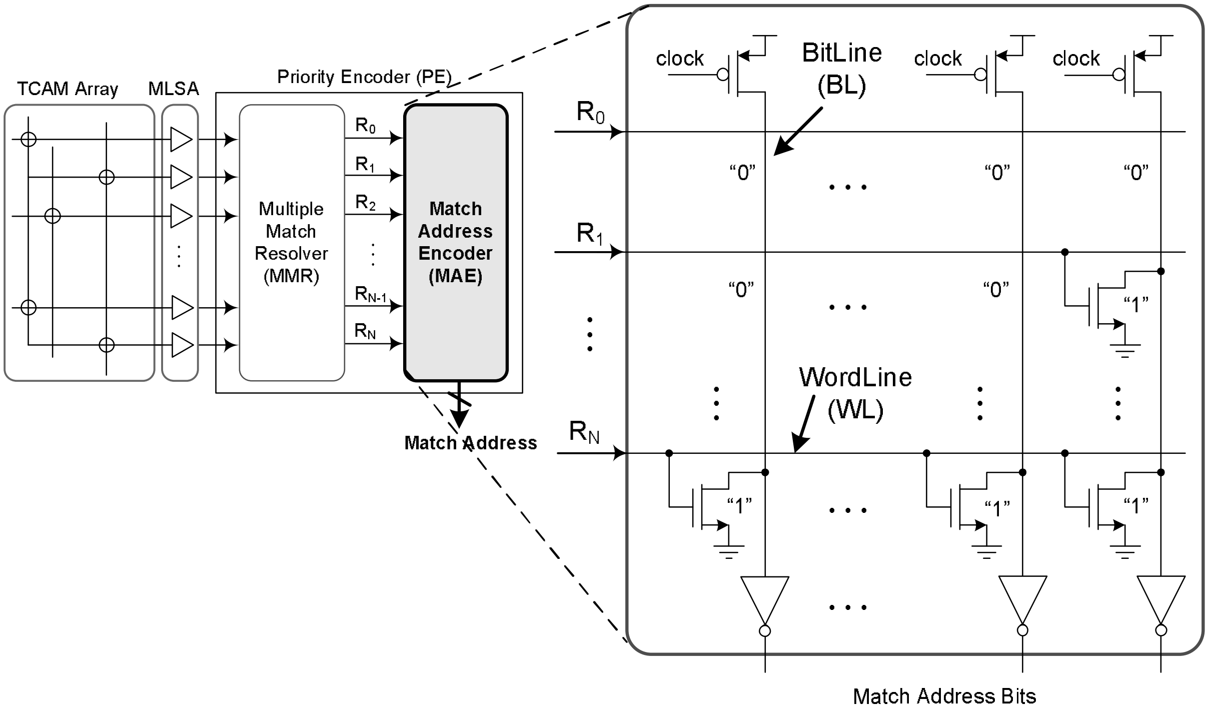


Figure 5.2: A Simple Dynamic CMOS Encoder

In the logic perspective, the simple encoder in Figure 5.2 is functionally correct. It is capable to perform match address encoding. However, this design is only good enough for small encoders, such as 8-to-3 or 16-to-4 bits encoding. The primary limitations are “speed” and “leakage”. It is a well-known fact in semiconductor science that a MOS transistor is still conducting current when V_{gs} is 0 V. This “off current” increases exponentially from generation to generation [25]. If a 256-to-8 bit encoder is designed based on the simple circuit in Figure 5.2, the pre-charge PMOS on the BL is definitely not strong enough to fight against all leaking paths. For 0.13 μm CMOS technology, the wired-OR logic gate is only reliable up to 16-bit inputs [26]. In the coming sections, we will try to explore the techniques for compensating the leakages. Analysis on power consumption of different ROM structures will be presented in conjunction.

5.3 Two Unique Properties of Match Address Encoder

In general, the TCAM environment imposes two unique properties on the design of MAE. These properties can help to relax the constraints in encoder design, and to save power consumption.

1. As previously studied in Chapter 2 and 3, when multiple matches occur, the MMR always favors the "highest priority match" associated with the lowest physical address. The MAE should be designed to take the advantage of this property. The idea is to make the common cases consuming lower power. This can be done if the higher priority wordlines (R_i) have a lower switching activities on the BLs of the MAE.

2. Unlike the high-density ROM circuits, where the density is the main concern, the MAE in TCAM is not density-critical [21]. A MAE cell is very small. It is usually stretched to have the same pitch as that of the TCAM cell. Additional logics can be built into these spaces to enhance the reliability and speed of MAE if required.

In the next section, we will explore several ROM circuits, and comment on their power consumption when they are employed in the TCAM environment.

5.4 Low Power ROM-like Encoders

5.4.1 Differential Sensing with Reference Circuits

The leakage problem, as previously described in Section 5.2, is a concern to the BL sense amplifier (BLSA) if it is using a fixed switching threshold to distinguish a "1" and a "0" on the BL of MAE. Hence, a better design is to compensate the threshold voltage by taking the leakages into consideration. However, the "off-current" of a MOS is dependent on temperature and process variations [15]. A simple reference circuit is not sufficient for accurate modeling. Figure 5.3 shows a reference circuit that models the process variations and temperature effects.

The reference circuit is composed of two complete columns. One column is responsible for modeling a "0", another is for modeling a "1". The average of those two is used as a decision threshold in the BLSAs. Note that the BLSAs in this scheme can be either voltage-based or current-based. From the energy consumption perspective, this scheme satisfies the first property as described in Section 5.3. It favors the higher priority inputs to reduce switching activities on the BLs. However, there are two drawbacks. First of all, the voltage swing of the reference circuit has a large energy-overhead as compared to the total MAE energy consumption. Secondly, the timing

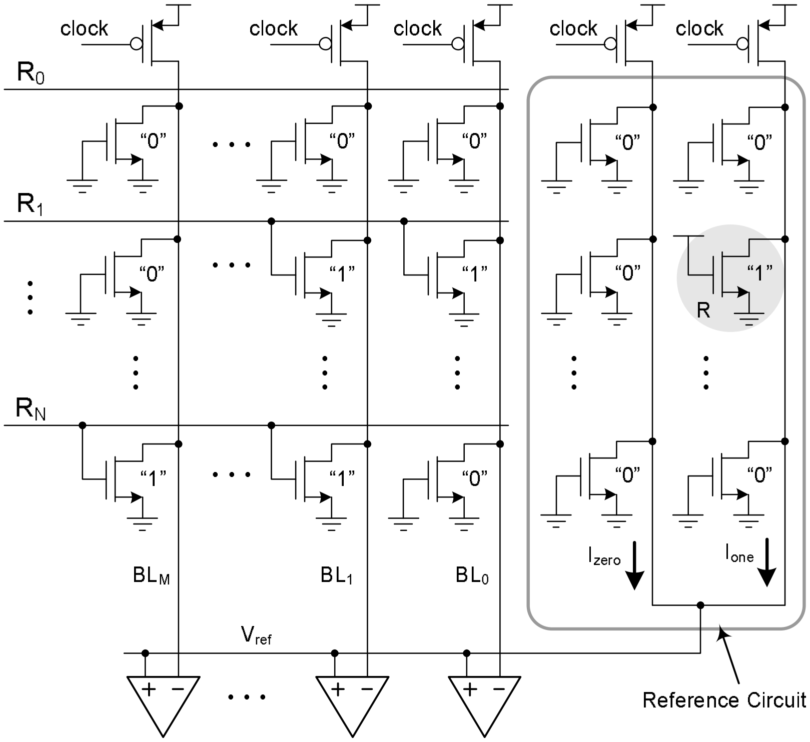


Figure 5.3: Differential Sensing with Reference Circuits

of the signal for turning on transistor R in the reference circuit, as shown in Figure 5.3, must be controlled very precisely.

5.4.2 Dual-BL Differential Sensing

Figure 5.4 illustrates a dual-BL differential sensing scheme for ROM-like encoder. Such differential BL pair intrinsically offers common-mode noise rejection [22][27]. Notice that "leakage current" is considered as one type of noise in this regard. The decision of whether an address bit is "1" or a "0" is determined based on the polarity of the BL_i and $\overline{BL_i}$. If a "0" is asserted, the voltage of $\overline{BL_i}$ will be lower than that of BL_i, and vice versa. This scheme is not limited to voltage sensing. Current sensing is also common in high-density ROMs. However, as mentioned earlier, density is not a concern to the MAE in TCAM.

This type of dual-BL ROM encoder is employed as the MAE in a commercial TCAM design [21]. It is reliable and robust with built-in compensation for temperature variations and process

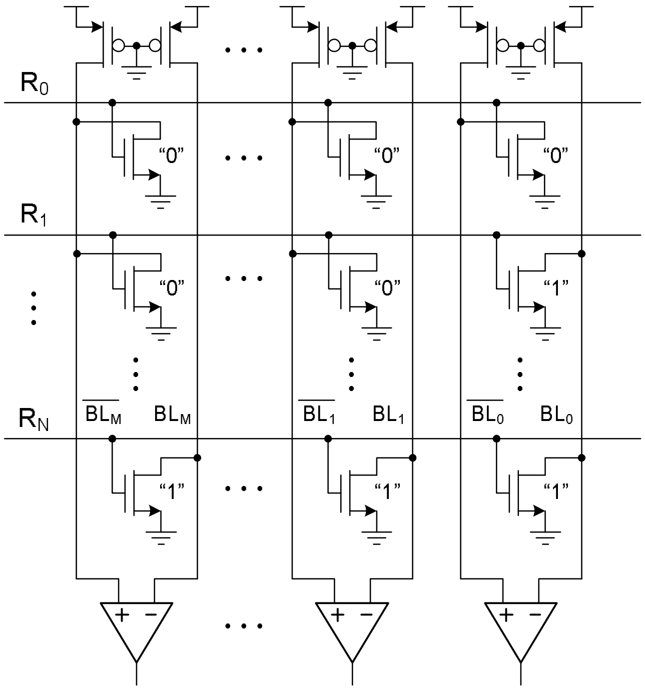


Figure 5.4: Dual-BL Differential Sensing

variations. However, it is not a low-power encoder. If an N-to-M address encoder is implemented using this scheme, there are M BLs being charged and discharged every clock cycle. This is close to the worst-case energy consumption in the former example in Section 5.4.1. The voltage swing of the BLs are expected to reach full-rail (V_{dd} to V_{ss}) because the BL capacitance is relatively small for a 128-to-7 MAE or a 256-to-8 MAE. Hence, the advantages of this dual-BL ROM encoder are not applicable to the TCAM environment. In addition, this scheme does not take the advantage of "making the common case low power".

5.4.3 Current-Race Sensing with Reference Circuits

Arsovski in [28] proposed a Current-Race Sensing scheme for Matchline Sense Amplification. The same idea can be applied in MAE for BL sensing, as shown in Figure 5.5. In fact, this circuit is very similar to the scheme in Section 5.4.1. However, the difference is that all BLs are pre-charged to V_{ss} instead of V_{dd}. The sensing is done by comparing the charge-up time of the BLs to the reference circuit. A more completed description of this circuit can be found in Section 6.4.2 or [28].

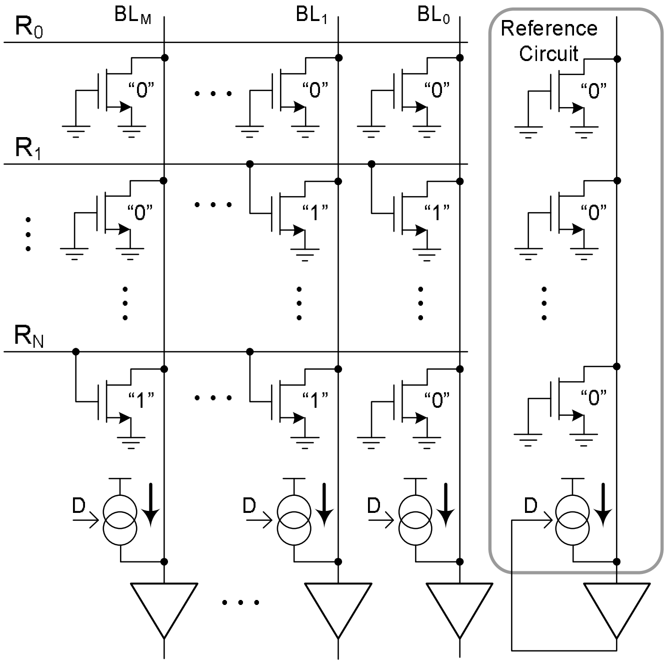


Figure 5.5: Current-Race Sensing with Reference Circuits

This type of sensing scheme is beneficial for Matchline sensing in TCAM array. However, it is also not a low-power encoder. Its design does not favor higher priority inputs, as discussed in Section 5.3. The power consumption of this scheme is nearly independent of the "match" location if it is used as the MAE.

5.4.4 Digital Sensing using Hierarchical BL Architecture

Figure 5.6 shows a simple hierarchical encoding circuit. The BL is split into two levels to reduce the fan-in of the wired-OR gate. This type of architecture is usually employed in logic design or datapath designs, but not in high-density memory environment. However, as described in Section 5.3, the cells in MAE are loosely coupled. The additional logics can be fit into the empty spaces without increasing area.

This design is low power in two ways. First, the higher priority wordlines have a lower switching activities on the BLs of the MAE. Secondly, the Global BL (GBL) capacitance is small in compared to the BL capacitance of the prior schemes. Hence, a full-swing charging and discharging of the GBLs are not consuming as much power in compared to the prior designs.

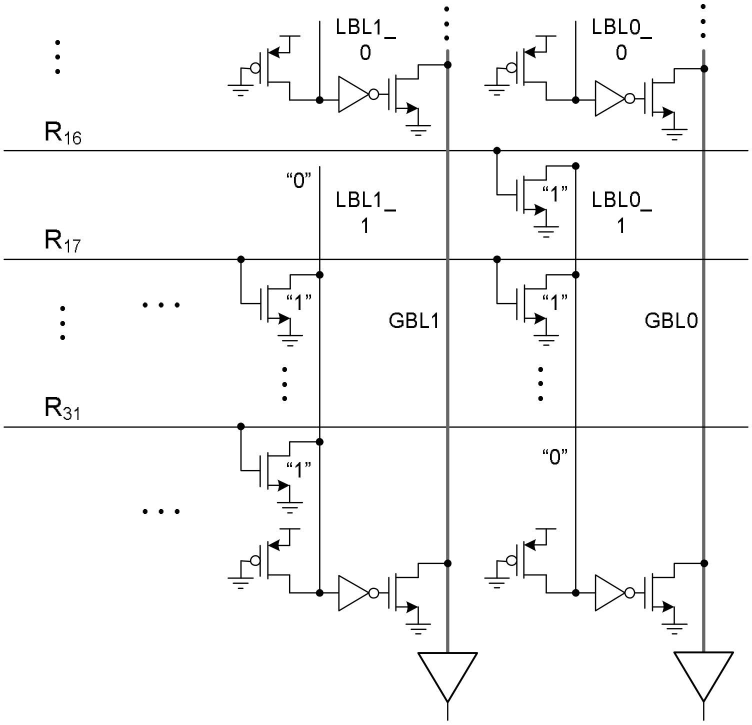


Figure 5.6: Simple Hierarchical BL Architecture

5.5 Issues in Physical Layout of MAE

A high-density TCAM is usually segmented into smaller blocks of TCAM arrays. In a conventional design, each block is equipped with a local MAE and a local MMR. The arrays are usually mirrored so that the MAEs are positioned back-to-back as shown in Figure 5.7.

As previously studied in Section 5.3, a ROM cell is only a simple circuit composed of one or two MOS transistors. However, a conventional TCAM cell has 16 transistors for the static implementation, and 6 transistors and 2 capacitors for the dynamic implementation [4]. It is clear that either one consumes a much larger silicon area as compared to that of a ROM cell. Pitch-matching the ROM cells to the TCAM array may create a lot of wasted chip area. One possible layout method is to physically mingle the two local MAEs, so that the wordlines (WLs) of the MAE are driven by the MMRs from both sides in an interleaved manner, as depicted in Figure 5.8(a). This interleaved WL approach has been demonstrated in a commercial TCAM design [21][29].

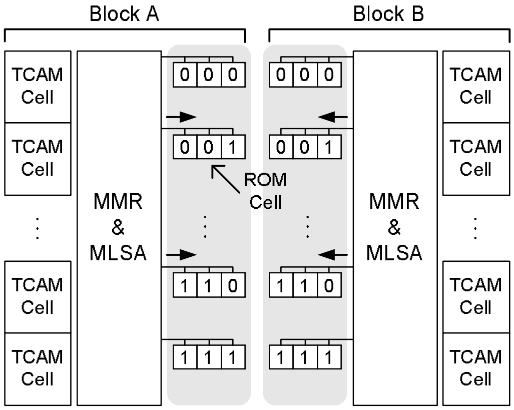


Figure 5.7: A Conventional Layout of MAE

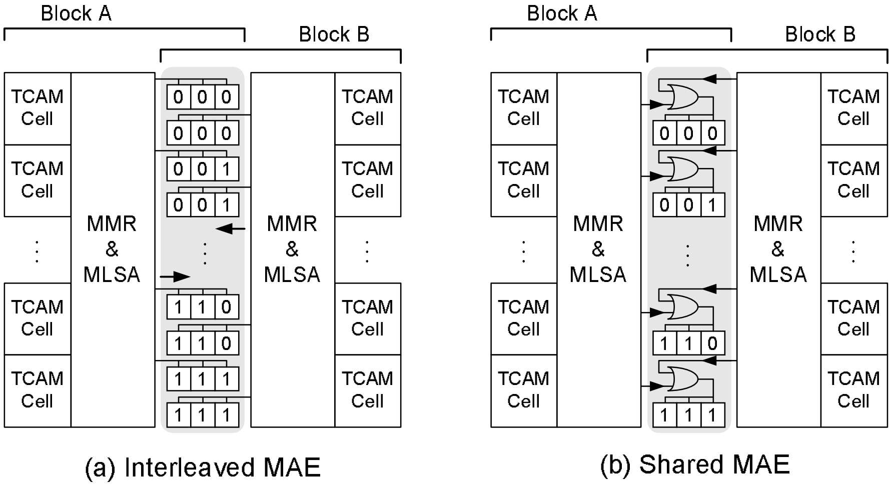


Figure 5.8: Efficient Layout of MAEs (a) Interleaved (b) Shared WL

The interleaved scheme has the advantage of reducing the area overheads occupied by the MAEs. In theory, the area reduction can be as much as 50%. However, this approach increases the MAE BL sensing delay as a result of doubling the diffusion capacitance. Figure 5.8(b) illustrates a proposed alternative to replace the interleaved scheme. It employs a shared wordline architecture. An additional OR gate is coupled to every wordline in the MAE for interfacing with the MMRs from both sides. Although static OR gates are shown in the diagram, the proposed design is not limited to pseudo-NMOS wired-OR logics. These additional wired-OR gates can be placed into the unused spaces without area penalties. With proper layout design, this scheme can achieve a 40% reduction in MAE bit-line capacitance as compared to the interleaved WL approach.

Chapter 6

Multiple Match Detection

In this chapter, we will focus on the design of Multiple Match Detector (MMD) for TCAMs in CMOS technology. This step is performed in parallel with the multiple match resolution stage. Figure 6.1 illustrates the role of MMD in a TCAM Search.

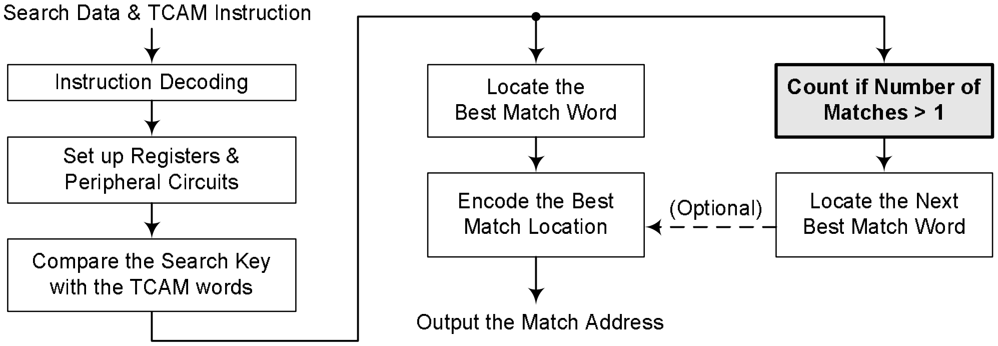

Search Data & TCAM Instruction
Instruction Decoding
Set up Registers & Peripheral Circuits
Compare the Search Key with the TCAM words
Locate the Best Match Word
Encode the Best Match Location
Output the Match Address
Count if Number of Matches > 1
Locate the Next Best Match Word
(Optional)

Figure 6.1: Multiple Match Detection in the Flow of a TCAM Search

6.1 The Need of Mulitple Match Detection

In the early development of TCAMs, the lower priority “internal match” signals were all discarded and never acknowledged. However, many recent algorithms in computer networking and image processing require partial matching and sequential output of all match addresses in prioritized order. This requirement proposed the need of a sensing circuit to detect multiple matches, and a simple method to output the lower priority “match addresses” in consecutive cycles upon request.

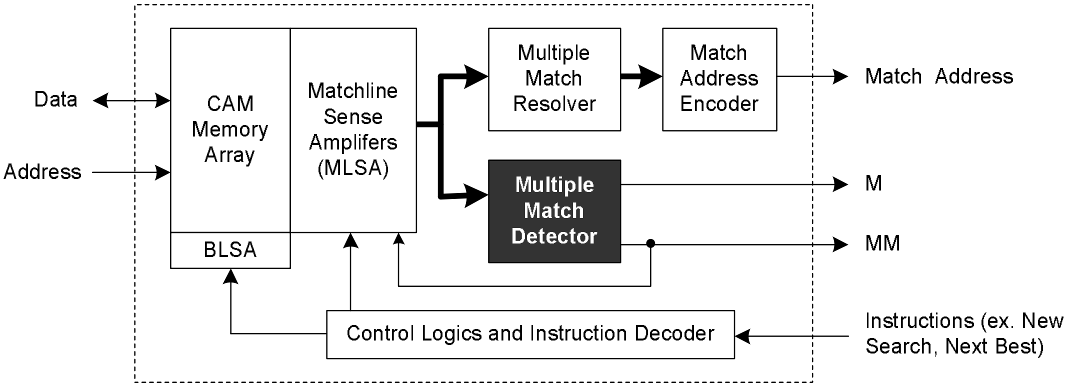
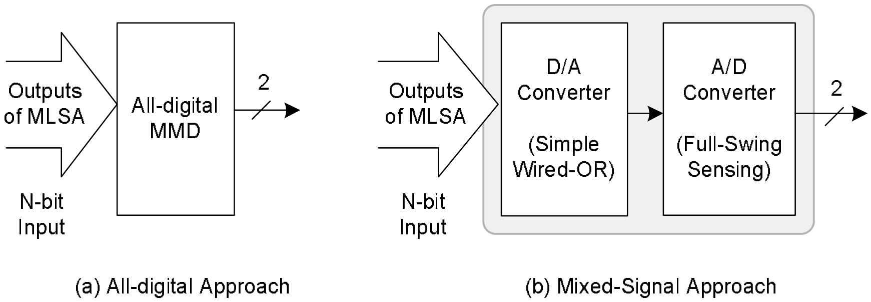


Figure 6.2: Multiple Match Detection in Ternary CAM

As illustrated in Figure 6.2, an output signal “MM” serves the role of notifying an external processor if there are multiple matches in a TCAM search. The external processor has the option to retrieve all other “match results” in a “burst” mode, or to start a new search in the following clock cycle. The decision is based on the instruction provided to the TCAM. Hence, a TCAM is actually a co-processor with the instruction set targeting for high-speed lookup applications.

6.2 General Architecture

Unlike the sense amplifiers for Matchlines (MLs) and Bitlines (BLs), which employ a single threshold to characterize an analog input as either a ’1’ or a ’0’, a multiple match detector (MMD) is a ternary decision circuit. This ternary result is usually encoded into a 2-bit output.



(a) All-digital Approach

(b) Mixed-Signal Approach

Figure 6.3: Various Methods for Multiple Match Detection

In general, there are two categories of MMDs, as shown in Figure 6.3. The all-digital approach is slower, and usually requires complex digital circuitry. However, it is more reliable if given enough time for evaluation. In contrast, the mixed-signal approach is faster and generally requires low transistor counts. It is based on sensing either the difference in voltage or in current.

6.3 All-Digital Multiple Match Detectors

6.3.1 General Considerations

The logic of detecting one match and multiple matches in TCAM can be expressed using Equation (6.1) and (6.2) respectively, where M denotes the match flag, and MM denotes the multiple match flag. A similar type of complexity analysis was done, in parallel to this work, in [30] and [31].

$$M = In_0 + In_1 + In_2 + \ldots + In_N \tag{6.1}$$

$$MM = In_0 \cdot In_1 + In_0 \cdot In_2 + \ldots + In_0 \cdot In_N + In_1 \cdot In_2 + \ldots + In_{N-1} \cdot In_N \tag{6.2}$$

An all-digital CMOS realization of (6.1) and (6.2) is shown in Figure 6.4. Each 2-input AND gate in (6.2) is realized using 2 NMOS transistors connected in series as the pull-down path. The pull-up can be either a grounded PMOS (pseudo-NMOS logic) or a clocked PMOS with a small keeper. Sense amplification is also an option here to reduce the detection delay.

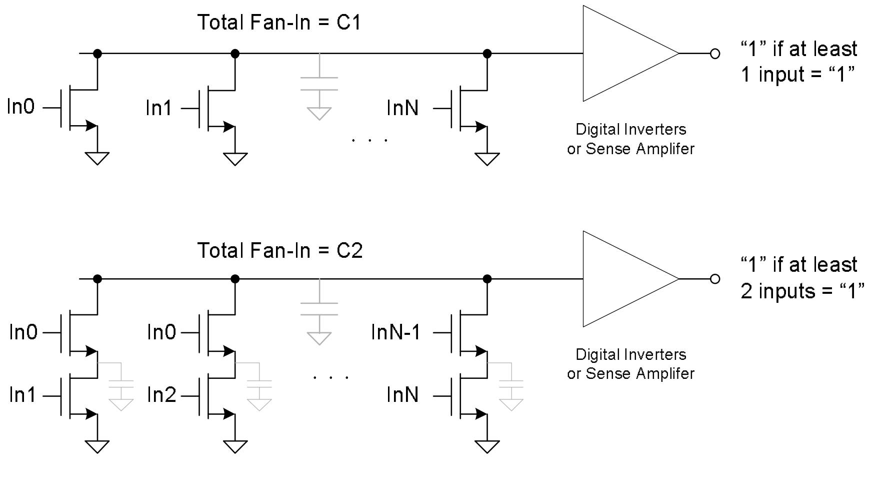


Figure 6.4: Wired-OR CMOS Realization of Equation (6.1) and Equation (6.2)

This digital method looks simple and easy-to-implement, but physically it is impractical. The fan-in of the OR-gate for single match detection, denoted by C1, is N. The increase is linear and still manageable for N-bit input. For multiple match detection, the fan-in of the OR-gate, denoted by C2, is given by Equation (6.3). To have a better understanding of the complexity, Figure 6.5 shows C1 and C2 versus the number of MMD inputs (which are the MLSA outputs from the prior stage).

$$C2 = \frac{N(N-1)}{2} = \frac{1}{2}N^2 - \frac{1}{2}N \tag{6.3}$$

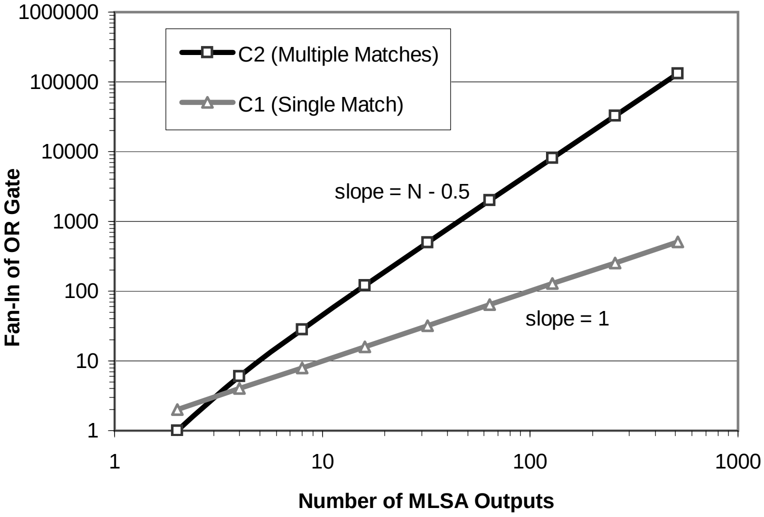


Figure 6.5: Complexity of the OR-logic vs. Number of MLSA Outputs

In order to detect multiple matches in a group of 256 inputs, the detector requires an OR-gate with fan-in of 32640. Definitely the physical area for realizing this 32640-input OR-gate is a concern. Other issues include: (1) the number of inter-connections, (2) the capacitive loading on the MLSA output drivers, (3) the long sensing delay, and (4) poor pitch-matching to the cell array, which is a primary concern to high-density TCAM circuits. Most of the shortcomings mentioned above are consequences of large fan-in. It is apparent that this simple digital logic method is not practical for wide-input multiple match detection.

6.3.2 Multiple Match Logic Simplification using MMR Outputs

One way to reduce the complexity of (6.2) is to group the AND terms together as shown in (6.4).

$$MM = In_1\left(In_0\right) + In_2\left(In_1 + In_0\right) + \ldots + In_N\left(In_{N-1} + \ldots + In_1 + In_0\right) \tag{6.4}$$

According to the De Morgan's Theorem, (6.4) can be further re-arranged into the form as shown in (6.5). All three equations have the same logic equivalence.

$$MM = In_1 \cdot \overline{\left(\overline{In_0}\right)} + In_2 \cdot \overline{\left(\overline{In_1} \cdot \overline{In_0}\right)} + \ldots + In_N \cdot \overline{\left(\overline{In_{N-1}} \ldots \overline{In_1} \cdot \overline{In_0}\right)} \tag{6.5}$$

Although this rearrangment is trivial, it has an important implication such that the MMR outputs can be used to reduce the multiple match logic complexity. With little further re-arrangment, (6.5) can be re-written in the form as shown in (6.6). Notice that the terms inside the brackets are exactly the logic representations of the MMR outputs. Hence, the multiple match detection can be done based on logic equation (6.7), where R_i is the corresponding MMR output signals. Using this method, the complexity of detecting multiple matches can be reduced from second-order to first-order. The simplified OR-gate has a fan-in of N, which is identical to the logic for single match detection.

$$MM = In_1\overline{\left(In_1 \cdot \overline{In_0}\right)} + In_2\overline{\left(In_2 \cdot \overline{In_1} \cdot \overline{In_0}\right)} + \ldots + In_N\overline{\left(In_N \cdot \overline{In_{N-1}} \ldots \overline{In_1} \cdot \overline{In_0}\right)} \tag{6.6}$$

$$MM = In_1 \cdot \overline{(R_1)} + In_2 \cdot \overline{(R_2)} + \cdots + In_N \cdot \overline{(R_N)} \tag{6.7}$$

This idea of using MMR outputs to reduce the Multiple Match logic complexity was proposed and patented by Jiang in [32]. When an input of MMR is a "1" (a "Match"), the corresponding output would also be a "1" if it is the highest priority match. Otherwise, it is a "0" because there is at least a higher priority match prior to the current input. Hence, we can conclude that there are multiple matches in the block if, at the end of the MMR evaluation, at least one pair has an input of “1” and an output of “0”. The idea is summarized in Table 6.1.

MMR Input (In_i)	MMR Output (R_i)	Multiple Matches?
0	0	Don't Know
1	1	Don't Know
1	0	Yes
0	1	An Error in MMR

Table 6.1: Detecting Multiple Matches based on the Input/Output Patterns of MMR

An efficient realization of Equation (6.7) is shown in Figure 6.6. The circuit can be implemented using pure digital circuits. This scheme is particularly suitable for automated TCAM memory compiler, where the TCAM block size can be customized at the compile time. Automated design tools can use this method because the entire circuit is digital [32]. Digital logic can guarantee correct functionality if given enough time for evaluation. This technique also allows ease of cascading numerous MMDs in multiple levels or across multiple CAM chips. Another advantage is its support of variable word width feature for commercial TCAMs. With the variable word width circuit, the inputs to the MMD or MMR are not coming directly from the MLSAs [33].

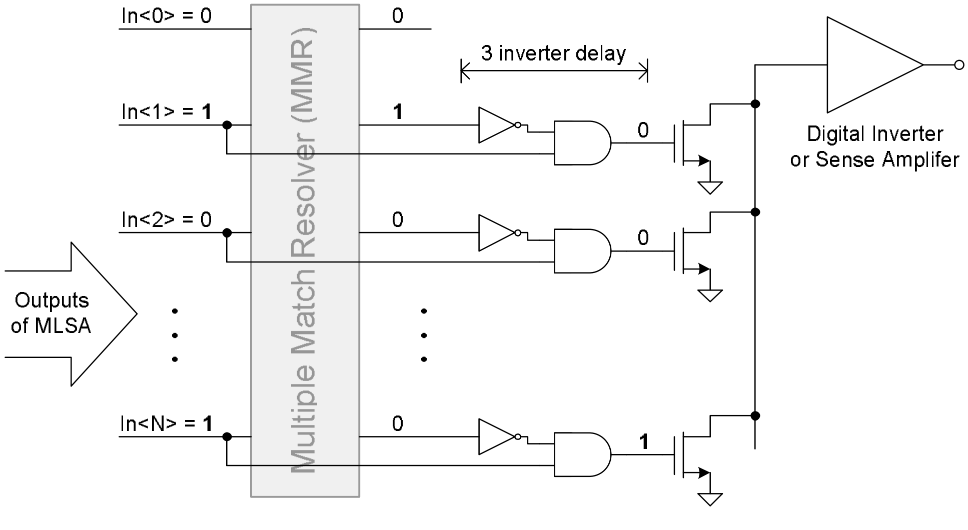


Figure 6.6: Transforming Multiple Match Detection into Single Match Detection

On the negative side, this implementation has several drawbacks. First of all, the evaluation phase of this MMD cannot begin until all the MMR outputs are settled. An early start would cause either false evaluation, or excess energy consumption due to the unnecessary switchings. Hence, the total latency of such design is equal to the sum of “the worst-case MMR delay”, “the wired-OR logic delay”, plus 3 inverter propagation delay. It is long and either the MMD or MMR has to be idle without having work done. In addition, this delay has a significant impact on the clock period. Pipelining the circuits can increase throughput, but on the other hand, the latency is further deteriorated due to the clock element overheads. For completeness, Figure 6.7 illustrates an example of inter-block multiple match detection using this digital scheme.

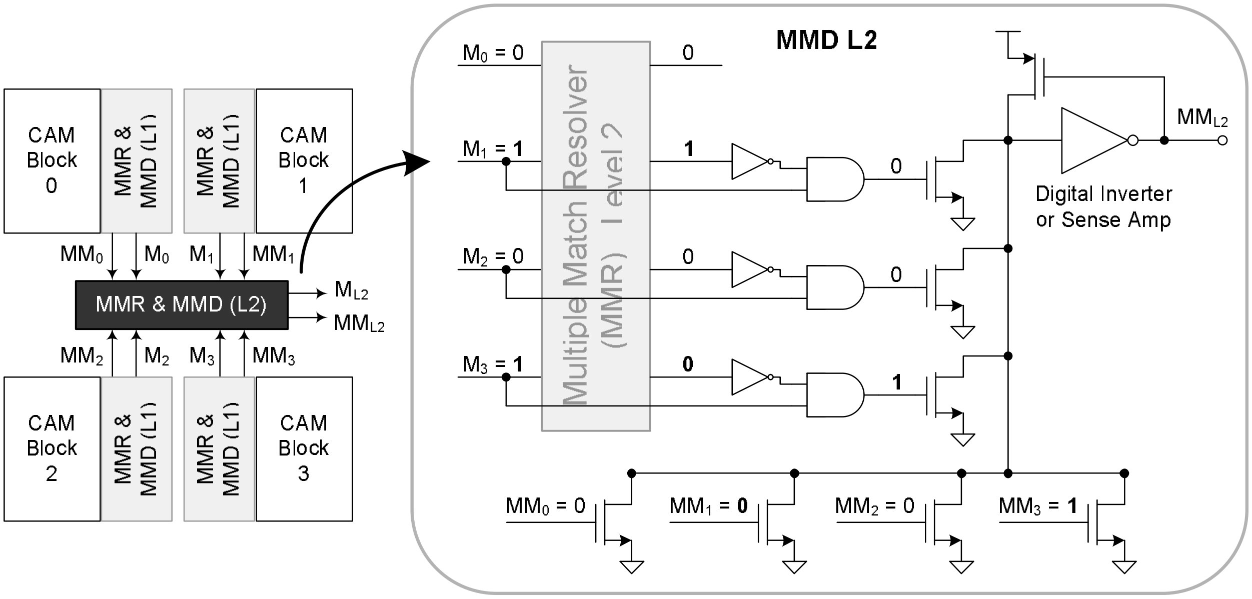


Figure 6.7: Inter-block Multiple Match Detection using Multi-level MMR Outputs

6.4 Mixed-Signal Multiple Match Detectors

6.4.1 A “Voltage-Compare” Multiple Match Detection Scheme

If the digital methods are too complicated, we can always trade the robustness of digital logic for additional design flexibilities offered by its analog counterpart. Figure 6.8 shows the block-level diagram of a voltage-based multiple match detector. The wired-OR logic is used to convert the digital MLSA outputs to a time varying analog voltage (or current). This voltage is then compared against two reference voltages using two analog comparators. If the voltage of "Multiple Matchline" (MML) is below the reference voltage $V_M(t)$, there is at least one match. Likewise, if the voltage of MML is also below the second reference voltage $V_{MM}(t)$, there are multiple matches in the TCAM block.

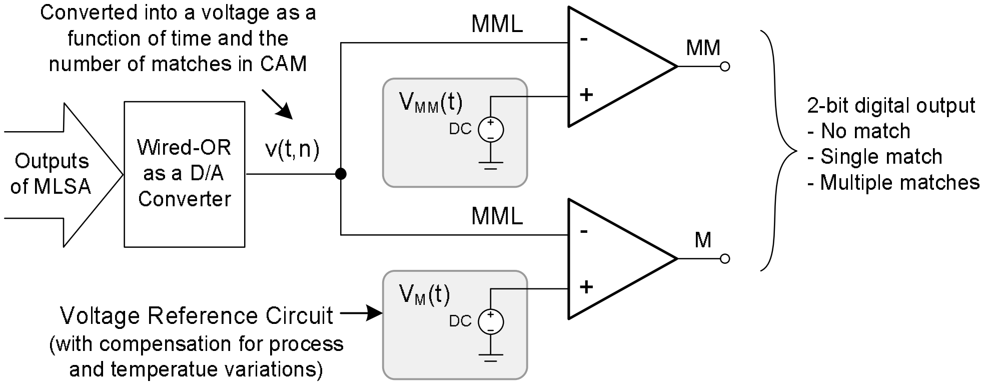

Converted into a voltage as a function of time and the number of matches in CAM
Outputs of MLSA
Wired-OR as a D/A Converter
v(t,n)
MML
-
+
MM
VMM(t)
DC
MML
-
+
M
VM(t)
DC
Voltage Reference Circuit
(with compensation for process and temperatue variations)
2-bit digital output
- No match
- Single match
- Multiple matches

Figure 6.8: A Simple Mixed-Signal Multiple Match Detector

The comparators in the diagram can be any differential pair with full-swing digital output. The reference elements V_M(t) and V_{MM}(t) are usually not fixed voltage references. Compensation for temperature, process variations, and supply noise rejection are built into the circuit for accurate modeling of the decision thresholds. Figure 6.9 shows the circuit schematic of an implementation proposed by Bosnyak in [34]. Note that this diagram shows only the comparator for output MM. The complete circuit consists of two comparators and two reference elements.

Circuit Operation

The transistors T1 and T2 form a source-coupled differential pair for sensing the voltage-difference on the MML and RMML. The circuit is in idle state when the external control SHL is at "0". Prior to the detection, nodes B, C, MML and RMML are all pre-charged to V_{dd} by transistors T11, T12, T9, and T10 respectively. At the on-set of the detection, the MML is pulled down by the NMOS evaluation transistors (TNs). The discharge rate is determined by the number of matches in the TCAM array. At the other corner, the RMML is pulled down by a single dummy NMOS (1.5x

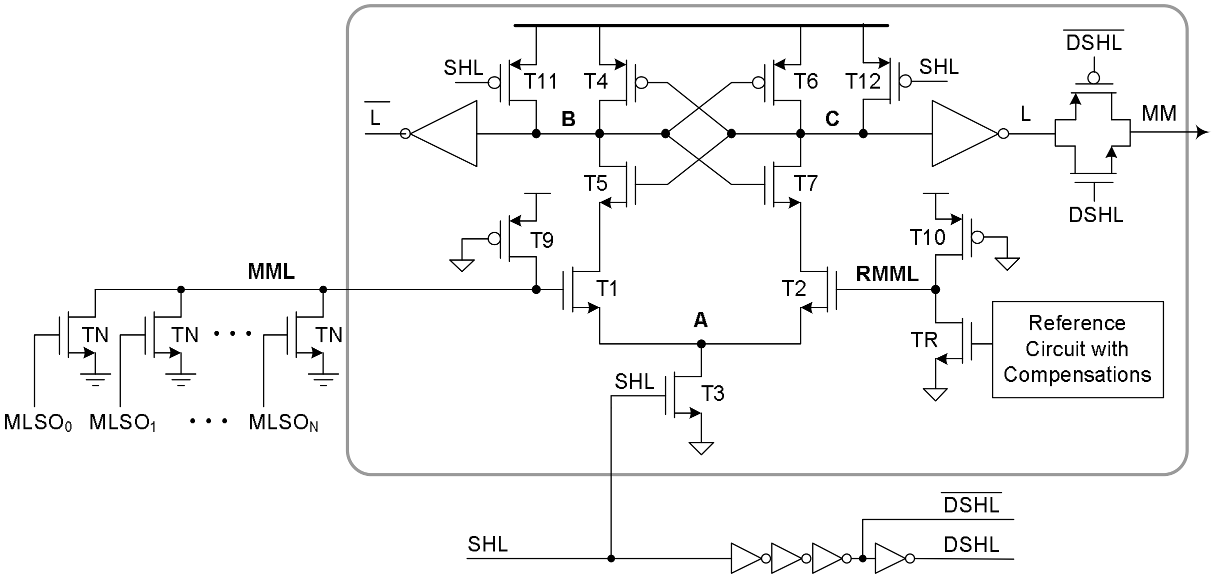


Figure 6.9: A Multiple Match Detection Scheme proposed by Bosnyak

larger) to emulate “1.5 matches”. For the “no match” and “single match” cases, the MML voltage is pulled down at a slower rate than the rate of RMML counterpart. Hence, transistor T1 is less resistive in compared to T2. When the 0-to-1 transition of SHL arrives, node B is pulled downward to a lower voltage-level than that of node C. The positive cross-coupled feedback further amplifies their difference, and pull node B to V_{ss} and node C completely to V_{dd}. Hence, the output L has a final value of “0” that indicates no “no match” or “single match”. The vice versa occurs for the case of “multiple matches”.

The Shortcomings

The implementation in Figure 6.9 is simple but has a number of shortcomings. First of all, sizing TR to 1.5x of the width of TN does not place the decision threshold in the middle of “single match” and “two matches”. It is because the MML is long and resistive, which adds additional resistance to the discharging path. Secondly, the reference circuit controlling the gate of TR is extremely complex. It is hard to align this control signal to match the phase of the MLSA outputs. Thirdly, the pseudo-NMOS transistors T9 and T10 are consuming static power during the detection. This design generally consumes high-power. It is not a good design for low-power TCAM chips.

Ahmed in [35] proposed another implementation to suppress some of these shortcomings. Note that the reference element can be something else other than a dummy NMOS transistor. The circuit schematic of the improved comparator is shown in Figure 6.10. A complete RMML is placed in parallel to the MML for better matching of process variation and temperature variation. The outputs L and $\overline{L}$ are OR-ed to switch off T9 and T10 as soon as the sensing is completed to reduce static power.

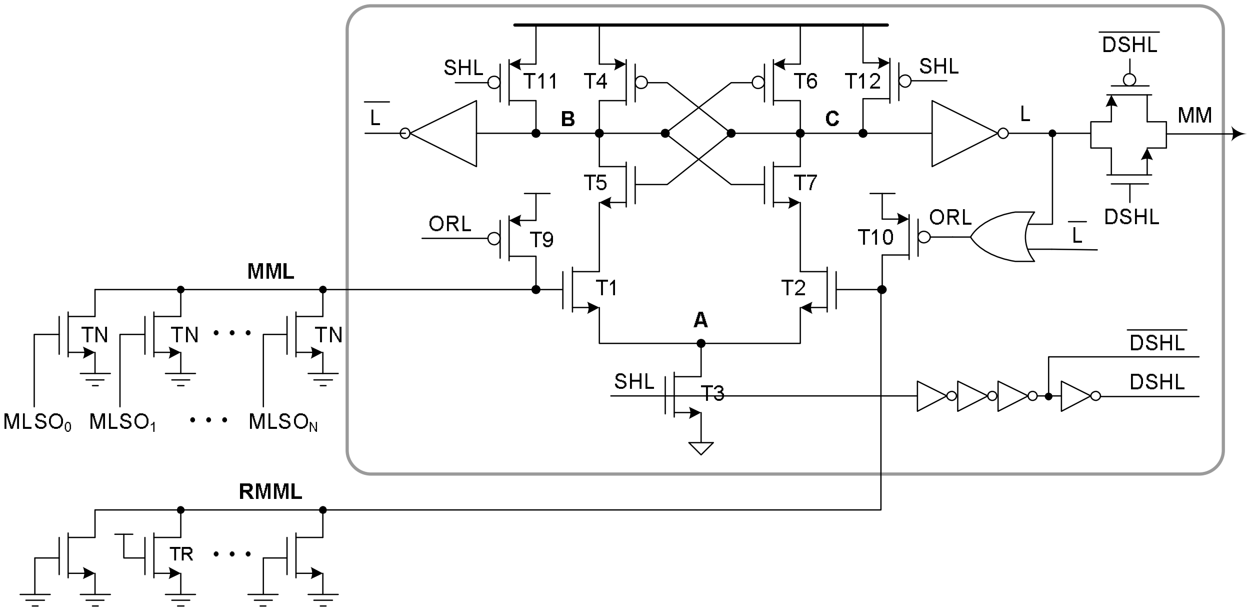


Figure 6.10: A Multiple Match Detection Scheme proposed by Ahmed

Many references, such as [31] and [35] states that the optimal size of TR is 1.5x of the width of TN. However, TR should sized a bit smaller than 1.5x because the wire resistance is not scaled as a function of number of matches. Simulations show that the optimal size is around 1.4x (depending on the wire length and technology).

The improved implementation still has a number of shortcomings. They include high leakage power during the idle state, and high dynamic power consumption due to the requirement of two comparators for complete multiple match detection. The high leakage power is due to the large fan-in on MML and RMML.

6.4.2 A "Current-Race" Multiple Match Detection Scheme

The cross-coupled differential pair, previously described in Section 6.4.1, can provide only a binary output. For example, an output of "1" represents "more than 1 matches", while a "0" represents "less than or equal to 1 match". In order to distinguish "No match", "1 match", and "more than 1 matches", two sets of cross-coupled differential pairs and the reference circuits are required. This is relatively inefficient in terms of area and energy consumption.

Ma in [36] proposed a multiple match detection circuit that can generate a 2-bit encoded result representing either "no match", "single match", or "multiple matches". It employs only one reference line to detect the three conditions. The mechanism of the circuit is to compare the rising voltage rate of the MML against the rising voltage rate of a reference MML (RMML). This circuit has a self-timed control signal (EN1) to end the detection, and automatically place itself back to the pre-charge mode. Figure 6.11 shows the circuit schematic of this multiple match detector.

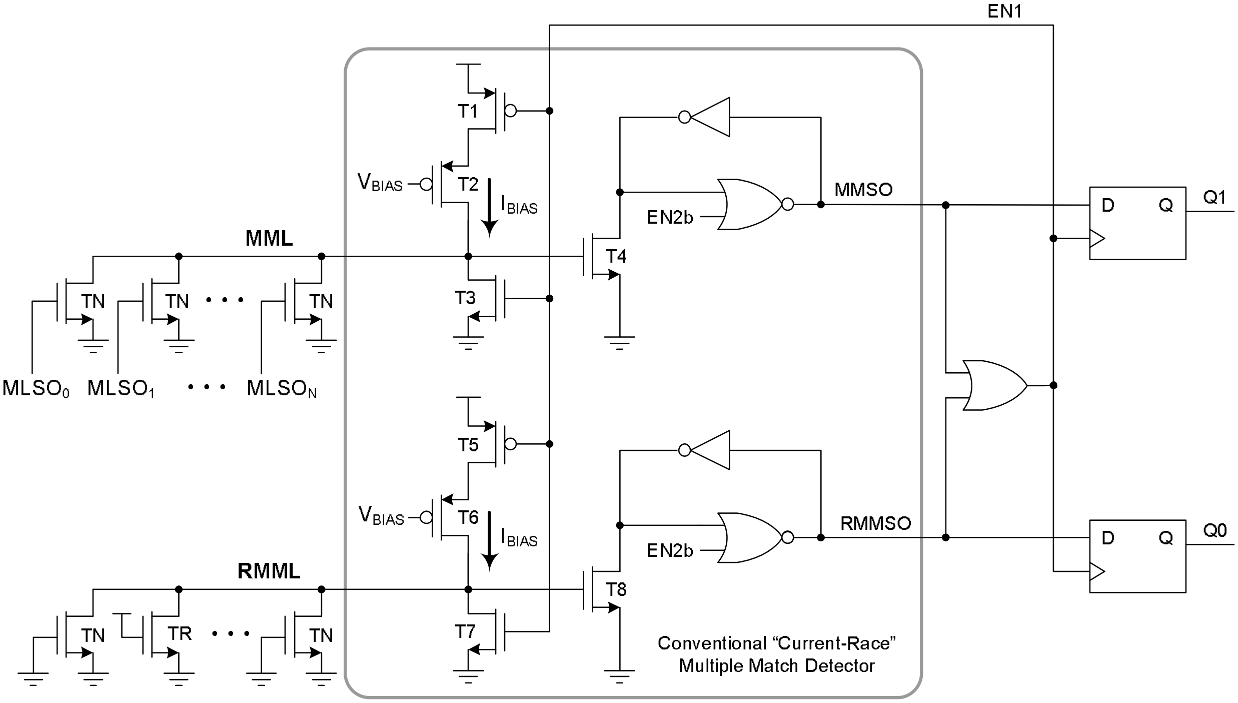


Figure 6.11: A "Current-Race" Multiple Match Detector Proposed by Ma in [36]

If the control signals are not considered, this "Current-Race" MMD has two inputs and two outputs. One input is the MML, another one is the RMML. The two outputs, MMSO and RMMSO, are connected to an OR-gate and the inputs of two D flip-flops. About the reference transistor TR, it is always conducting. The width of TR to the width of TN is in a 1:1 ratio for emulation of a “single match” case on RMML. The transistors T4 and T8 forms a simple differential pair [37]. Their “source” nodes are coupled together through the ground (V_{ss}). A current source for biasing is not needed in this case because the inputs, MML and RMML, are already pre-charged to V_{ss} prior to the on-set of the sense amplification. This simple differential pair offers the same noise rejection ratio, and a higher output swing than the differential pair with a biasing current source.

Circuit Operation

A signal timing diagram for the “no match” case of the “Current-Race” scheme is shown in Figure 6.12. Prior to the detection phase, the external signal EN2b is held at V_{ss}. The output nodes MMSO and RMMSO are either a “1” or a “0”, depending on the result of the previous detection cycle. Another control signal EN1 is at V_{dd} to pre-charge both MML and RMML to V_{ss}. This is different from the scheme in Section 6.4.1 where the lines are pre-charged to V_{dd}.

When all the MLSA outputs are settled, the multiple match detection can be started by a 0-to-1 pulse on signal EN2b. This pulse sets both MMSO and RMMSO to V_{ss}, so is the output of the OR-gate (EN1). As a consequence, it turns on the current sources coupled to MML and RMML. Each input node is charged up by a constant current source. This constant pull-up current (I_{BIAS}) is then in a race with a variable pull-down current. The magnitude is a function of the number of “matches” in the MLSA outputs. The net pull-up current determines the rising voltage rate at each input node. In the “no match” case, the rate of increase on the MML voltage would be faster than the rate on the RMML voltage. When the MML voltage is above V_{tn}, the common-source amplifier, formed by T4, is turned on. Simultaneous switching at the output nodes MMSO and the output of the OR-gate (EN1) then follow. The EN1 serves as a self-timed signal to clock MMSO and RMMSO into the D flip-flops, and to reset the MML and RMML back to V_{ss}. In this example, the two-bit encoded result {Q1,Q0} for the “no match” case is 10_2.

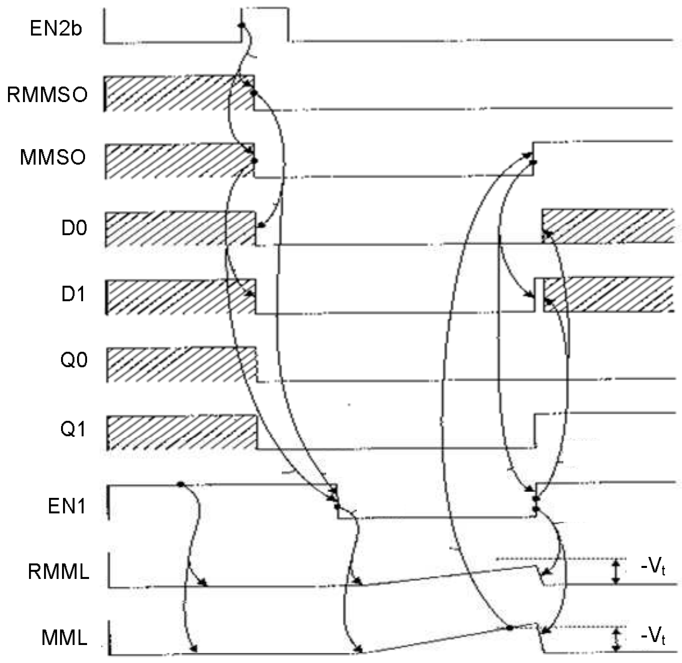


Figure 6.12: Signal Timing Diagram for the "No Match" case of the "Current-Race" Scheme (adapted from [36])

The same circuit operation applies to the "single match" case and "multiple matches" case. The only difference is the rate of increase in MML voltage and RMML voltage. Table 6.2 summarizes the conditions and encoded results for the three conditions.

Condition	Q1	Q0	Interpretation
$V_{MML} > V_{RMML}$	1	0	No Match
$V_{MML} \approx V_{RMML}$	1	1	Single Match
$V_{MML} < V_{RMML}$	0	1	Multiple Matches

Table 6.2: Interpretations of the "Current-Race" MMD Outputs (2-bit Encoded)

Key Advantages

This "Current-Race" scheme has several advantages. First, the leakage problem is not a concern at all during the "idle" mode because both MML and RMML are pre-charged to V_{ss}. A zero potential difference across the "drain" and "source" of a MOS transistor causes no current flow, and thus no leakage. During the detection phase, the leakage on both MML and RMML are considered as a common-mode noise to the differential pair T4 and T8. The second advantage is that this design is nearly self-timed and requires only one external control signal (EN2b). Thirdly, the scheme requires only one reference line and a low transistor count (almost half the number of transistors as compared to the former scheme).

6.5 Design of a Novel Multiple Match Detector (MMD)

The "Current-Race" multiple match detection scheme, as described in Section 6.4.2, is promising and attractive for low power environment. However, the circuit implementation as previously shown in Figure 6.11 does not demonstrate the true benefits of this "Current-Race" scheme. In this section, we will try to explore some circuit techniques for improving the shortcomings in the prior implementation.

6.5.1 Limitations of The Prior Implementation

In the prior implementation, the sensing speed is limited by the time of charging MML or RMML from 0V to a certain margin above V_{tn}. There are several conceptual ways to reduce the sensing delay. (Note: please refer to Figure 6.11 for interpretation of the transistor names in the following description)

1. Increase the W/L ratio or the gain of the transistors T4 and T8

2. Up-size the current sources to achieve faster rate of increase

3. Replace the normal-V_t transistors T4 and T8 by low-V_t devices

Unfortunately, none of the above ideas work well. For instance, the noise margins separating "no match", "single match", and "multiple matches" are related to the magnitude of the net pull-up current. If the pull-up current source is too strong, the net currents for all three conditions would be comparable. Likewise, the employment of low-V_t devices makes the circuit very susceptible to noises introduced at the beginning of the detection phase.

6.5.2 Innovative Circuit Ideas

One idea to increase the speed of multiple match detection is to give up the "excessive" robustness in the circuit. Figure 6.13 shows a model of the MML with distributed parasitic resistance and capacitance. The current source for charging up the MML is placed at one end close to $MLSO_N$. For a single match condition, the resistance of the pull-down path ($R_{pull-down}$) can vary from (R_{on} + r) to (R_{on} + N × r), where R_{on} is the on-resistance of the NMOS pull-down transistor. Apparently, the sensing time is shorter if a match is located at $MLSO_0$, in compared to that if a match is located at $MLSO_N$. This is because the parasitic RC network is shielding the "Sensing Point" from the pull-down NMOS transistor at the far end.

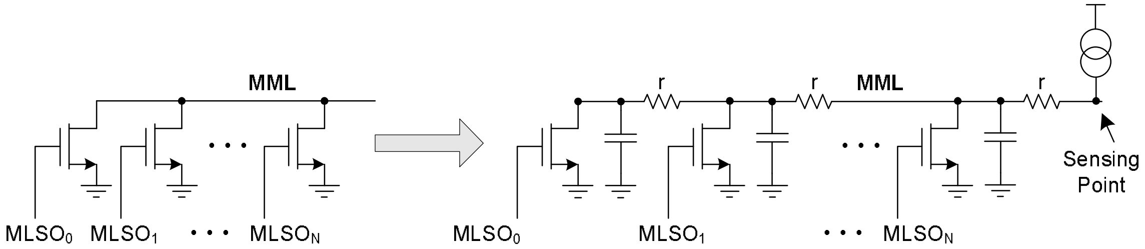


Figure 6.13: The Distributed RC Model for the Multiple Match Line (MML)

This observation has an important implication. The sensing delay of the MMD can be shortened if there is an intentional resistor shielding the sensing point from the MML. Figure 6.14 illustrates the concepts. An intentional resistor, with resistance R, is added into the picture. The rate of charging up the new sensing point, as shown in the diagram, is at the maximum if R is ∞. However, it means that the MML is completely isolated (open-circuit) from the sensing point. On the other hand, if R is too small, the new scheme has nearly no advantage in compared to the conventional implementation. The goal is to size this resistor to a value that offers a reasonable performance gain but with little deterioration to the robustness and functionality of the MMD.

This "Shielding" resistor can be easily and accurately implemented using a poly-resistor in CMOS technology. Another option is to model the resistance using a MOS pass-transistor. Note that the channel resistance of a MOS is non-linear and quite susceptible to process and temperature

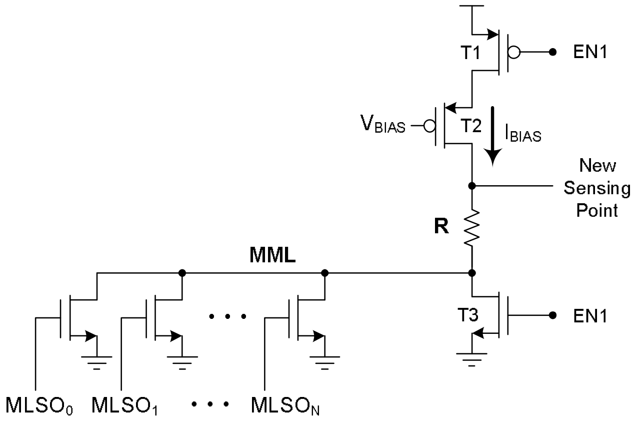


Figure 6.14: Addition of a “Shielding” Resistor for Increasing the Sensing Speed of MMD

variations. However, this behavior is not an issue because the non-linearity of the MOS channel resistance is actually offering a “feedback” to compensate the non-linearity of the current source. In a summary, our new “Current-Race” implementation employs a NMOS pass-transistor to shield the sensing point from MML (or RMML for the other half of the differential pair). Before discussing the benefits, let’s first take a look at the circuit schematic of the novel implementation as shown in Figure 6.15.

This MMD is equipped with a novel Multiple Match Sense Amplifier (MMSA). The major innovation here is the introduction of transistors T9 and T19, as shown in the figure. These two transistors help to speed up the detection process in three ways.

1. An increase at the source voltage of T9 (or T19) during the detection phase would increase the threshold voltage V_t of T9 (or T19) due to the body effect [15]. For the “no match” condition, this V_t-shift is significant. For the “single match” condition, it is moderate. For the “multiple match” condition, it is minor or even not noticeable. With respect to the sensing point MMSP, this temporal V_t-shift helps to increase the net pull-up current conditionally, and in turn helps to increase the overall sensing speed and widen the noise margins.

2. The resistance of pass-transistor T9 (or T19) shoots up when its drain voltage (V_D) is approaching V_G - V_t. Once again, this property favors the “no match” condition because MMSP is rising at the fastest rate. For the “single match” case, it is also benefited. However, for

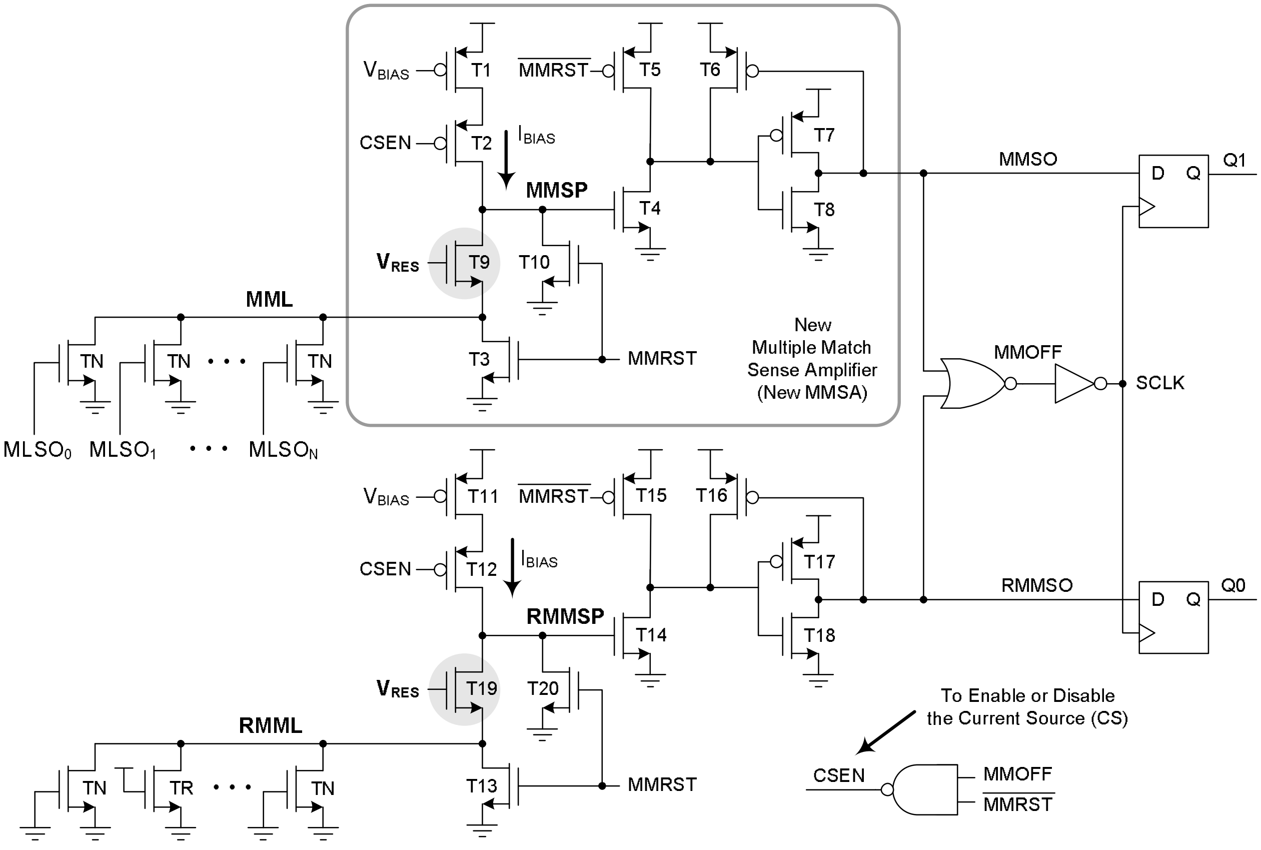


Figure 6.15: A "Current-Race" MMD with novel Multiple Match Sense Amplifier (MMSA)

the "multiple match" case, this property is of no use because the MMSP voltage is very far from V_G - V_t for the given amount of time. This property becomes handy if V_G is wisely and effectively chosen to maximize the noise margin between the "single match" condition and the "multiple match" condition.

3. A steady current flow across T9 (or T19) creates an IR drop during the detection phase. This intrinsically reduces the voltage swing on the MML (or RMML) and at the same time having a large sensing voltage at MMSP (or RMMSP). The energy saving comes from the faster sensing time because the pull-up current is utilized more efficient for the sensing part, instead of being wasted for charging up the entire MML or RMML.

In addition, the conditional modulation of the pull-down strength helps to suppress the nonlinearity of the current source. In particular, we are referring to the the channel-length modulation effect in the PMOS transistors (T1, T2, T11, and T12). This non-linearity mainly comes from the short-channel effects [27]. However, in memory circuits, we do not have the luxury of sizing MOS transistors with 2 microns in length for linear current biasing.

6.5.3 Circuit Operation

The operation of this MMD is similar to the conventional scheme described in Section 6.4.2. Therefore detail descriptions is not presented here. Figure 6.16 shows the timing diagram for the novel scheme when there are two matches in the TCAM array.

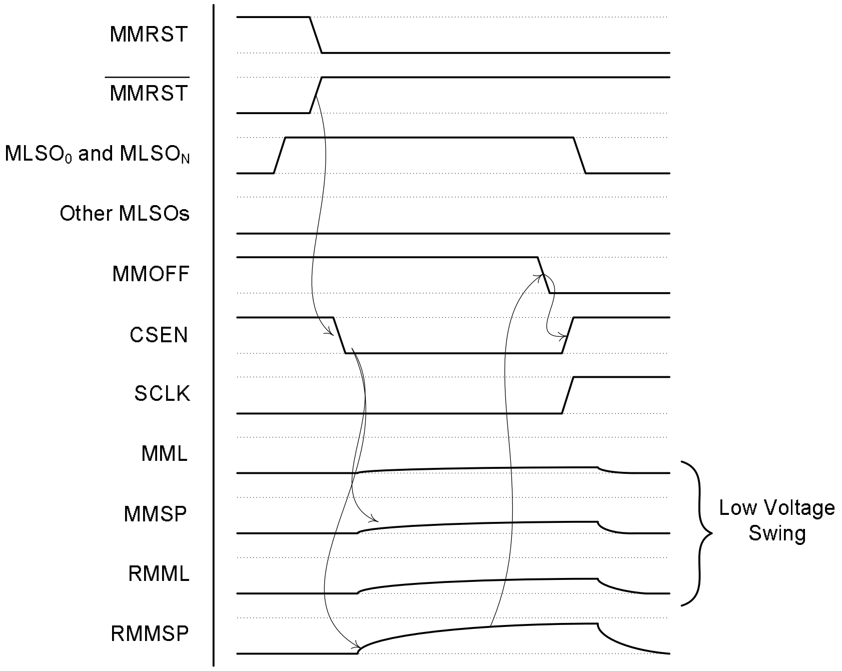


Figure 6.16: Timing Diagram for the Novel Multiple Match Detection Scheme

The MMSA starts sensing when the external control signal MMRST is switching from 1 to 0. This turns on the current sources and initiates the race. Since RMML is emulating a “single match” condition, and there are two matches on MML as specified, the voltage at RMMSP will increase at a faster rate. As a consequence, RMMSO will first switch from a “0” to a “1”. A sampling clock (SCLK) will be generated to sample and latch the outputs. The circuit will be reset to the idle state at the rising edge of MMRST.

Figure 6.17 shows the simulated waveforms for the "multiple match" condition. Notice that RMML and RMMSP are rising at the same constant rate at the beginning of the detection phase. However, as time goes on, the body effect is slowly causing a V_t shift, the resistance of the pass transistor (T19) is also shooting up when the drain voltage approaches V_G - V_t. In compared to the conventional scheme, the new design reduces the sensing time by "dt", as shown in Figure 6.16.

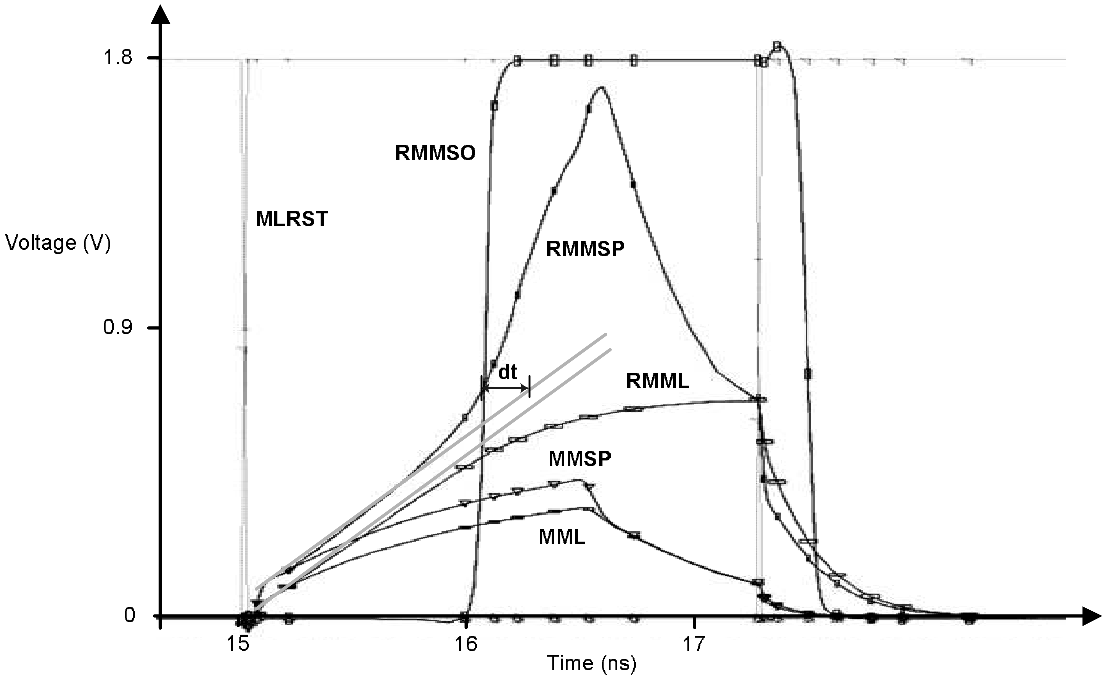


Figure 6.17: Simulated Waveforms for the Novel Multiple Match Detection Scheme

6.5.4 The Optimal Gate Voltage for Best Performance

As previously mentioned in Section 6.5.2, the gate voltage of T9 and T19 should be chosen wisely and effectively to maximize the benefit of the proposed scheme. Using the same "multiple match" example, a parametric analysis is performed and the results are illustrated in Figure 6.18. The goal of this analysis is to find the optimal gate voltage such that the circuit favors only one of the two sensing points.

Based on Figure 6.18 (a) and (b), it is clear that the voltage at RMMSP is rising at a faster

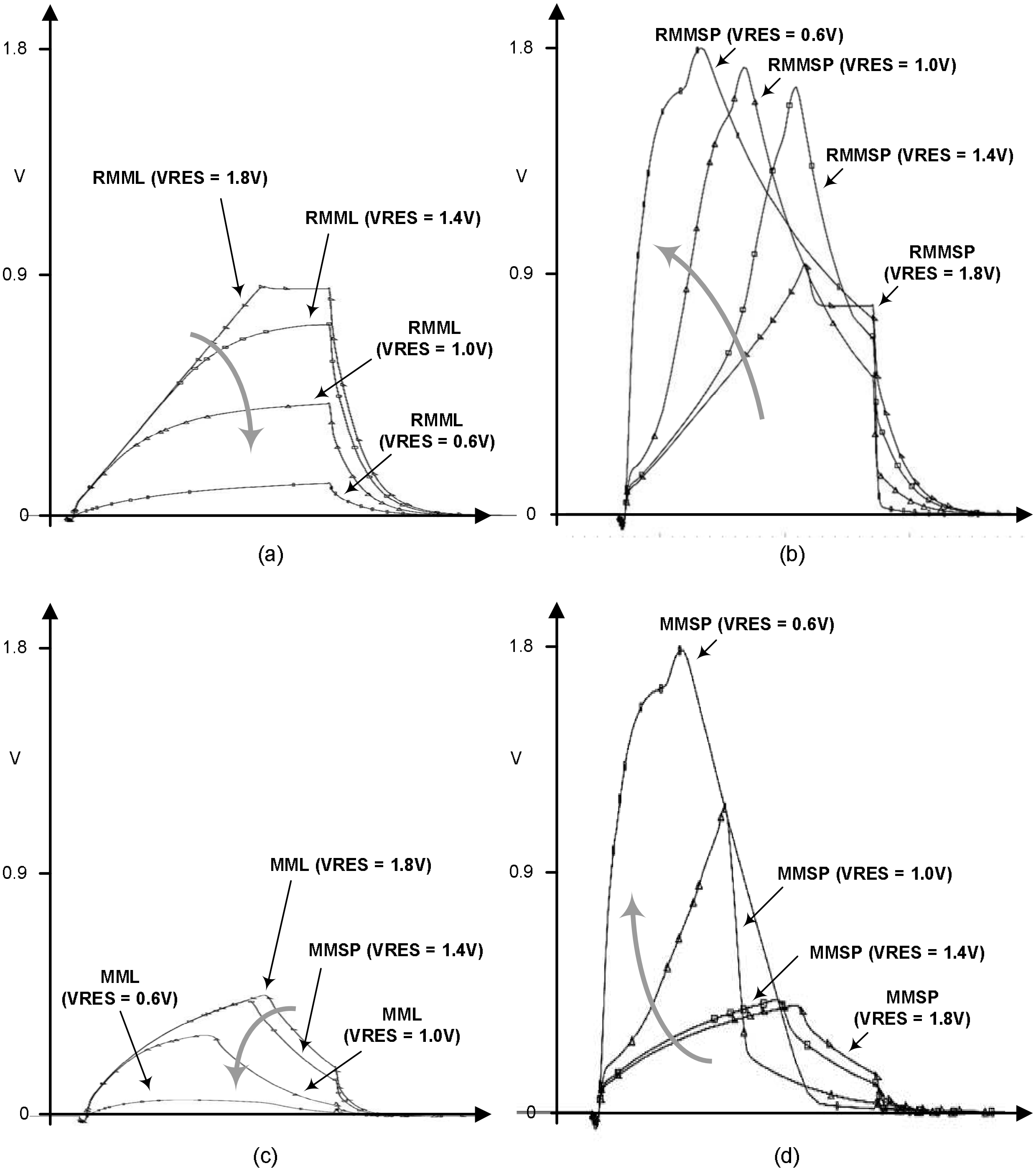


Figure 6.18: Parametric Analysis on the Robustness of the Proposed Scheme

rate when the gate voltage of T19 decreases. However, too much scaling on this gate voltage would cause "false" shoot-up at the other sensing point (MMSP). This phenomenon is shown in Figure

6.18 (c) and (d). For a 256-bit MMD with 1.8V supply voltage, the optimal gate voltage of T9 and T19 is found to be around 1.4V. This number has been confirmed in all process corners and in extreme temperature range.

6.5.5 Post-Layout Simulation Results

The novel multiple match detector, as described in the previous sections, has been designed and fabricated using TSMC 0.18 μm CMOS technology. The layout plot is shown in Figure 6.19.

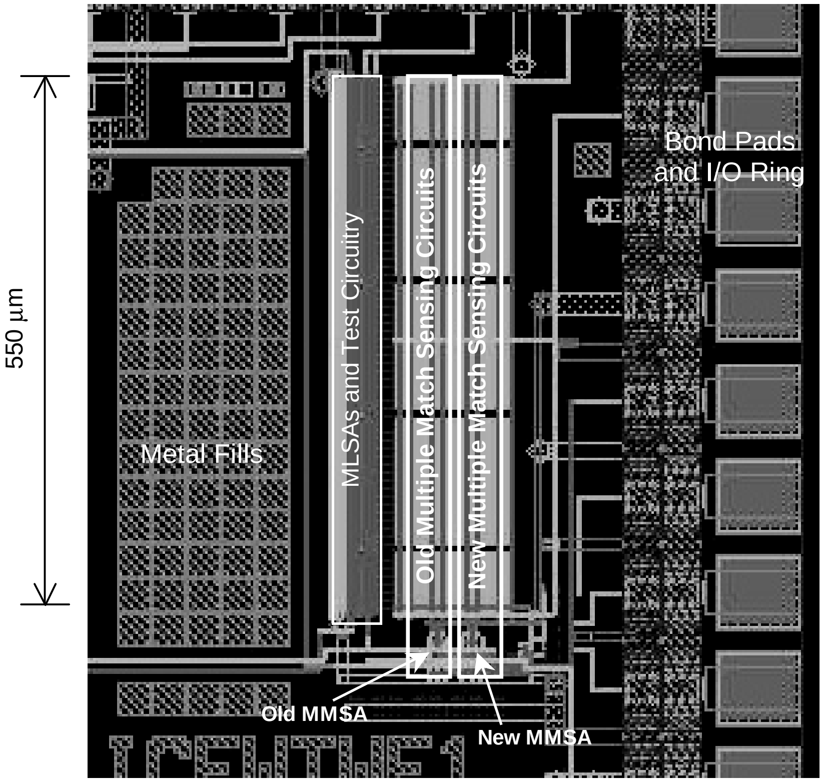


Figure 6.19: Layout Plot of a Test Chip with the Conventional and the Proposed "Current-Race" Implementations

The chip has been simulated using Cadence Spectre and Synopsys Nanosim with external bondwire parasitics and package parasitics. Table 6.3 and 6.4 show the expected worst-case results, for both the conventional scheme and the proposed scheme, in physical measurements. The post-layout simulation testbench includes the CMC (Canadian Microelectronics Corporation) customized bondwire models, the package models, and PCB trace and probe models.

	Delay (in ps)	Energy Consumption (Freq = 125 MHz)
No Match	784.6	0.8766 pJ / cycle
1 Match	1794.0	1.6222 pJ / cycle
2 Matches	1789.8	1.5604 pJ / cycle

Table 6.3: Post-Layout Simulation Results for the Conventional MMSA

	Delay (in ps)	Energy Consumption (Freq = 125 MHz)
No Match	716.76	0.8207 pJ / cycle
1 Match	1401.3 (21.89% Faster)	1.2817 pJ / cycle (21% Lower Energy)
2 Matches	1366.96	1.2715 pJ / cycle

Table 6.4: Post-Layout Simulation Results for the Proposed MMSA

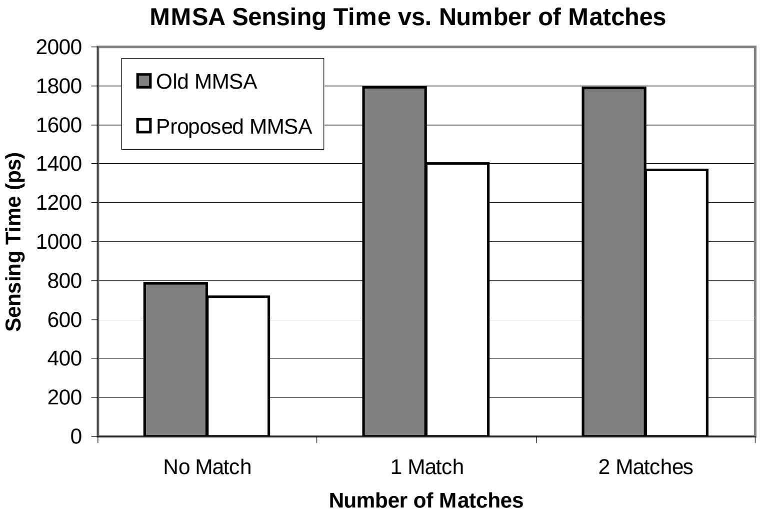


Figure 6.20: Post-Layout Simulation Results: Conventional MMSA vs Novel MMSA of this work

The reduction in overall energy consumption is hard to justify because it depends on the probability of ”no match”, ”1 match”, and so on. In terms of sensing speed, the new scheme is 22%

faster than the old scheme in post layout simulation results. Note that the overall sensing speed is determined by the worst-case delay.

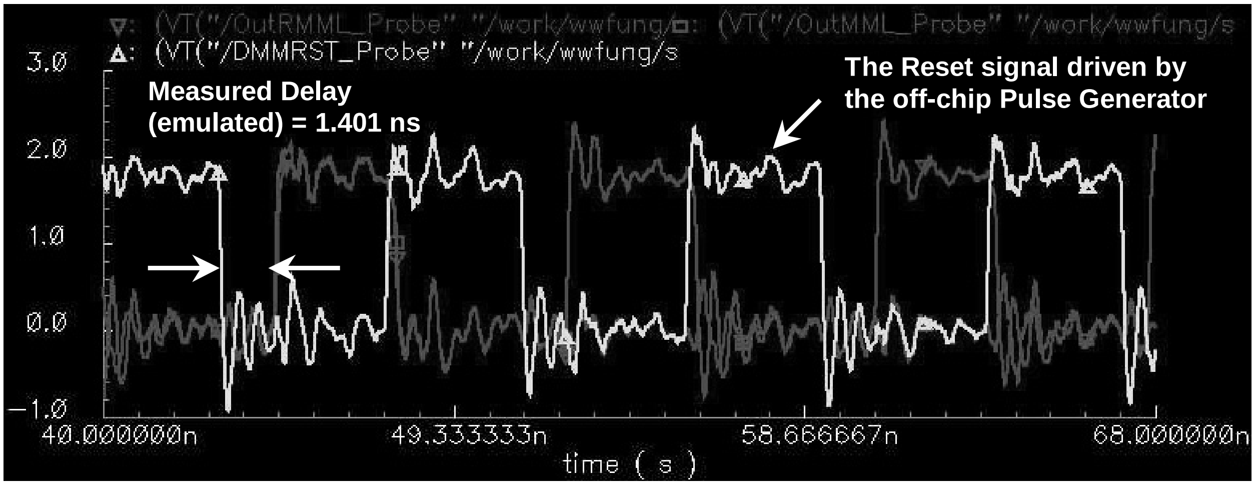


Figure 6.21: Post-Layout Simulated Waveforms with Chip Parasitics

Chapter 7

Next-Best Match Resolution

As mentioned in the earlier chapters, the external processor has the option to initiate a new TCAM search every cycle, or to inform the TCAM to output all match addresses in a "burst" mode. The decision is based on the instruction provided to the TCAM [29]. In this chapter, we will explore the circuit building blocks and methods to resolve the next-best match. These methods are generic enough to work with various styles of MLSA, MMR, and MMD.

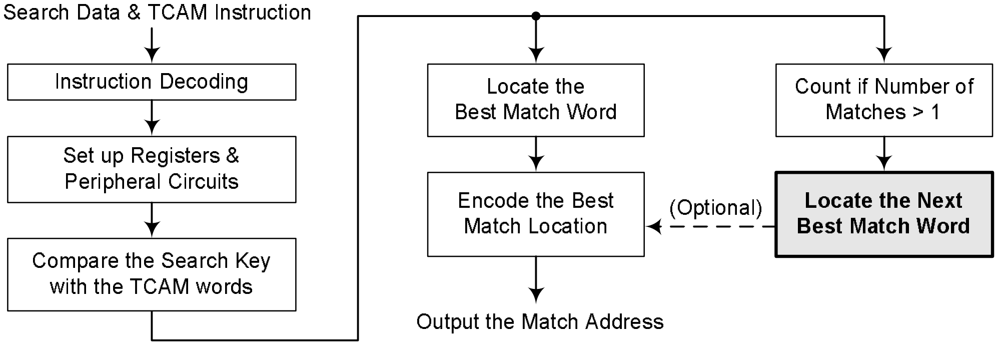

Search Data & TCAM Instruction
Instruction Decoding
Set up Registers & Peripheral Circuits
Compare the Search Key with the TCAM words
Locate the Best Match Word
Encode the Best Match Location
Output the Match Address
Count if Number of Matches > 1
Locate the Next Best Match Word
(Optional)

Figure 7.1: Next-Best Match Resolution in the Flow of a TCAM Search

7.1 The *Shift-and-Count* Approach

An extremely simple method of encoding and reading out all match addresses, from the highest priority to the lowest, is to employ an N-bit circular shift register in combination with an address counter. The architecture is illustrated in Figure 7.2. Notice that this approach requires no interaction with external MMR and MAE, because this method is itself capable of resolving multiple matches and encoding the addresses in prioritized order. The output of the MLSAs are first loaded into the shift registers in parallel. The address counter is initialized to value “0” prior to the multiple match readout. There is an asynchronous “Reset” signal connecting to, and only to the highest priority bit of the shift register. For now, assume it is “0” (not resetting).

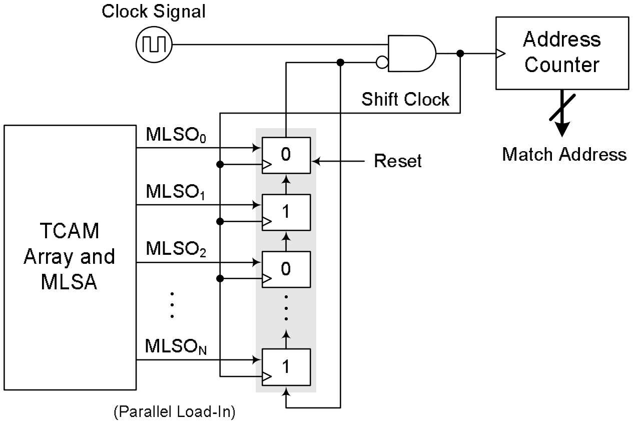


Figure 7.2: Next-Best Match Readout using Shift-Register and Address Counter

Figure 7.3 illustrates the mechanism of resolving all match addresses using the “Shift Register” approach. For simplicity, the Ternary CAM is assumed to have only 4 words, with only a 4-bit shift register and 2-bit encoded address space. Note that the input clock connecting the “address counter” and the “shift register” are conditionally gated by the temporal uppermost bit of the “shift register”. Upon the rising edge of the global clock, if the uppermost bit is a “0”, the output of the AND gate will switch from 0 → 1. This gated clock increments the address counter and shifts the registers until the first ”1” is reaching the uppermost bit, as shown in Figure 7.3(b). The value of the address counter represents the address of the best match in the search. Here, the encoded match address is 0x01.

To find the next-best match in the array, we can simply assert the “Reset” signal to “1”. This removes the highest priority “1” from the chain, and resumes the generation of conditional clock signals for the address counter and shift register. This process will stop again when the next ”1” reaching the uppermost bit of the shift register, as shown in Figure 7.3(c). The address of the

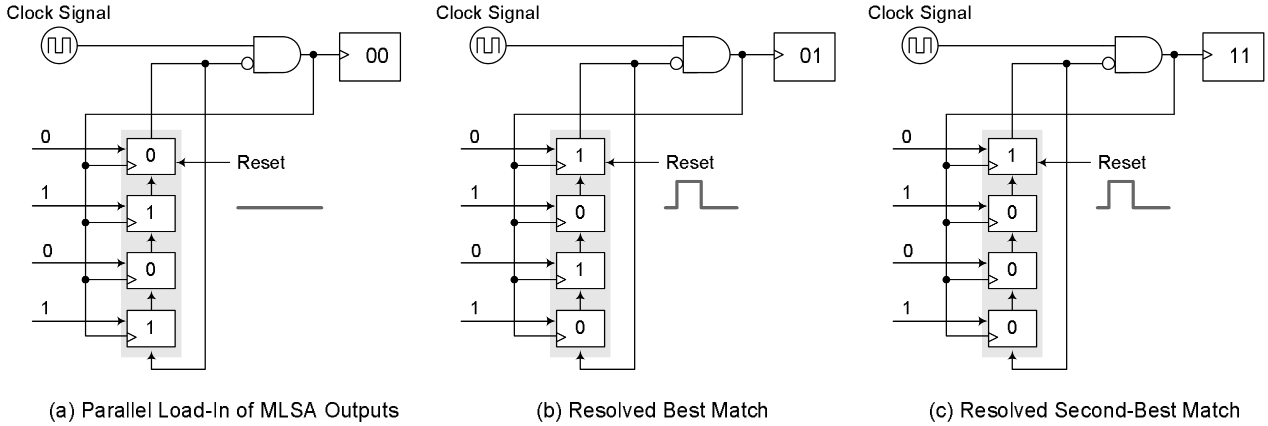


Figure 7.3: The Mechanism of the Shift Register Approach (N = 4)

next-best match can be read out from the address counter with no additional effort required. To find the addresses of all matches in the Ternary CAM, one can repeat the above steps as long as the total number of shifts ≤ N. The purpose of the circular feedback is to automatically reset all shift register outputs to 0.

This idea is simple and analogous to the integrating approach for Analog-to-Digital conversion [27], where both employ a counter to simplify the circuits. However, when N is large, the worst-case delay is N clock cycle. In addition, the capacitive loading at the output of the AND gate is huge. Inserting buffers to this node will harm the maximum frequency of the shift operation. These drawbacks unfortunately offset the advantages, in circuit simplicity and control signal management, offered by this approach. Although this scheme is never widely employed, some memory architectural techniques, such as multi-level segmentation, can make this scheme attractive and suitable for large Ternary CAMs.

7.2 The ***Latch-and-Reset*** Approach

In Chapter 4, we have studied the circuits for high-performance multiple match resolution. In Chapter 5, the methods for match address encoding are also disclosed. When the Multiple Match Resolver (MMR) inputs are directly connected to the Matchline Sense Amplifier (MLSA) outputs, the MMR is resolving the best match in the array. However, if the highest priority “1” is masked out, the resolved highest priority "1" (with respect to the MMR) is actually the second best match in the array. In other words, all we need is an additional interface between MLSA and MMR for masking the data pattern on the fly. This approach is denoted as "Latch-and-Reset" because MLSA outputs are first latched into registers, and the highest priority "1" is manually reset to "0" for resolving the next-best match in the CAM array. The concept of "Latch-and-Reset" is illustrated in Figure 7.4.

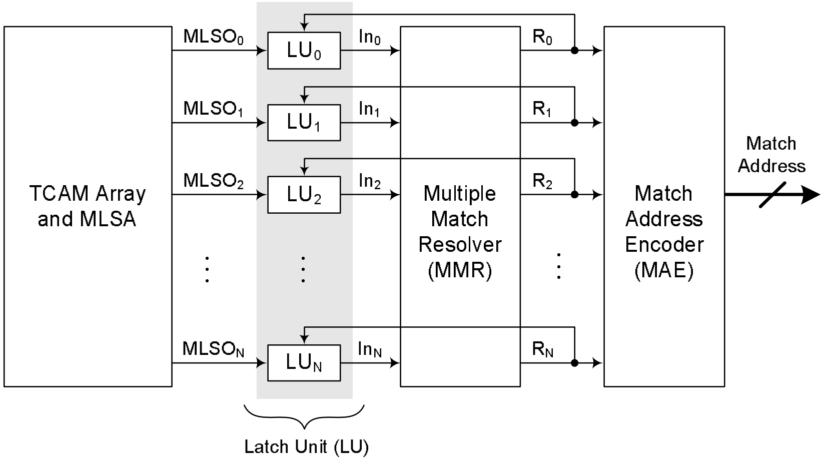


Figure 7.4: The Basic Architecture of the Latch-and-Reset Approach

Notice that only the MMR output corresponding to the highest priority input is asserted. This property suggests that a feedback path from the MMR outputs to the latch units (LUs) can conditionally reset the highest priority match, as shown in Figure 7.4. It is important that the LU reset is synchronous, otherwise race condition or false discharge may happen. The LU can be realized using many types of storage elements, such as SR, JK, and D flipflop or latch. Figure 7.5 shows an example of the "Latch-and-Reset" approach using clocked JK flip-flops [38]. The clocked version is used because otherwise resetting the best match would immediately induce resetting of the next best match etc.

According to the JK flip-flop logic table [10], if both J and K are "1", the output Q is toggling such that $Q_{n+1} = \overline{Q_n}$. This property has an important implication. In order to mask out all "match" signals that have already been processed, the MLSA outputs, which are connecting to the

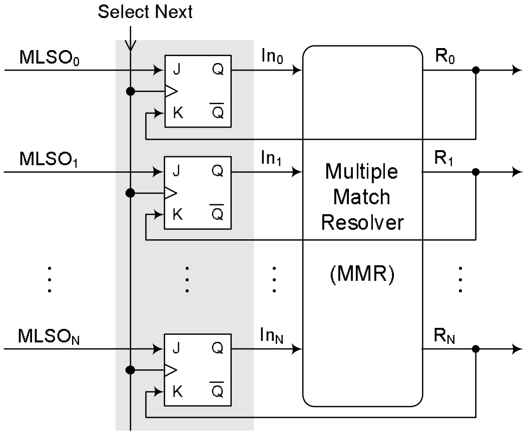


Figure 7.5: JK Flip-Flop Implementation of Latch-and-Reset

J input of the JK flipflop, must be precharged back to “0”. Otherwise, the masking is effective for only one clock cycle, and the zombie “match” signals would come alive again once the conceal is removed. This undesirable behavior makes the JK Implementation not efficient for pipelining. The MLSA outputs must be at “0” during the next-best match resolution. This implies that the MLSAs would be idle for at least one cycle before the initiation of a new search.

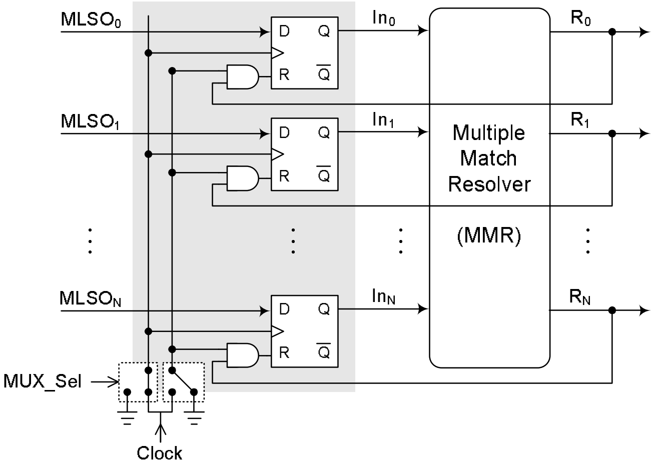


Figure 7.6: A Proposed Implementation of Latch-and-Reset using Dual Clocking

A proposed D flipflop implementation with conditionally reset using dual clocking strategy is shown in Figure 7.6. It offers the capability of resolving all match addresses in prioritized order, and yet pipeline-efficient. A similar implementation had also been disclosed in [16]. During the normal TCAM search operation, the global clock signal is driving clock input of the D flipflops. If the TCAM receives command from the external processor for resolving all match addresses, the global clock will be connecting to the AND gates for conditional resetting of the D flipflops. The next-best match addresses can be resolved, encoded, and read out upon the rising edge of the following cycles. Once the data is latched, the MLSA can start processing the next TCAM search. It requires no delay cycle between the current search and the next search.

Notice that the proposed scheme is generic enough for both dynamic-based or static-based MMR. One can also embed the AND gate into the design of the D flipflop for better abstraction. In addition, the proposed implementation is compatible to Design-For-Test (DFT). Although the test methodologies for MMR are not covered in this thesis, curious readers are welcome to look into [5] for further information.

7.3 The *Validity Bit* Approach

The Latch Units (LUs) in the "Latch-and-Reset" approach are sometimes expensive in terms of silicon area. Based on a different philosophy of tackling the problem, we can simply let the highest priority "match" inhibits all lower priority ones in the MMR. As soon as the MAE has encoded the best match address, this TCAM location (corresponding to the best match) will be marked in some way to show that it has been processed. Finding the next-best match, in this case, involves the initiation of a new search in the array. In compared to the "Latch-and-Reset" approach, this strategy is less power-efficient because not only the LUs and MMR are activated, the entire TCAM array and MLSAs must be active and running for each next-best match resolution.

To mark whether a match has been processed, one way is to pad an additional "Validity" bit into each TCAM word, and use this bit to intentionally cause a mis-match in the next search cycle. Another way is to disable the corresponding MLSA based on the status of this validity bit. Both ways can mask out the highest priority match(es) in the subsequent search cycles. It is obvious that disabling the MLSA for the masked words would result in lower power consumption. However, none of the TCAM vendors shows a sign of implementing such design into commercial TCAMs.

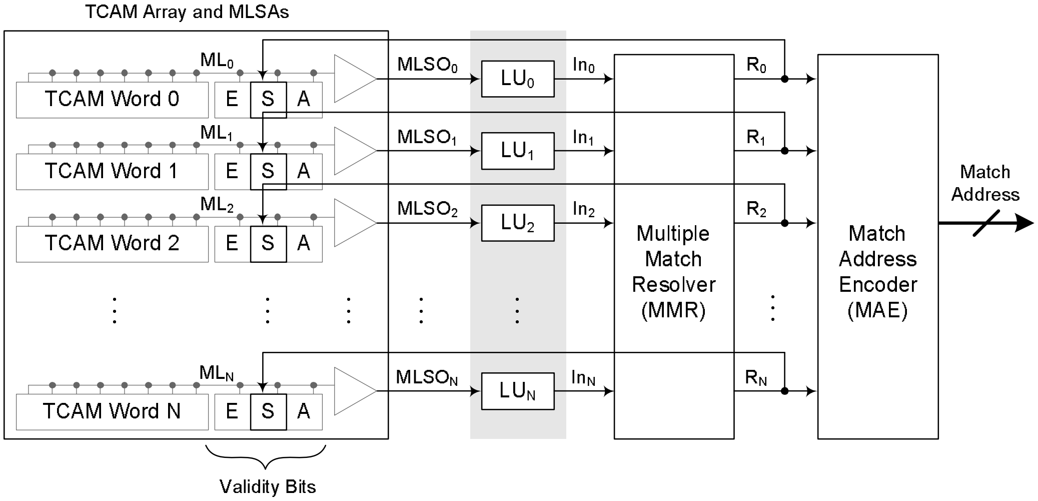


Figure 7.7: The Use of Validity Bits in Marking Processed “Match” Words

Figure 7.7 illustrates the concepts of the “Validity Bit” approach. Note that more than one validity bits can exist in the TCAM array. Each of them can serve a different role in enhancing the search operations. Some commonly used validity bits are “Empty” (E), “Skip” (S), and “Aged” (A) etc. Although the “Validity Bit” approach is more power consuming, it offers more flexibility for table management and partitioning for modern high-speed lookup applications [39]. The external processor can also intentionally mask out some specific TCAM words in a search. It is believed that this approach is employed in a number of commercial TCAM designs [39]. Figure 7.8 shows the procedure for locating all matches a TCAM.

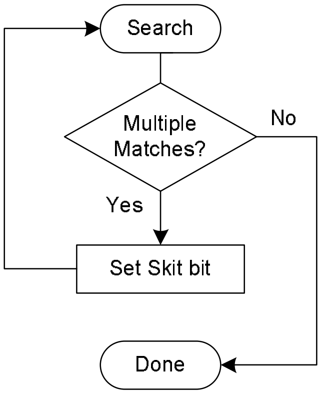


Figure 7.8: Procedure for Locating Multiple Matches using Validity bits (adapted from [39])

7.4 Inter-Block Considerations

In a high-density TCAM, the MMR and MAE are usually partitioned into smaller blocks as depicted in Figure 7.9. Any multiple match resolution and readout method described in this chapter can be applied here to locate the match locations in the array.

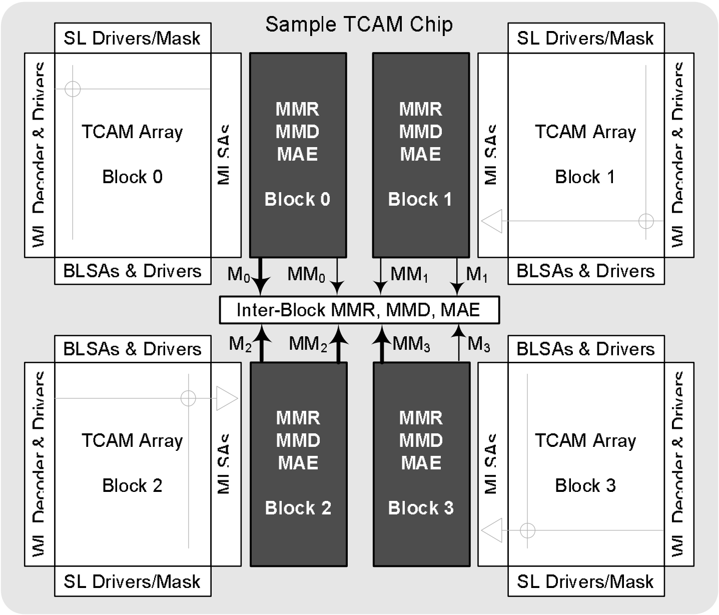


Figure 7.9: Chip-level Architecture of Multiple Match Readout

In the inter-block level, there are actually a number of ways to reduce power consumption. Although they are trivial, they are present here for completeness. In our example (refers to Figure 7.9), assuming that after a TCAM search, there is 1 match in block 0, multiple matches in block 2, and 1 match in block 3, as indicated by the “M” and “MM” signals. To resolve the second-best “match”, we can first mask out the “match” in block 0. Since there is only 1 match in block 0, it is redundant to activate the MLSAs in block 0 during the next-best search cycle. We can simply mask out this block entirely to save power.

Furthermore, notice that there are more than one matches in block 2, as indicated by MM_2. One way to efficiently locate the third-best “match” is to activate only the MLSAs in block 2. The rest of the TCAM chip can be in sleep mode or idle mode for further power saving. It is important that the Inter-block MMR and the global control logic have the intelligence to activate only the relevant blocks in a search. These type of block-level masking can save a tremendous amount of power if designed properly.

Chapter 8

Concluding Remarks

8.1 Conclusions

A number of innovative circuit ideas on multiple match resolution and detection for TCAMs are proposed in this thesis. They offer low power, high performance, small area overhead, good scalability, and ease of pitch-matching to the TCAM array. The contributions and "key messages" of each chapter are summarized in the following.

- A token-based MMR has a lower power profile in compared to an inhibit-based MMR. Besides, the idea of replacing the normal-V_t pass-transistors by low-V_t devices is not recommended. The leakage can cause "false" discharging in the MMR cell. This can lead to a situation where the supposedly resolved highest priority match never appear at the MMR output.

- The size of the MMR macro-block is limited by (i) the capacitance on the BE line, and (ii) the RC delay in the pass-transistor chain. The novel MMR presented in this thesis has low BE line capacitance, and a bypassing scheme to reduce the worst-case RC delay. A promising method for embedding the bypassing circuits into the MMR cell array has also been introduced.

- The two unique properties of MAE should be considered in the design of MAE in TCAM. These properties can help to relax the constraints and to save power consumption. In addition, the MAEs should be interleaved, or laid out using the proposed share wordline approach (to save power and area)

- The all-digital MMDs are complex and require extremely wide fan-in logic gates. The mixed-signal approach consumes lower power and has lower area overhead in compared to the all-digital detection scheme.

- The idea to increase the speed of the MMD is to give up the"excessive" robustness in the circuit. Examples include the addition of a "shielding" resistor or a pass-transistor as a common-gate stage.

8.2 Future Research and Recommendations

The multiple match resolution methods presented in this thesis are based on "hard priority", which means the priority of the TCAM word is fixed and not programmable. The programmable priority, or sometimes called the "Soft Priority", is an area for future research to further enhance the versatility of TCAMs.

The novel multiple match detection circuit has been implemented and fabricated in TSMC 0.18 μm CMOS technology. However, additional research can be done to analyze (i) the V_{RES} to V_{dd} ratio for optimal performance, and (ii) the benefits of this scheme when CMOS scales beyond 0.1 μm.

Bibliography

[1] M. Ichiriu. “High Performance Layer 3 Forwarding: The Need for Dedicated Hardware Solutions”. *NetLogic Microsystems, White Paper*, 2000.

[2] D. Paul. “CAM Whitepaper: Where is the Tecnology and Market Going?”. *Semiconductor Insights Inc., White Paper*, April 2003.

[3] J. Pereira. “Moving Classification and Forwarding to OC-768”. *NetLogic Microsystems, White Paper*, 2002.

[4] M. Wirth. “The Next Generation of Content Addressable Memories”. *MOSAID Technologies Inc.*, March 2003.

[5] N. Mohan, W. Fung, D. Wright, and M. Sachdev. “Design Techniques and Test Methodology for Low-power TCAMs”. *IEEE Transactions on VLSI*, (To be published).

[6] N. Mohan and M. Sachdev. “Low Power Dual Match-Line Ternary Content Addressable Memory”. *Proc. of Int. Symp. on Circuits and Systems (ISCAS) 2004*, pages pp. 633 – 636, May 2004.

[7] T. Hamamoto Y. Murai T. Kobayashi M. Yamada T. Yamagata, M. Mihara and H. Ozaki. “A 288-kb Fully Parallel Content Addressable Memory Using a Stacked-Capacitor Cell Structure”. *IEEE Journal of Solid-State Circuits*, Vol. 27(12):pp. 1927 – 1933, December 1992.

[8] Z. Pfeffer B. Gamache and S. Khatri. “A Fast Ternary CAM Design for IP Networking Appications”. *Proceedings of IEEE International Conference on Computer Communications and Networks (IC3N) 2003*, pages pp. 434 – 439, 2003.

[9] M. Kuroiwa A. Amo A. Hachisuka H. J. Mattausch T. Koide S. Soeda K. Dosaka K. Arimoto H. Noda, K. Inoue. "A 143MHz 1.1W 4.5Mb Dynamic TCAM with Hierarchical Searching and Shift Redundancy Architecture". *Proceedings of IEEE ISSCC 2004*, Vol. 47(12):pp. 208 – 209, Feb 2004.

[10] M. Morris Mano. *"Digital Design"*. Prentice Hall, second edition, 1990.

[11] J. Delgado-Frias and J. Nyathi. "A High-Performance Encoder With Priority Lookahead". *IEEE Transactions on Circuits and Systems I*, Vol. 47(9):pp. 1390 – 1393, September 2000.

[12] "Data Sheet for MC14532B". *Motorola Seconductor Technical Data Sheets*, January 1994.

[13] "Ayama 10000 Network Search Engine Preliminary Data Sheet". *Cypress Semiconductor*, July 2004.

[14] J. Wang and C. Huang. "High-Speed and Low-Power CMOS Priority Encoders". *IEEE Journal of Solid-State Circuits*, Vol. 35(10):pp. 1511 – 1514, October 2000.

[15] Jan M. Rabaey. *"Digital Integrated Circuits: A Design Perspective"*. Prentice Hall, 1st edition, 1996.

[16] J. Wade and C. Sodini. "A Ternary Content Addressable Search Engine". *IEEE Journal of Solid-State Circuits*, Vol. 24(4):pp. 1003 – 1013, August 1989.

[17] Y. Huang C. Huang, J. Wang. "Design of High-Performance CMOS Priority Encoders and Incrementer/Decrementers Using Multilevel Lookahead and Multilevel Folding Techniques". *IEEE Journal of Solid-State Circuits*, Vol. 37(1):pp. 63 – 76, January 2002.

[18] Y. Hirota T. Satoh T. Miwa, H. Yamada and H. Hara. "A 1-Mb 2 Tr/b Nonvolatile CAM Based on Flash Memory Technologies". *IEEE Journal of Solid-State Circuits*, Vol. 31(11):pp. 1601 – 1609, November 1996.

[19] Y. Nishimichi H Kudoh H. Kadota, J. Miyake and K. Kagawa. "An 8-kbit Content-Addressable and Reentrant Memory". *IEEE Journal of Solid-State Circuits*, Vol. 20(5):pp. 951 – 957, October 1985.

[20] J. Eneland H. Bergh and L. Lundstrom. "A Fault-Tolerant Associative Memory with High-Speed Operation". *IEEE Journal of Solid-State Circuits*, Vol. 25(4):pp. 912 – 919, August 1990.

[21] R. Foss and A. Roth. "Priority Encoder Circuit and Method for Content Addressable Memory". *Canadian Patent 2,365,891*, MOSAID Technologies Inc., April 30, 2003.

[22] Ken Martin. *"Digital Integrated Circuit Design"*. Oxford University Press, University of Toronto, 1st edition, 1999.

[23] K. Schultz and P. Gulak. "Fully Parallel Integrated CAM/RAM Using Preclassification to Enable large Capacities". *IEEE Journal of Solid-State Circuits*, 31(5):pp. 689 – 699, May 1996.

[24] M. Miller and M. Baumann. "Content Addressable Memory Array Having Flexible Priority Support". *U.S. Patent 2003/0005146*, Integrated Device Technology Inc., January 2, 2003.

[25] D. Buss. "When MOSFET Switches Become MOSFET Dimmers". *Visual Supplement in the Proceedings of IEEE International Solid-State Circuit Conference (ISSCC) 2002*, pages pp. 571 – 573, 2002.

[26] J. Greason S. Thompson, I. Young and M. Bohr. "Dual Threshold Voltages and Substrate Bias: Keys to High Performance, Low Power, 0.1 μm Logic Designs". *1997 Symposium on VLSI Techonology Digest of Technical Papers*, pages pp. 69 – 70, 1997.

[27] David Johns and Ken Martin. *"Analog Integrated Circuit Design"*. John Wiley & Sons, University of Toronto, 1st edition, 1997.

[28] T. Chandler I. Arsovski and A. Sheikholeslami. "A Ternary Content-Addressable Memory (TCAM) Based on 4T Static Storage and Including a Current-Race Sensing Scheme". *IEEE Journal of Solid-State Circuits*, 38(1):pp. 155 – 158, January 2003.

[29] A. Roth D. Foss R. McKenzie and D. Perry. "Advanced Ternary CAM Circuits on 0.13μm Logic Process Technology". *Proceedings of IEEE Custom Integrated Circuit Conference (CICC) 2004*, October 2004.

[30] Z. Regev. “Multi-Match Detection Circuit for Use with Content Addressable Memories”. *U.S. Patent 6,707,694 B2*, Micron Technology Inc., March 16, 2004.

[31] E. Voelked S. Sywyk and S. Chu. “Multiple Signal Detection Circuit”. *U.S. Patent 6,195,277 B1*, Lara Technology Inc., February 27, 2001.

[32] C. Jiang and A. Roth. “Circuit and Method for Detecting Multiple Matches in a Content Addressable Memory”. *U.S. Patent 2003/0145178*, July 31, 2003.

[33] A. Roth. “Method and Apparatus for Performing Variable Word Width Searches in a Content Addressable Memory”. *U.S. Patent 2003/0223259*, MOSAID Technologies Inc., December 4, 2003.

[34] R. Bosnyak and M. Santoro. “Method and Appartus for Detecting Multiple Address Matches in a Content Addressable Memory”. *U.S. Patent 5,446,686*, Sun Microsystems Inc., August 29, 1995.

[35] A. Ahmed and V. Lines. “Circuits and Methods for Multiple Match Detection in Content Address Memories”. *U.S. Patent 6,667,924 B2*, MOSAID Technologies Inc., December 23, 2003.

[36] S. Ma and P. Ma. “Multiple Match Detection Circuit and Method”. *Canadian Patent 2,310,295*, MOSAID Technologies Inc., November 30, 2001.

[37] Bahzad Razavi. *“Design of Analog CMOS Integrated Circuits”*. McGraw-Hill, University of California, Los Angelas, 1st edition, 2001.

[38] T. Kohonen. *“Content Addressable Memories”*. Springer-Verlag, second edition, 1980.

[39] “Intra-Device Configurability in Network Search Engine”. *NetLogic Microsystems, Application Note NCS012*, January 2001.