

# Chapter 4

## Low Power Schemes for Energy-Efficient TCAM Design

CAMs are popular in network routers for packet forwarding and packet classification in the recent years. Network routers forward data packets from an incoming port to an outgoing port, using an address-lookup function. Ternary cells, in addition, store an “X” value. The “X” value is a don’t-care that represents both “0” and “1”, allowing a wildcard operation. Wildcard operation means that an “X” value stored in a cell causes a match regardless of the input bit. This is a feature used in packet forwarding in Internet routers. With the evolving nature of the Internet, the current Internet Protocol version 4 (IPv4) will be rapidly falling into disuse because of its limited address space, lack of needed functionality, and inadequate security features. With the manifest of the shortcomings of the existing IP, a new protocol, known as IPv6 (IP version 6), has been defined to ultimately replace IPv4. The addresses in the new Internet protocol are 144 bit long, whereas they are 32 bit long in the current IPv4-based Internet. Given the exponential growth of the Internet, and the advent of terabit per second transmission media, Internet routers are expected to make routing decisions within nanoseconds; the Internet will require high-speed forwarding engines. An important factor of these forwarding engines in an IPv6-based Internet is the speed and cost of doing routing table lookup on hierarchical 144-bit IPv6 addresses.

An energy-efficient scheme of ternary content addressable memory, which reduce search power significantly while maintaining good search speed was proposed previously [92]. Several low power schemes will be discussed in this chapter. All the novel low power TCAM design is based on these schemes.

In this chapter, the concept of IPV6 address look up will be discussed in detail first. Next the several low power techniques of previously proposed energy-efficient

TCAM will be introduced briefly, including butterfly match-line scheme (Section 4.3.1), XOR-based conditional keeper (Section 4.3.2), don't care based power gating match-line scheme (Section 4.3.3), don't care based hierarchy search line scheme (Section 4.3.4). The overall architecture will be presented in the end of this chapter.

## 4.1 IPv6 Addressing Architecture

IPv6 addresses are 128-bit identifiers for interfaces and sets of interfaces. There are three types of addresses:

**Unicast:** An identifier for a single interface. A packet sent to a unicast address is delivered to the interface identified by that address.

**Anycast:** An identifier for a set of interfaces (typically belonging to different nodes). A packet sent to an anycast address is delivered to one of the interfaces identified by that address (the "nearest" one, according to the routing protocols' measure of distance).

**Multicast:** An identifier for a set of interfaces (typically belonging to different nodes). A packet sent to a multicast address is delivered to all interfaces identified by that address.

The type of an IPv6 address is identified by the high-order bits of the address, as shown in Table 4.1:

Table 4.1 IPv6 address type identification

Address type	Binary prefix	IPv6 notation
Unspecified	00...0 (128 bits)	::/128
Loopback	00...1 (128 bits)	::1/128
Multicast	11111111	FF00::/8
Link-local unicast	1111111010	FE80::/10
Site-local unicast	1111111011	FEC0::/10
Global unicast	(everything else)	

IPv6 unicast addresses are aggregable with prefixes of arbitrary bit-length similar to IPv4 addresses under Classless Interdomain Routing. There are several types of unicast addresses in IPv6, in particular global unicast, site-local unicast, and link-local unicast. There are also some special-purpose subtypes of global unicast, such as IPv6 addresses with embedded IPv4 addresses or encoded NSAP addresses. Additional

address types or subtypes can be defined in the future.

### 4.1.1 Unspecified Unicast Addresses

The address 0:0:0:0:0:0:0:0 is called the unspecified address. It must never be assigned to any node. It indicates the absence of an address. One example of its use is in the Source Address field of any IPv6 packets sent by an initializing host before it has learned its own address. The unspecified address must not be used as the destination address of IPv6 packets or in IPv6 Routing Headers. An IPv6 packet with a source address of unspecified must never be forwarded by an IPv6 router.

### 4.1.2 Loopback Unicast Addresses

The address 0:0:0:0:0:0:0:0 is called the unspecified address. It must never be assigned to any node. It indicates the absence of an address. One example of its use is in the Source Address field of any IPv6 packets sent by an initializing host before it has learned its own address. The unspecified address must not be used as the destination address of IPv6 packets or in IPv6 Routing Headers. An IPv6 packet with a source address of unspecified must never be forwarded by an IPv6 router.

### 4.1.3 Global Unicast Addresses

The general format for IPv6 global unicast addresses is shown in Fig. 4.1, where the global routing prefix is a (typically hierarchically-structured) value assigned to a site (a cluster of subnets/links), the subnet ID is an identifier of a link within the site, and the interface ID is used to identify interfaces on a link. All global unicast addresses other than those that start with binary 000 have a 64-bit interface ID field (i.e.,  $n + m = 64$ ). Global unicast addresses that start with binary 000 have no such constraint on the size or structure of the interface ID field. Examples of global unicast addresses that start with binary 000 are the IPv6 address with embedded IPv4 addresses and the IPv6 address containing encoded NSAP addresses specified in [NSAP]. An example of global addresses starting with a binary value other than 000 (and therefore having a 64-bit interface ID field) can be found in [AGGR].

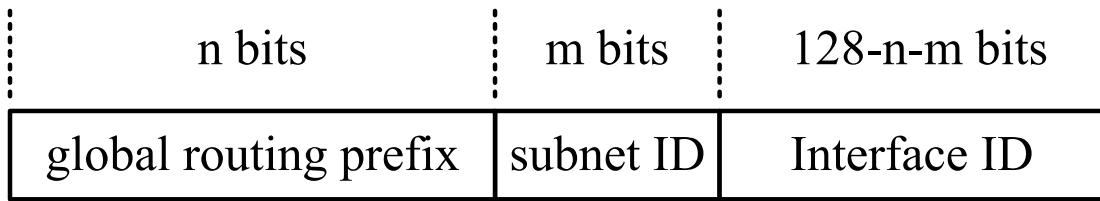
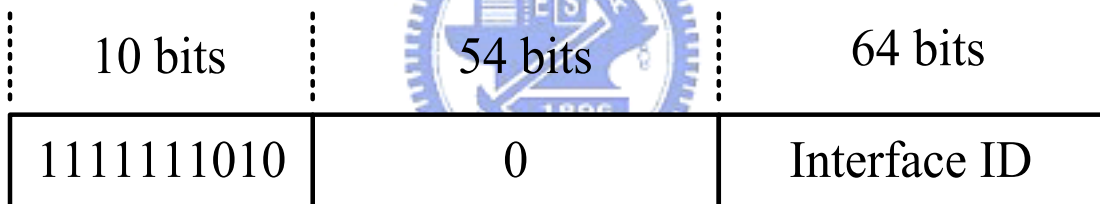


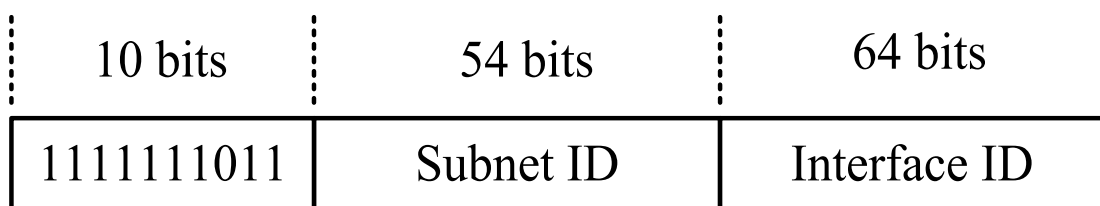
Fig. 4.1 The general format for IPv6 global unicast addresses

#### 4.1.4 Local-Use IPv6 Unicast Addresses

There are two types of local-use unicast addresses defined. These are Link-Local and Site-Local. The Link-Local is for use on a single link and the Site-Local is for use in a single site. Fig. 4.2 (a) shows the format of Link-Local addresses. Link-Local addresses are designed to be used for addressing on a single link for purposes such as automatic address configuration, neighbor discovery, or when no routers are present. Routers must not forward any packets with link-local source or destination addresses to other links. Fig. 4.2 (b) shows the format of Site-Local addresses.



(a)



(b)

Fig. 4.2 The format of local-use IPv6 unicast addresses

## 4.2 IP Address Lookup

The primary role of routers is to forward packets toward their final destination. To this purpose, a router must decide for each incoming packet where to send it next.

More exactly, the forwarding decision consists of finding the address of the next-hop router as well as the egress port through which the packet should be sent. This forwarding information is stored in a forwarding table computed by the router based on the information gathered by routing protocols. To consult the forwarding table, the router uses the packet's destination address as a key; this operation is called address lookup. Once the forwarding information is retrieved, the router can transfer the packet from the incoming link to the appropriate outgoing link, in a process called switching.

One of the fundamental objectives of the Internet Protocol is to interconnect networks, so routing on a network basis was a natural choice (rather than routing on a host basis). Thus, the IP address scheme initially used a simple two-level hierarchy. This hierarchy is reflected in the fact that an IP address consists of two parts, a network part and a host part. The network part identifies the network to which a host is attached, and thus all hosts attached to the same network will be identical in the network part of their IP addresses. Since the network part corresponds to the first bit of the IP address, it is called the address prefix. With a two-level hierarchy, IP routers forwarded packets based only on the network part, until packets reached the destination network. As a result, a forwarding table only needed to store a single entry to forward packets to all the hosts attached to the same network. This technique is called address aggregation and allows prefixes used to represent a group of addresses.

To allow the use of the IP address space more efficiently and to slow down the growth of the backbone forwarding tables, a new scheme called classless inter-domain routing (CIDR) was introduced. CIDR allows finer granularity in the prefix lengths to achieve such advantage. That is why prefixes can be of arbitrary length. To address the problem of forwarding table explosion, CIDR allows address aggregation at several levels. The idea is that the allocation of addresses has a topological significance. Then, we can recursively aggregate addresses at various points within the hierarchy of the Internet's topology. As a result, backbone routers do not maintain forwarding information at the network level, but at the level of arbitrary aggregates of networks. Thus, recursive address aggregation reduces the number of entries in the forwarding table of backbone routers. A router must find the most specific match, which is the longest matching prefix. In summary, the address lookup problem in routers requires searching the forwarding table for the longest prefix that matches the

destination address of a packet.

While CIDR allows the size of the forwarding tables to be reduced, the address lookup problem now becomes more complex. With CIDR, the destination prefixes in the forwarding tables have arbitrary lengths and no longer correspond to the network part since they are the result of an arbitrary number of network aggregations. Therefore, when using CIDR, the search in a forwarding table can no longer be performed by exact matching because the length of the prefix cannot be derived from the address itself. As a result, determining the longest matching prefix involves not only comparing the bit pattern itself, but also finding the appropriate length.

When the IPv6 routing protocol is introduced where the address length is 128 bits, the problem of routing millions of communication packets every second becomes a labyrinth. Especially, the most difficult point is based on longest prefix matching. In these circumstances, where IP routing tables are expanding in both the dimensions, i.e., address length and number of prefixes, the routing mechanisms developed should be capable of providing the throughput demand from high-speed transmission links.

Ternary content-addressable memory (TCAM) is a popular hardware device that can be used to perform fast lookups. TCAM adopts parallel technique and can reach  $O(1)$  lookup complexity. Compare with other lookup technologies, the advantages of TCAM are high lookup speed and simple operation. However, TCAM has three explicit disadvantages. First, with the same storage capacity, TCAM chips are much more expensive than SDRAM or DRAM chips. Second, TCAM chips perform key word comparisons parallel and power consumption is high. Third, to perform longest prefix matching lookups, prefixes must be arranged in order inside TCAM chips. Incremental prefix updates are rather complex and difficult. A packet routing based on longest prefix matching mechanism is shown in Fig. 4.3.

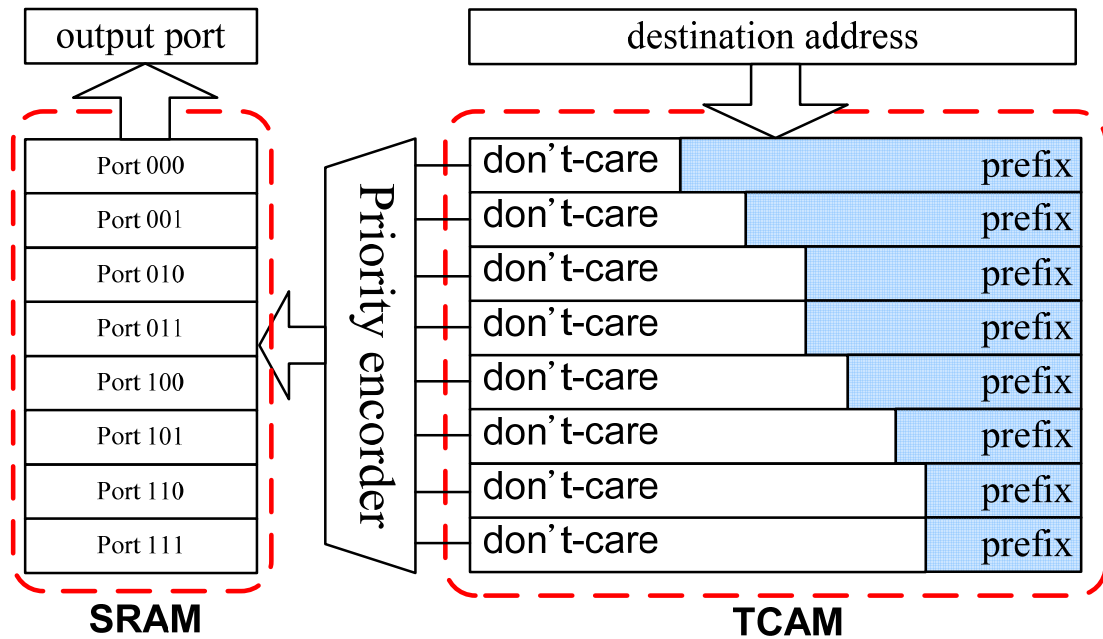


Fig. 4.3 Packet routing based on longest prefix matching mechanism

### 4.3 IPv6 prefix length distribution in the router

Don't-care based low power technique is based on properties of IPv6 addressing lookup table. In order to get more accuracy simulation result, we need to know the properties of routing table in current internet. Table 4.2 shows the prefix length distribution in the router of 6Bone. Obviously, we can observe two points from Table 4.2. First, the distributions of the prefix length equal to 32, 48, 35, 24, and 28 are approximately 86.21%, 4.76%, 3.78%, 1.31% and 1.31%. Second, there is no prefix length longer than 64 that is interface ID. The network address prefix distribution of IPv6 is shown in Fig. 4.4. According to this statistics, most TCAM cells in the routing table are don't-care. Both power-gating and hierarchical search-line based on don't-care are efficiently if there are more don't-care TCAM cells.

Table 4.2 Prefix length distribution in the router of 6Bone

Prefix length	Route Entries	%
16	1	0.16
19	1	0.16
20	3	0.49
21	2	0.33

24	8	1.31
27	1	0.16
28	8	1.31
29	1	0.16
30	2	0.33
32	525	86.21
33	3	0.49
34	1	0.16
35	23	3.78
36	1	0.16
48	29	4.76

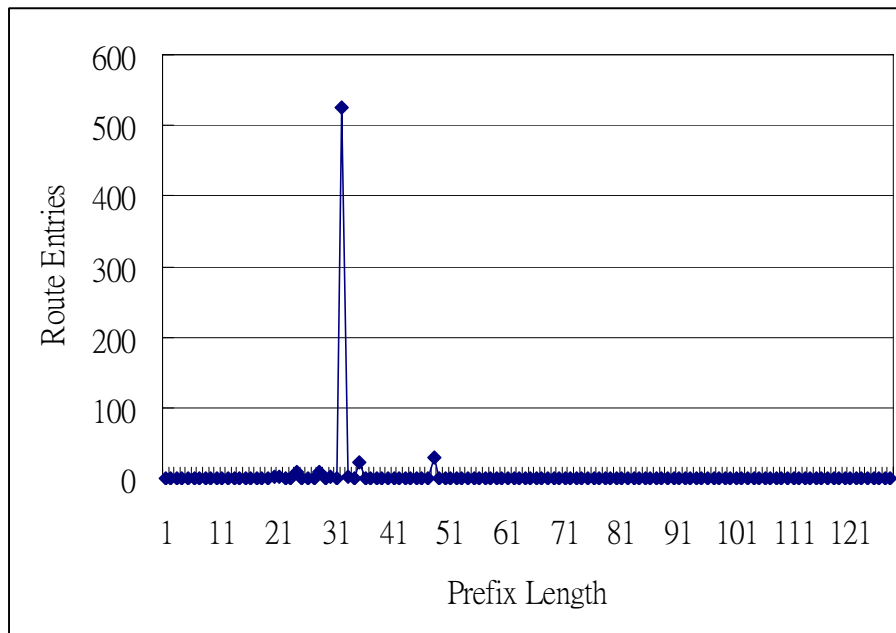


Fig. 4.4 Network address prefix distribution of IPv6

## 4.4 Energy Efficient Ternary Content-Addressable Memory

The background architecture of proposed ternary content-addressable memory (TCAM), which is proposed in [92], will be introduced in this section. This low-power and high-speed TCAM design is realized with several low power technique including butterfly match-line scheme, XOR based conditional keeper,



don't care based power gating match-line scheme, don't care based hierarchy search line scheme. The overall architecture is given in the end of this section. In addition, we transferred the energy efficient TCAM array to 65nm CMOS technology based on Berkeley Predicted Technology Model (BPTM) and extend to 144 bits.

#### 4.4.1 Butterfly Match-line Scheme

A PF-CDPD AND-type match-line scheme [89] is shown as Fig. 4.5. The match-line is divided into two parts, and each part consists of eleven CAM segments. The critical path of this scheme includes twelve CAM segments and one AND-gate delay for 128 bits. The butterfly match-line scheme is proposed as Fig. 4.6 (a). The degree of the parallelism is increased twice as the conventional AND-type match-line scheme [89]. Therefore, the critical path is reduced from twelve segments to six stages. Furthermore, in order to reduce the power consumption, we use the butterfly connection style by intersecting the connections between two independent match-lines. When one of the CAM segment is mismatch, the proposed match-line scheme will turn off more CAM segments by the butterfly connection style. The butterfly match-line scheme increases the dependence between the four parallel match-lines. All the search operations behind the mismatched segment would be terminated. Fig. 4.6 (b) shows two TCAM segments which are connected by the butterfly connection style. The NOR-gate is used to connect two TCAM segments, and the controlled signal of next stage is generated by the NOR-gate output.

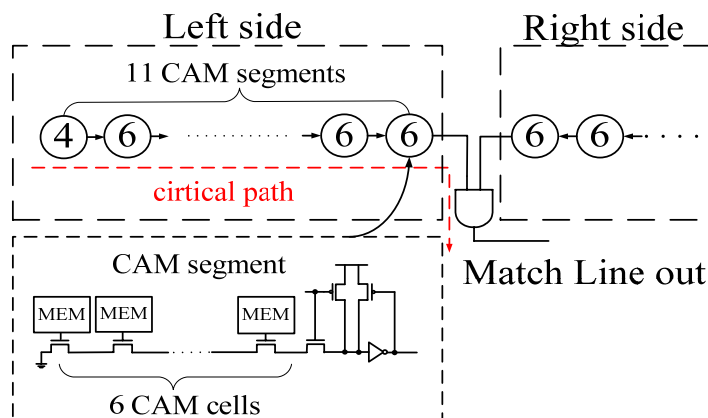


Fig.4.5. PF-CDPD and-type match-line scheme

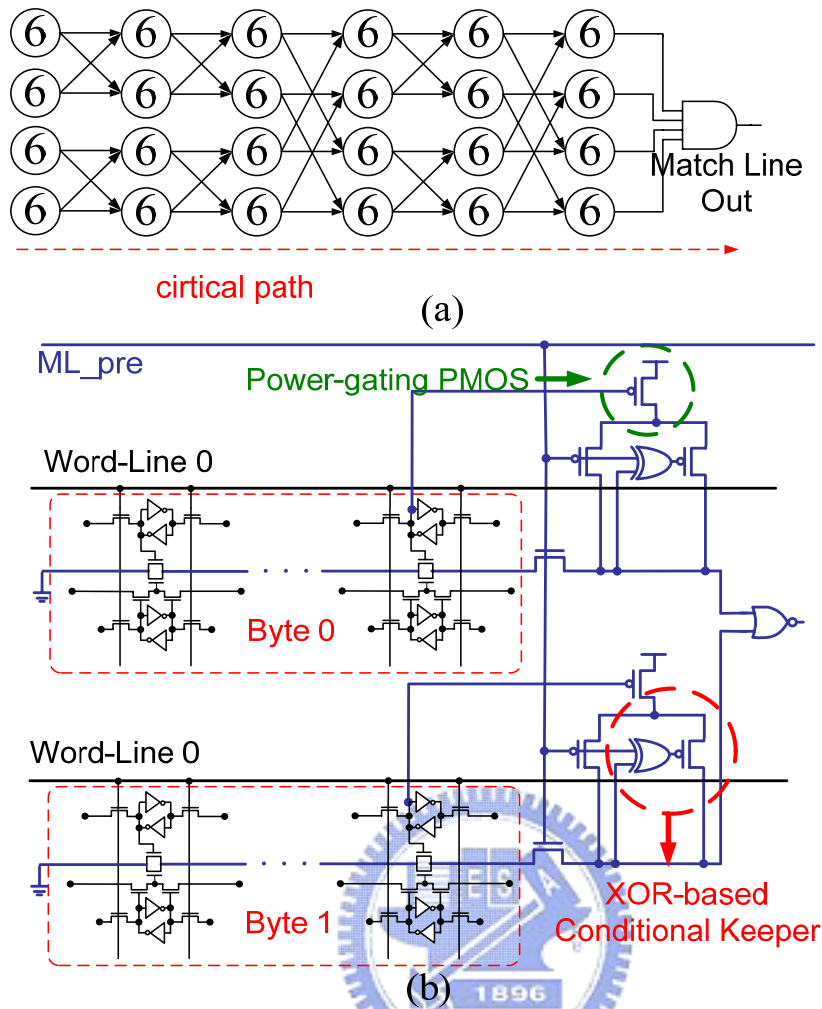


Fig.4.6 (a) butterfly match-line scheme (b) butterfly connection style for two TCAM segment

#### 4.4.2 XOR-Based Conditional Keeper

Conventional keepers have worse performance in propagation delay and power consumption. Therefore, a novel keeper, XOR-based conditional keeper [85], would be presented in this section. The main idea of the proposed XOR-based conditional keeper is to avoid keeper turning on at the beginning of evaluation phase in dynamic circuit. The new control signals and their corresponding keeper status are shown in Table 4.3 and described as follows:

Table 4.3 Control organism of XOR-based conditional keeper

<b>CLK</b>	<b>Floating Node</b>	<b>Control Signal on gate of keeper</b>
Low	Low	Low, to speed up the process of pre-charge
Low	High	High, to avoid the impact on performance at the very beginning of evaluation
High	Low	High, keeper should be off
High	High	Low, keeper should be activated to enhance the capability of noise immunity

When match-line pre-charge signal and the floating node are both low, the TCAM circuit is at the beginning of the pre-charge period and we should turn on the conditional keeper to speed up the pre-charge procedure. As match-line pre-charge signal is low and the floating node is high, the pre-charge process is completed and the gate is ready for evaluation. Therefore, the conditional keeper should be turned off to avoid the impact on the delay and unnecessary power consumption.

While match-line pre-charge signal and floating node are both high, match-line is either at the beginning of the evaluation process or stores state HIGH on the floating node at the end of the evaluation process. If it is at the beginning of the evaluation process, the floating node will be eventually at the proper voltage level as long as the delay of the XOR gate is larger than the propagation delay of dynamic circuits. Since the delay time of dynamic circuits is shorter, the conditional keeper is slightly turned on at the beginning of the evaluation process and is fully turned on or off on the grounds of the final value stored on the floating node.

When match-line pre-charge signal is high and the floating node is low, the evaluation mode is finished and the final value stored on the floating node is low. Consequently, the conditional keeper should be fully turned off. According to the desired control signals, an XOR gate is what we need for generating these signals. Fig. 4.7 shows the XOR gates we used and Fig. 4.8 shows the timing diagram for the XOR-based conditional keeper.

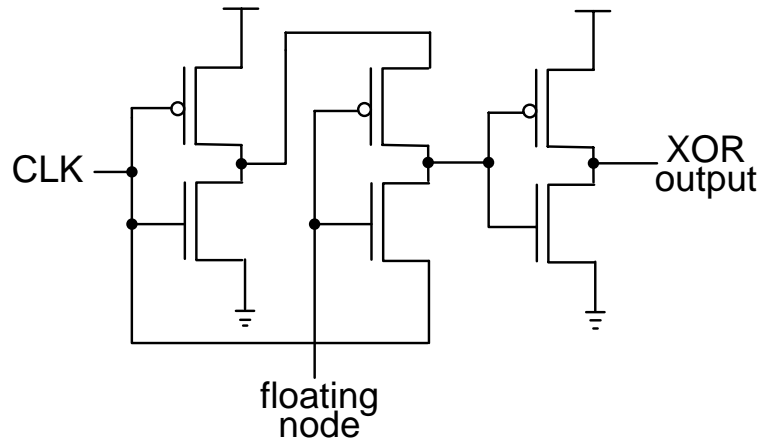


Fig. 4.7 The XOR gate implementation

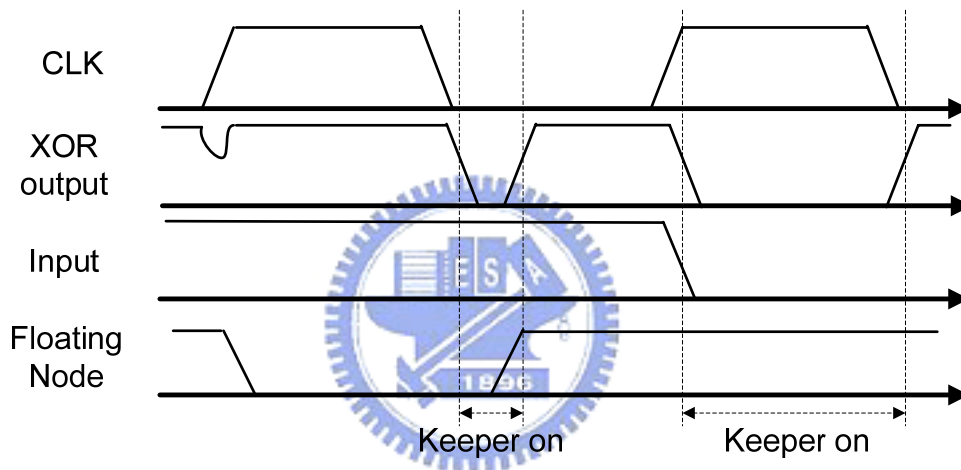


Fig. 4.8 Timing diagram for XOR-based conditional keeper

The reason why this XOR gate is chosen is that it requires fewer transistors and generates rail to rail output. There are still other 4 transistor XOR gates appeared in publications. However, most of those XOR gates are pass-transistor-based, thus the gates cannot have full swing output. This disadvantage degrades the noise immunity of conditional keepers.

### 4.4.3 Don't Care Based Power Gating Match-line Scheme

The voltage at the floating node of a match-line is pulled up in pre-charge phases by pre-charge circuits. When a specified pattern occurs, all don't-care bits are set as true in a segment. It represents that the floating node must be discharged at the evaluation phase in every searching operation. Hence, the power-gating PMOS is

applied to disconnect the voltage source and the floating node as Fig. 4.6 (b). The unnecessary pre-charge and discharge of the floating node can be eliminated by the power-gating PMOS. The gate of the power-gating PMOS is connected to the MSB of each TCAM segment. If the MSB is in don't-care state, the power-gating PMOS will be turned off to isolate the floating node and unnecessary power consumption is thus eliminated.

#### **4.4.4 Don't Care Based Hierarchy Search-Line Scheme**

A novel don't-care based hierarchical search-line scheme was investigated to decrease the switching capacitances without any search time overhead. It divides the search-lines into a two-level hierarchy, global search-lines (GSLs) and local search-lines (LSLs). The GSLs are active every cycle, but the LSLs are active only when necessary. A simplified architecture of proposed don't-care based hierarchical search-line scheme is shown in Fig. 4.9 (a). The entire TCAM is divided into four blocks. Each block consists of several match-lines and one global search-line to local search-line buffer (GSL-to-LSL buffer). For IPv6 addressing lookup tables with TCAM, prefix is sorted in the order of prefix length. The longest prefix is located at the bottom of the TCAM as Fig 4.9 (a). Hence, the upper TCAM cells are certainly don't-care terms. The GSL-to-LSL buffer can be controlled by the don't-care state.

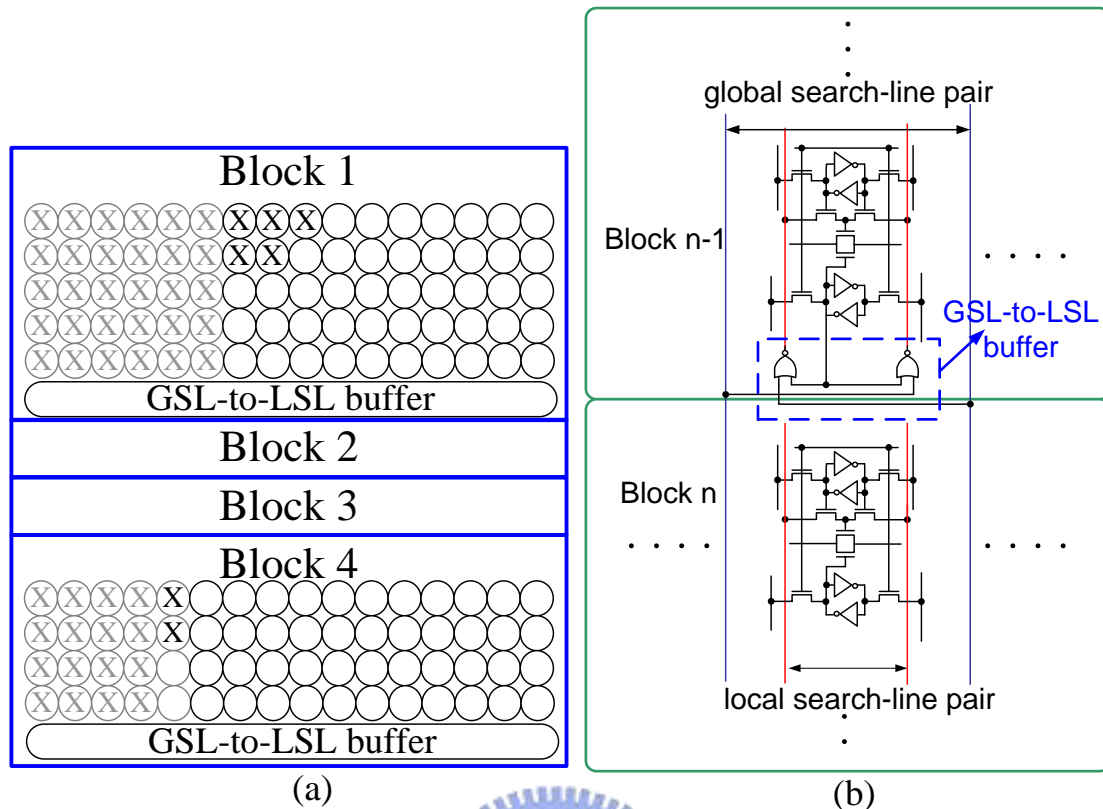


Fig. 4.9 (a) A simplified architecture (b) Implementation of don't care based hierarchy search line

## 4.5 Architecture

IPv6 addresses are 144-bit identifiers for interfaces and sets of interfaces. No matter NOR-type or AND-type match-line scheme, it takes long time to finish the search operation. In order to solve this problem, pipelined match-line scheme is strongly necessary. The other main point is that TCAM consume high power in search operation. Therefore, the goal is to reduce power consumption associated with the large amount of parallel active circuitry without sacrificing speed or memory density. NOR-type has high search speed property. When any bit is mismatched in NOR-type (implying that this word is mismatch), the match-line will be discharged. In IP addressing lookup application, destination address matches with more than one word in lookup table. Compare to entire IP addressing lookup table, the percentage of matching word is still few. Therefore, NOR-type match-line scheme consumes a lot of power in search operation. Unlike NOR-type scheme, AND-type match-line scheme discharged only when all bits are matched (implying that this word is matched). It is

obviously that AND-type match-line scheme consumes less power than NOR-type match-line scheme. However, two drawbacks of the AND match-line are a quadratic delay dependence on the number of cells, and a low noise margin.

In this chapter, a novel butterfly match-line scheme is proposed to reduce the search delay. Furthermore, noise immunity is also improved by applying XOR-based conditional keeper. Thus, the disadvantage of AND-type match-line scheme could be solved. From eq. (4.1), decreasing switch activity is useful to reduce dynamic power consumption. Therefore, we choose the AND-type CAM architecture to implement the TCAM due to the switching activity issue. IPv6 addresses consists of 144-bit, this is unfavorable to search speed. In order to reduce the effect of this problem, we suggest the TCAM array might be divided into two sub-arrays, as illustrated in Fig. 4.10. If the two sub-arrays architecture is adopted, being two smaller groups, the search time can be decreased effectively because the compared bits number of match-line is divided by two.

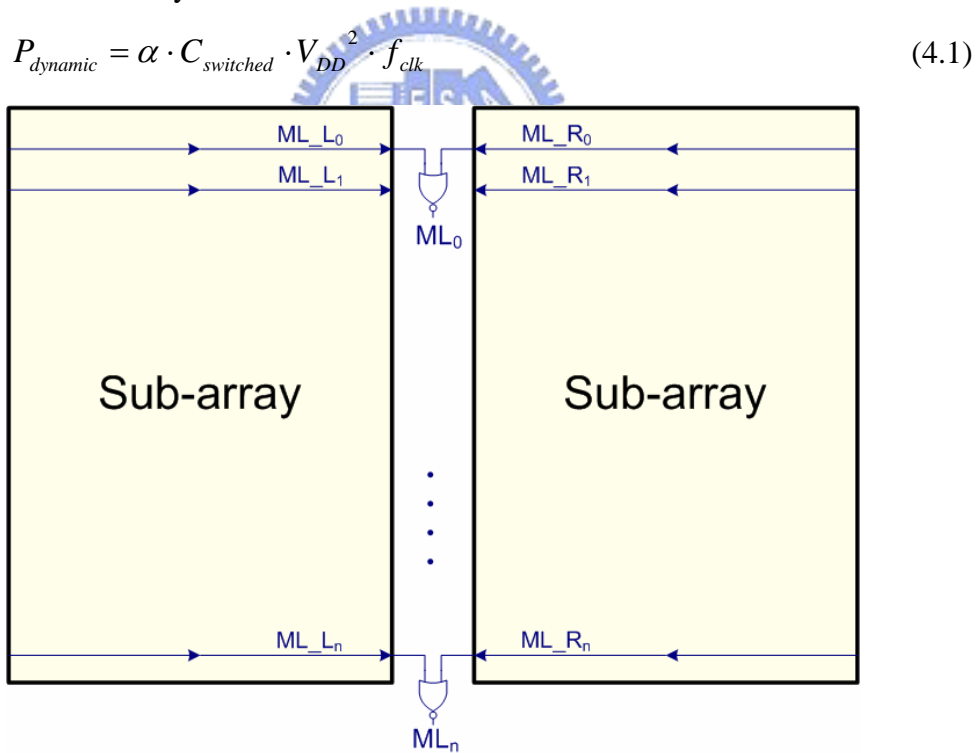


Fig. 4.10 TCAM array divided into two sub-arrays.

From Table 4.2, most prefix length in IP addressing lookup table is short. If the TCAM segment is don't-care, pre-charge of match-line is unnecessary. Furthermore, don't-care TCAM segment results a matching signal independent of destination address. For further power reduction, don't-care based low power technique will be

applied to our design. Don't-care based power-gating technique reduces switching activity on match-line. Don't-care based hierarchical search-line scheme reduce switching capacitance on GSL and switching activity on LSL.

## 4.5.1 Match-line Scheme

### 4.5.1.1 Butterfly AND-type match-line scheme for IPv6

There are three types of butterfly match-line schemes. The properties of IPv6 addressing architecture should be considered to get the best performance. In section 4.1, we know that most prefix length is shorter than 64-bit. Fig. 4.11 shows the IP addressing lookup table based on TCAM. TCAM cell has don't-care state will result a matching signal independent of destination address. Those don't-care bits are located at the left sub-array. Hence, there are more matching TCAM segments in left sub-array. If mismatch in right sub-array can also disable left sub-array, search power can be further reduced. As mentioned previously, the butterfly match-line has such property. If any TCAM segment in the same stage is mismatch, the stage after its following stage will be certainly disabled. Therefore, butterfly match-line scheme will be applied to this design. Fig. 4.12 shows the architecture applied to this design. The entire word is divided into two parts. If there is any one mismatching TCAM segment both left and right sub-array can be stopped.

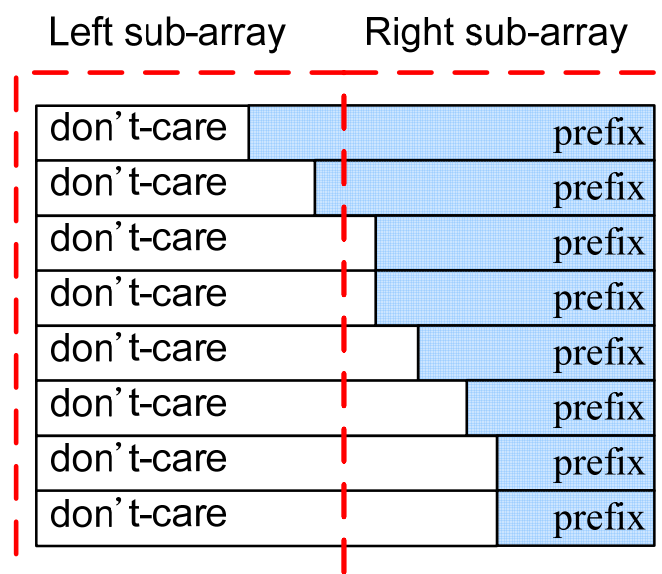


Fig. 4.11 IP addressing lookup table based on CAM



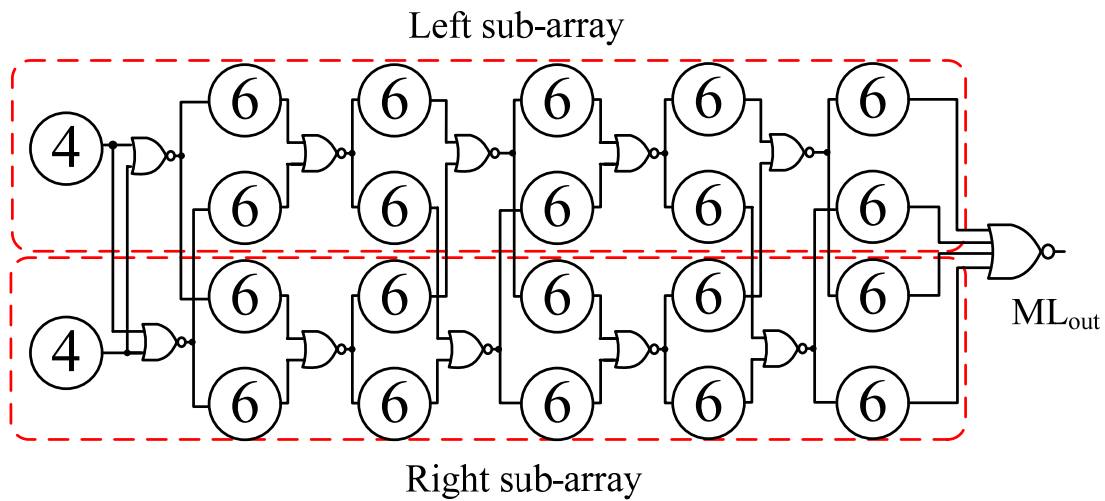


Fig. 4.12 Butterfly match-line scheme for IPv6 addressing lookup application

#### 4.5.1.2 Applying Don't-care based Power-Gating Technique

Don't-care based power-gating technique prevents unnecessary pre-charge in search operation. Fig. 4.13 shows the circuit implementation of proposed match-line scheme. The power-gating PMOS is added between pre-charge circuit and power source. It's obviously, applying don't-care based power-gating technique do not increase search delay.

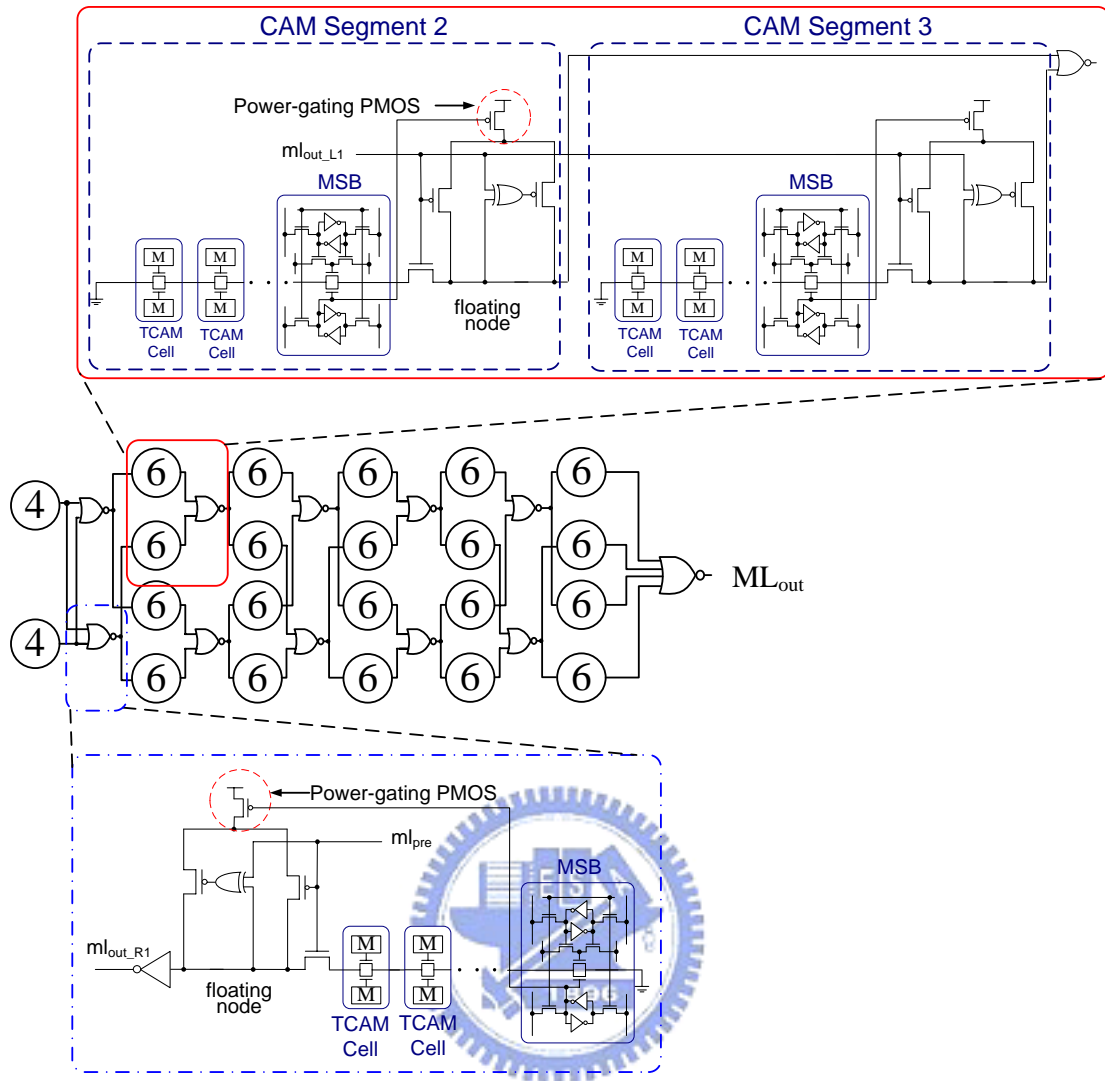


Fig. 4.13 Circuit implementation of proposed match-line scheme

## 4.5.2 Ternary CAM Cell

In Section 4.4, implementation of AND-type TCAM is clearly explained. However, the other important sources of power consumption are the high capacitive switching bit-lines and search-lines. Fig. 4.14(a) shows the 16-transistor TCAM cell. In this TCAM cell, the bit-lines pair and search-lines pair are shared thus the switching capacitances (consists of junction capacitances) of these shared lines become extra large. Either the write operation or read operation, the bit-lines are switching frequently. Besides, during the search operation, the search-lines are also switching frequently. On the other words, these highly capacitive bit-lines/search-lines are always switching except TCAM working on standby mode; therefore, they cause

large amount of power consumption. In order to avoid switching larger capacitances, we divide into two pairs of lines. One pair of lines are used to bit-lines and the other one are used to search-lines, as shown in Fig. 4.14 (b). We use an additional pair of lines, not only to reduce the power consumption during write, read, and search operations but also to simplify the peripheral circuits design. However, the penalty is that more lines complicate the layout router.

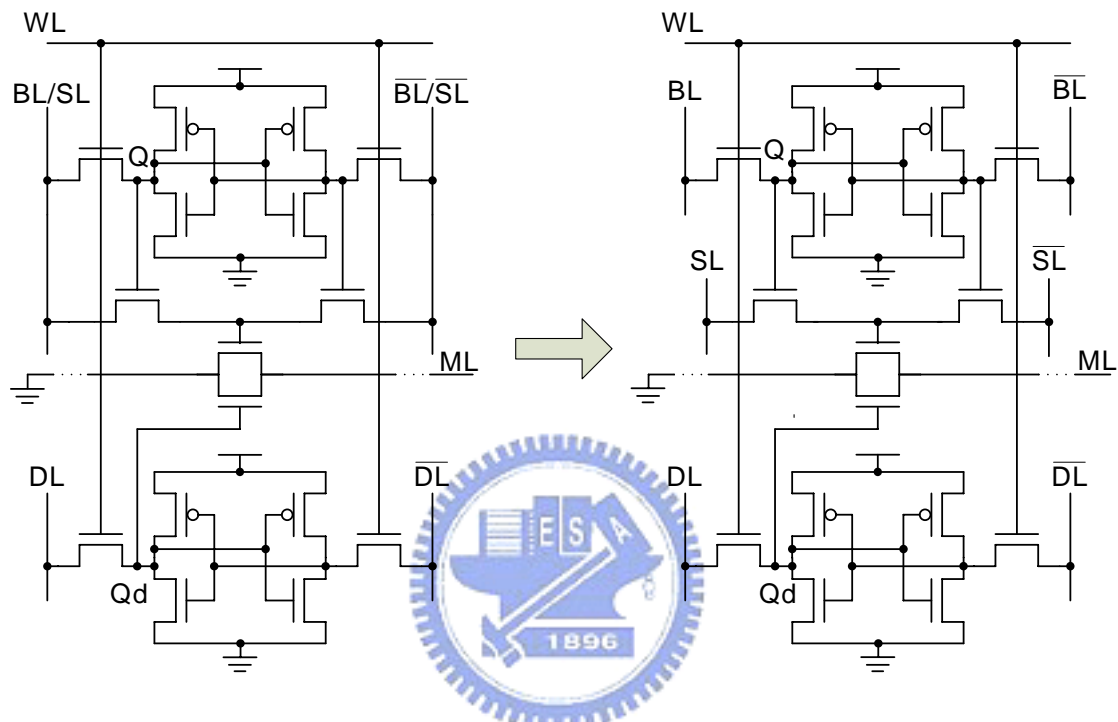


Fig. 4.14 16-transistor AND-type TCAM cell (a) Bit-line and Search-line are shared. (b) Bit-line and Search-line are separated.

Table 4.4 State assignments for TCAM cell.

State	Q	Qd	SL	ML
Zero (0)	0	0	0	0
	0	0	1	floating
One (1)	1	0	0	floating
	1	0	1	0
Don't Care (X)	0	1	0	0
	0	1	1	0
	1	1	0	0
	1	1	1	0

### 4.5.3 Don't-care based hierarchical search-line scheme for IPv6

The proposed don't-care based hierarchical search-line scheme is presented previously. From Table 4.4, most prefix length is distributed from 16 to 48. Most prefix are shorter than 64. In our design, we divide 256 words into 5 blocks. Each block consists of 64 words for block1, block2 and block 3 respectively. For block 4 and block 5, each block consists of 32 words. There is one GSL-to-LSL buffer between each block. Fig. 4.15 shows the block diagram of proposed hierarchical search-line scheme.

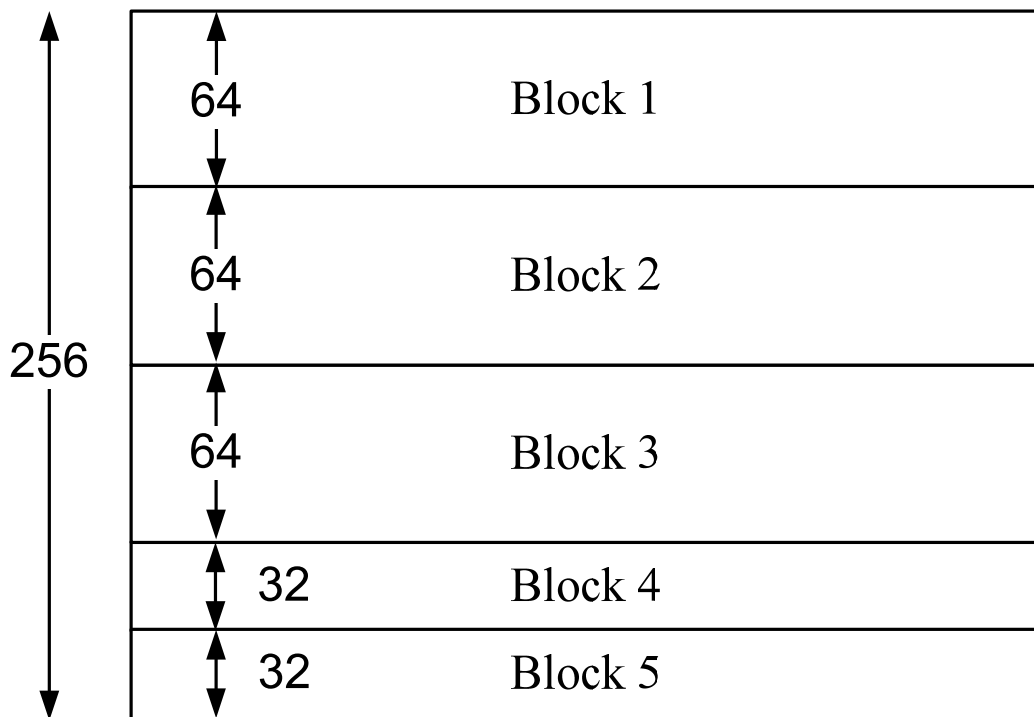


Fig. 4.15 Block diagram of proposed hierarchical search-line scheme