

Innovative TCAM Solutions for IPv6 Lookup: Don't Care Reduction and Data Relocation Techniques

Anh PHAM^{1,2}, Doanh BUI^{1,2}, Phuc Thien Phan NGUYEN^{1,2}, Linh TRAN^{1,2}

¹ Dept. of Electronics, Ho Chi Minh University of Technology (HCMUT), Ho Chi Minh City, Vietnam

² Vietnam National University Ho Chi Minh City (VNU-HCM), Ho Chi Minh City, Vietnam

linhtran@hcmut.edu.vn

Submitted August 8, 2024 / Accepted November 4, 2024 / Online first November 18, 2024

Abstract. Ternary Content-Addressable Memory (TCAM) enables high-speed searches by comparing search data with all stored data in a single clock cycle, using ternary logic ("0", "1", "X" for "don't care") for flexible matching. This makes TCAM ideal for applications like network routers and lookup tables. However, TCAM's speed increases silicon area and limits memory capacity. This paper introduces a low-area, enhanced-capacity TCAM for IPv6 lookup tables using Don't Care Reduction (DCR) and Data Relocation (DR) techniques. The DCR technique requires only $(N + \log_2(N))$ -bit memory for an N -bit IP address, reducing the need for $2N$ -bit memory. The DR technique improves TCAM storage capabilities by classifying the IPv6 into 4 different prefix length types and relocating the data in the prefix bit into the "X" cells. The design features a 256×128 -bit TCAM (eight 32×128 -bit memory banks) on a 65 nm process with a 1.2 V operation voltage. Results show a 71.47% increase in area efficiency per stored IP value compared to conventional TCAM and a 20.97% increase compared to data-relocation TCAM.

Keywords

Ternary content-addressable memory (TCAM), high-speed searches, IP version 6 (IPv6), IPv6 lookup table, don't care reduction (DCR), data relocation (DR), conventional TCAM (CV-TCAM), data-relocation TCAM (DR-TCAM), low-area technique, enhanced-capacity technique

1. Introduction

In network router applications, the router's performance is directly reflected in its capability of the network to handle real-time traffic [1]. One of the most time-consuming operations in this context is route lookup, which relies on a lookup table. To enhance the searching speed of the lookup table, Content-Addressable Memory (CAM) is used, including Binary Content-Addressable Memory (BCAM) and Ternary Content-Addressable Memory (TCAM).

As the parallel searching structure of CAM, they provide a fast data search function with a parallel searching process across all memory cells. BCAM is useful for exact match operations where the stored data needs to be compared exactly with the search data (as BCAM only has one memory cell for storing two states, typically represented as "1" or "0"). Meanwhile, TCAM extends the capabilities of BCAM by introducing a third state, often described as "X" or "don't care", by using two memory cells (a core cell for storing data and a mask cell for storing the don't care state). Therefore, TCAM is a good choice for implementing this lookup operation due to its fast search capability [2] and flexible matching process. While TCAM is used for the IPv6 lookup table, it requires more area cost with the large-scale lookup table due to the complex structure of TCAM. The area efficiency per stored IPv6 address of the lookup table is affected by the IP prefix length as mentioned above, which has large memory cells used for identifying the don't care "X" value.

Recently, the lookup table of IPv6 based on TCAM architecture has been proposed with design techniques to improve the drawbacks of area cost and limited capacity [3], [4]. For the area cost improvement, a Don't Care Reduction scheme (DCR) is proposed for modifying the mask cells to reduce the number of transistors [4]. For enhancing the memory capacity, the data-relocation technique (DR) is proposed for classifying the IPv6 address based on the bit-length of its prefix bit and relocating the prefix bits into mask cells [3]. Reducing area cost and enhancing the memory capacity based on these techniques come with the drawback of more complex TCAM architecture. Regarding CAM power consumption, several techniques have been developed to reduce the power consumption of sense lines (SLs). Some methods achieved this by recycling the charge on SLs [5], [6] or by decreasing the swing voltage on SLs [7]. The pipelined hierarchical search approach decreased SL power by selectively activating a few sub-SLs identified in the previous pipeline stage [8]. Additionally, the segmented SL scheme reduced the effective SL length by sorting all stored data with "X" (don't care) cells, and then blocking signal propagation to the segmented SLs behind these "X" cells [9], [10]. Many techniques have been developed to reduce the power consumption of match

lines (MLs). High-speed NOR-type CAMs were able to reduce ML power [9], [11]. Some methods achieved this by allocating less power to mismatched MLs [12], [13] or by decreasing the swing voltage on MLs [14], [15]. Other techniques reduced the number of activated MLs [16], [17]. Additionally, low-power NAND-type CAMs improved search speed using an AND-type ML scheme based on PF-CDPD logic [6], [18]. Some CAM structures have been proposed for improving power consumption. Although the NAND-type CAM consumes the least power in the MLs, it is slow [19]. In contrast, the NOR-type CAM is fast but consumes the largest power [20]. To achieve both low power consumption and high speed, the NOR-type CAMs were developed to reduce the power consumption [2], [21], and the NAND-type CAMs were developed to improve the searching speed [6], [9].

In this study, the TCAM structure, which is driven by the Don't Care Reduction scheme (DCR) and DR technique, is designed to improve the area cost and the area efficiency per stored IP address of the IPv6 lookup table. This work proposes a new TCAM lookup table architecture based on DR and DCR techniques with a detailed explanation and design configuration. To evaluate the performance of the proposed TCAM as an IPv6 lookup table, the design is scaled to a 256×128 -bit memory size. Schematic simulations are performed to validate the operation using 65 nm technology, confirming successful TCAM functionality

The remainder of the paper is structured as follows. Section 2 introduces the methods and research design of the Don't Care Reduction scheme and data-relocation technique which includes the methodology and architecture based on these two techniques. Section 3 validates the measured parameters which are used to compare the efficiency of the design with the conventional TCAM. Future work directions of this study are mentioned in Sec. 4, and Section 5 concludes this work.

2. Method and Research Design

2.1 Enhanced Capacity Technique

2.1.1 The Concept of Data-Relocation Technique

In this part, the concept of improving the memory size of the TCAM using a data-relocation scheme is proposed. The DR-TCAM (Data-Relocation TCAM) increases the number of IP addresses stored in the TCAM by relocating the data in the prefix bit into the "X" cells. The solution for this is to classify the data into 4 main types: Type 0, Type 1, Type 2, and Type 3. Type 0 is used to identify the empty (used to store don't care "X"). Type 1 is for the IP address with a prefix length between 1 and 32 bits so that we can store 4 IP addresses with this type in one bank (128 bits for 1 bank). For Type 2, we consider the IP address with the prefix length between 33 and 64 bits; so that in the 128-bit bank, we can store 2 IP addresses with this type instead of 1 IP address when using conventional TCAM. Type 3 is for

the address with the prefix bits from 65 to 128, it is just like the conventional TCAM which can only store one 128-bit IP address in one bank.

In Tab. 1, in DR-TCAM, the IP addresses are grouped into three types of words according to the prefix length. In this comparison, the ratio of the three types of words uses the statistic prefix length distribution [18]. For the Type 3 bank size of 32×128 -bit cells, we can store 1 Type 3 word and 31 Type 2 words ($31 = 32 - 1$). A Type 2 bank with 64×64 -bit cells stores 58 Type 2 words ($58 = 89 - 31$) and stores 6 Type 1 words ($6 = 64 - 58$). Finally, two Type 1 banks with 256×32 -bit cells (64×128 -bit cells) store 160 Type 1 words ($160 = 166 - 6$). In the Type 1 bank, 96×32 -bit cells ($96 = 256 - 160$) are still empty. The empty cells can store an additional 96 Type 1 words. We can also decrease the number of banks in the memory to 50% when $N = 256$, the area of the DR-TCAM can be improved up to 50% for $N = 256$ and about 34.4% for $N = 4K$ (as shown in Tab. 1).

2.1.2 The Architecture of DR-TCAM

Figure 1 shows the structure of a memory bank size 32×128 -bit, which consists of four 32×32 -bit memory cell blocks, a bank control unit, a Search Line Multiplexer (SL MUX), an address encoder, and these components apply the data relocation scheme. As we see in Tab. 2, each bank has the typical data for classifying the type of stored IP in the memory bank, which is the 2-bit Bank Selection Register – BSR[2:1] in the bank control unit. In the bank control unit, the BSR data will be decoded into the BS signal – BS[1:4] – for selecting the input search data into each block with a suitable IP type by the Search Line Multiplexer component (SL MUX) and adjusting the match line signals in each block based on the IP type stored in the bank through the address encoder component.

		TCAM	DR-TCAM		
Prefix length		1–128	1–32	33–64	65–128
Prefix length distribution		-	64.92%	34.91%	0.17%
Number of stored words	$N = 256$	256	166	89	1
	$N = 4K$	4096	2569	1430	7
Bank size		32×128 -bit cells			
Bank type		Type 3	Type 1	Type 2	Type 3
Number of banks	$N = 256$	8	2	1	1
	$N = 4K$	128	21	22	1
Number of cells	$N = 256$	32,768	16,384 (50%)		
	$N = 4K$	524,288	180,224 (34.4%)		

Tab. 1. Comparison of TCAM cell with 32×128 -bit bank [3].

Bank type	Type 0	Type 1	Type 2	Type 3
BSR[2:1]	00	01	10	11
BS[1:4]	BS[1] = 1	BS[1] = 0	BS[1] = 0	BS[1] = 0
	BS[2] = 0	BS[2] = 1	BS[2] = 0	BS[2] = 0
	BS[3] = 0	BS[3] = 0	BS[3] = 1	BS[3] = 0
	BS[4] = 0	BS[4] = 0	BS[4] = 0	BS[4] = 1
Bank_ML_en	0	1	1	1

Tab. 2. Output enable signal of the bank control unit of the 32×128 TCAM.

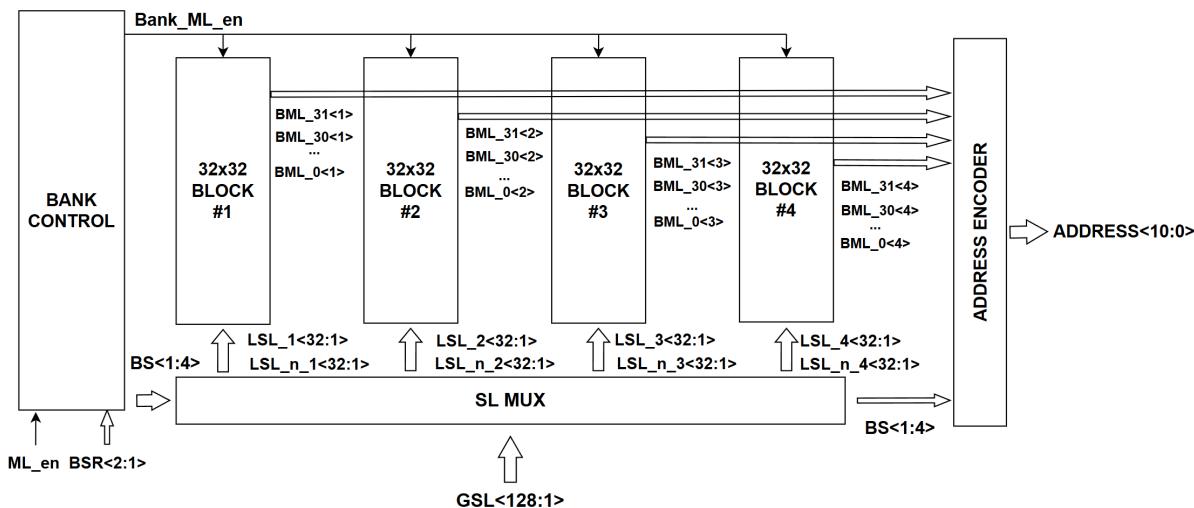


Fig. 1. 32 × 128-bit memory bank structure of the DR-TCAM.

Bank type	Type 0	Type 1	Type 2	Type 3
BSR[2:1]	00	01	10	11
BS[1:4]	BS[1] = 1 BS[2] = 0 BS[3] = 0 BS[4] = 0	BS[1] = 0 BS[2] = 1 BS[3] = 0 BS[4] = 0	BS[1] = 0 BS[2] = 0 BS[3] = 1 BS[4] = 0	BS[1] = 0 BS[2] = 0 BS[3] = 0 BS[4] = 1
LSL_1	No connection	GSL_1	GSL_1	GSL_1
LSL_2		GSL_1	GSL_2	GSL_2
LSL_3		GSL_1	GSL_1	GSL_3
LSL_4		GSL_1	GSL_2	GSL_4

Tab. 3. Output signals of SLMUX of the 32 × 128-bit DR-TCAM.

The signal BS[1:4] separates the global search line (GSL) signal, which is used to input the search IPv6 address into the memory bank, into four local search lines (LSL_1, LSL_2, LSL_3, LSL_4) with a suitable IP type stored in that bank through the SL MUX component. In Fig. 2(a), GSL[128:1] is separated into 4 parts, which include GSL_1[32:1], GSL_2[32:1], GSL_3[32:1], and GSL_4[32:1] from the MSB to the LSB. This component consists of 4 32-bit SL selection switches from MUX1 to MUX4 in Fig. 2(a), and each selection switch is combined with thirty-two 1-bit SL selection switches in Fig. 2(b)–(e). As we can see in Tab. 3, when BS[1] = "1", the memory bank, in this case, is considered a Type 0 bank, which means all ML enable signals (Bank_ML_en) and all local search lines (LSLs) are disabled for power saving. When BS[2] = "1", the bank memory becomes a Type 1 bank. The SL MUX component will connect four local search lines from LSL_1[32:1] to LSL_4[32:1] with the same 32-bit MSB of the global search line (GSL_1[32:1]). When BS[3] = "1", the memory bank is considered the Type 2 bank, the LSL_1[32:1] and LSL_2[32:1] signals are connected to the GSL_1[32:1] and GSL_2[32:1] respectively (GSL_2[32:1] = GSL[96:65]). The LSL_3[32:1] and LSL_4[32:1] signals are connected to the GSL_1[32:1] and GSL_2[32:1] respectively. When BS[4] = "1", the bank becomes a Type 3 bank. So, the LSLs from block #1 to block #4 (LSL_1[32:1], LSL_2[32:1], LSL_3[32:1], LSL_4[32:1]) are connected to their corre-

sponding global search lines, which are from GSL_1[32:1] to GSL_4[32:1] respectively.

In Fig. 2, four block MLs (BML[1:4]) match results from four 32-bit words in the same row of four blocks. The word length of the stored data changes from 32 bits to 128 bits according to the bank type. Therefore, four ML outputs (ML_out[1:4]), which are the outputs of the ML Selector (Shown in Fig. 2), are constructed by the combinations of the four BMLs, as shown in Tab. 4. In Type 0 bank, ML_out signal does not exist (All ML_out[1:4] = "0000") because there is no IP address data stored in the bank. In the Type 1 bank, the ML_out[1:4] are the same as their corresponding BML[1:4]. In the Type 2 bank, the ML_out[1] and ML_out[3] represent two ML outputs in this type. In one bank memory row, if the BML[1] and BML[2] are matched, the output match line (ML_out[1]) is matched. In the second part in one row, the ML_out[3] is matched when both BML[3] and BML[4] are matched. In Type 3 bank, The ML_out[1] is matched when all BML[1:4] are matched.

As we can see in Fig. 3, the address encoder (with low priority demand based on ROM architecture) conducts the encoder address (EA[6:0]) from the output ML signals (ML_out[1:4] from row 31 to 0 in one 32 × 128-bit memory bank) to find the IP Address that matches the input data in search operation with the low priority demand. In Tab. 5, this component encodes the single input from 0 to 127 into 7'b0 to 7'b111_1111.

2.2 Low-Area Technique

2.2.1 The Concept of Don't Care Reduction Scheme

In this concept, we discuss the don't care value in the IP prefix address. The advantage of TCAM is that it can identify the don't care value by using 2 memory cells (the core memory cell for storing the data – "0" or "1", and the mask memory cell is used to store the value that is used in the search operation to compare to the data in core memory to

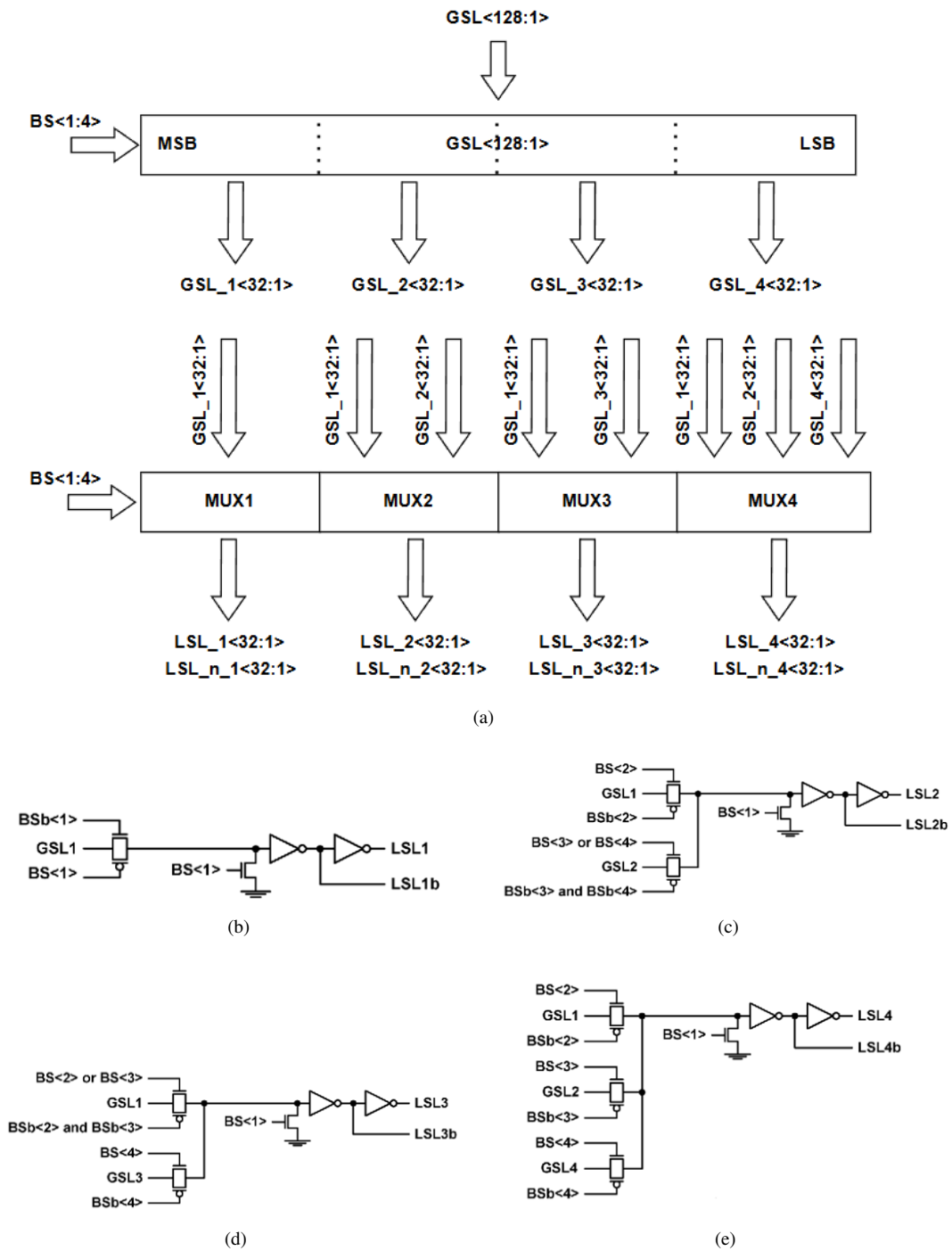


Fig. 2. (a) Detail block diagram of SL MUX of the 32 x 128-bit DR-TCAM; (b) SL MUX switches in MUX1; (c) SL MUX switches in MUX2; (d) SL MUX switches in MUX3; (e) SL MUX switches in MUX4.

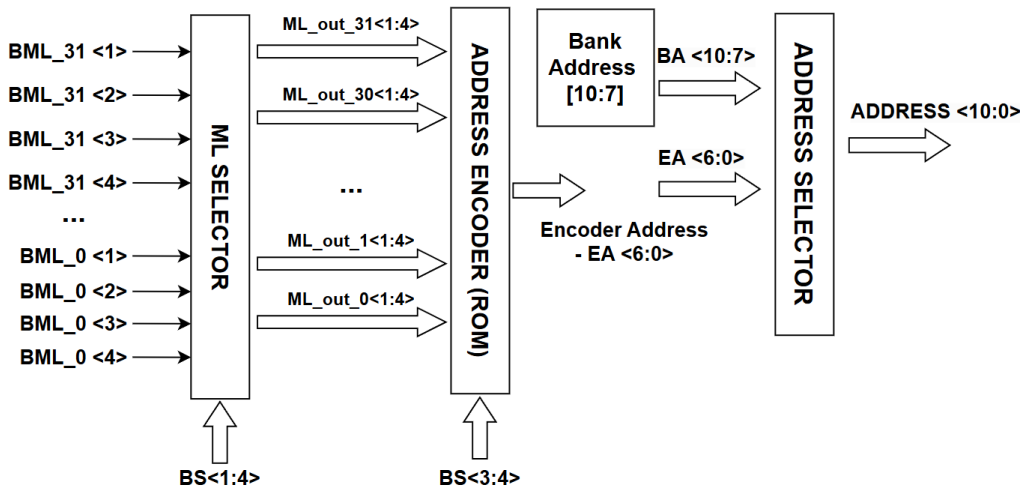


Fig. 3. Address encoder detail block diagram of the 32 × 128-bit DR-TCAM.

Bank type	Type 0	Type 1	Type 2	Type 3
BSR[2:1]	00	01	10	11
BS[1:4]	BS[1] = 1 BS[2] = 0 BS[3] = 0 BS[4] = 0	BS[1] = 0 BS[2] = 1 BS[3] = 0 BS[4] = 0	BS[1] = 0 BS[2] = 0 BS[3] = 1 BS[4] = 0	BS[1] = 0 BS[2] = 0 BS[3] = 0 BS[4] = 1
ML_out [1]	0	BML_[1]	BML_[1] & BML_[2]	BML_[1] & BML_[2] & BML_[3] & BML_[4]
ML_out [2]	0	BML_[2]	0	0
ML_out [3]	0	BML_[3]	BML_[3] & BML_[4]	0
ML_out [4]	0	BML_[4]	0	0

Tab. 4. Output signal for ML selector.

Input							Output						
A127	A126	A125	...	A2	A1	A0	Y6	Y5	Y4	Y3	Y2	Y1	Y0
X	X	X	...	X	X	1	0	0	0	0	0	0	0
X	X	X	...	X	1	0	0	0	0	0	0	0	1
X	X	X	...	1	0	0	0	0	0	0	0	1	0
...
X	X	1	...	0	0	0	1	1	1	1	1	0	1
X	1	0	...	0	0	0	1	1	1	1	1	1	0
1	0	0	...	0	0	0	1	1	1	1	1	1	1

Tab. 5. 128 × 7 low priority encoder true table.

figure out the TCAM cell is stored "X" or not). As we see in Fig. 4(a), with the N -bit IP address, the conventional TCAM needs $2N$ -bit memory cells to store, with N -bit for Data and N -bit for Don't Care. The solution is based on the specification of the IPv6 address mentioned in the previous path (the IPv6 address input is the combinational of prefix bits and the prefix length), we will modify the N -bit memory for Don't Care into $\log_2 N$ -bit memory storing the first "X" position appear in the storing IP prefix address. Thus, the CAM cell in the TCAM using the Don't Care Reduction scheme is changed into BCAM instead of TCAM (because they don't need to identify the Don't Care value). So, the first "X" position value will detect the position for the search operation to bypass the mask bits (skip the region stored don't care value) as shown in Fig. 4(b). By using this technique, the reduction

of transistor used in TCAM can be reduced by reducing the mask cell of TCAM and the efficiency of this technique is enhanced when the TCAM is designed on a large scale.

2.2.2 The Architecture of DCR-TCAM

The 32-bit DCR-TCAM uses the cascaded three-level tree AND-ML [9] in consideration of the performance, power area, and stability, as shown in Fig. 5(a). The First 4-bit Bypass TCAM Cascade Block (#0) will use the ML_en signal from the input of the 32-bit DCR-TCAM and for the following block, they will use the BML# n of the previous block as the ML_EN signal for triggering the MLSA component in the search operation. In level 1 cascade, we consider 2 blocks (Block#0 and Block#1) which manage the bit data from 32 to 25. For cascade level 2, we have 2 blocks (Block#2 and

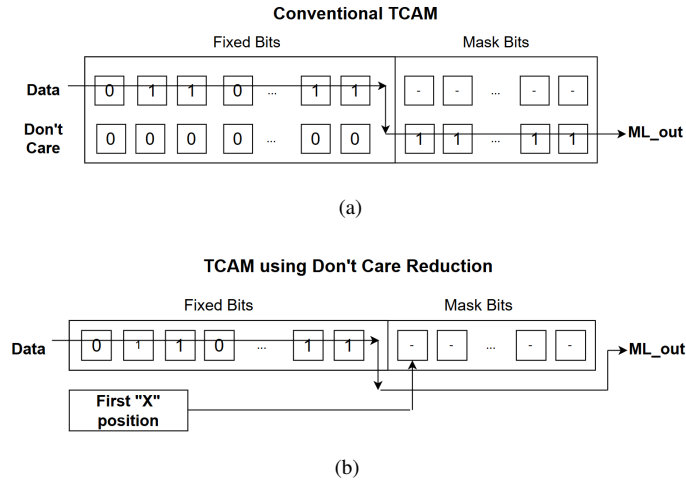


Fig. 4. Ideal of Don't Care Reduction technique with (a) N -bit IP address of the conventional TCAM in the search operation; (b) N -bit IP address of the DCR-TCAM in the search operation.

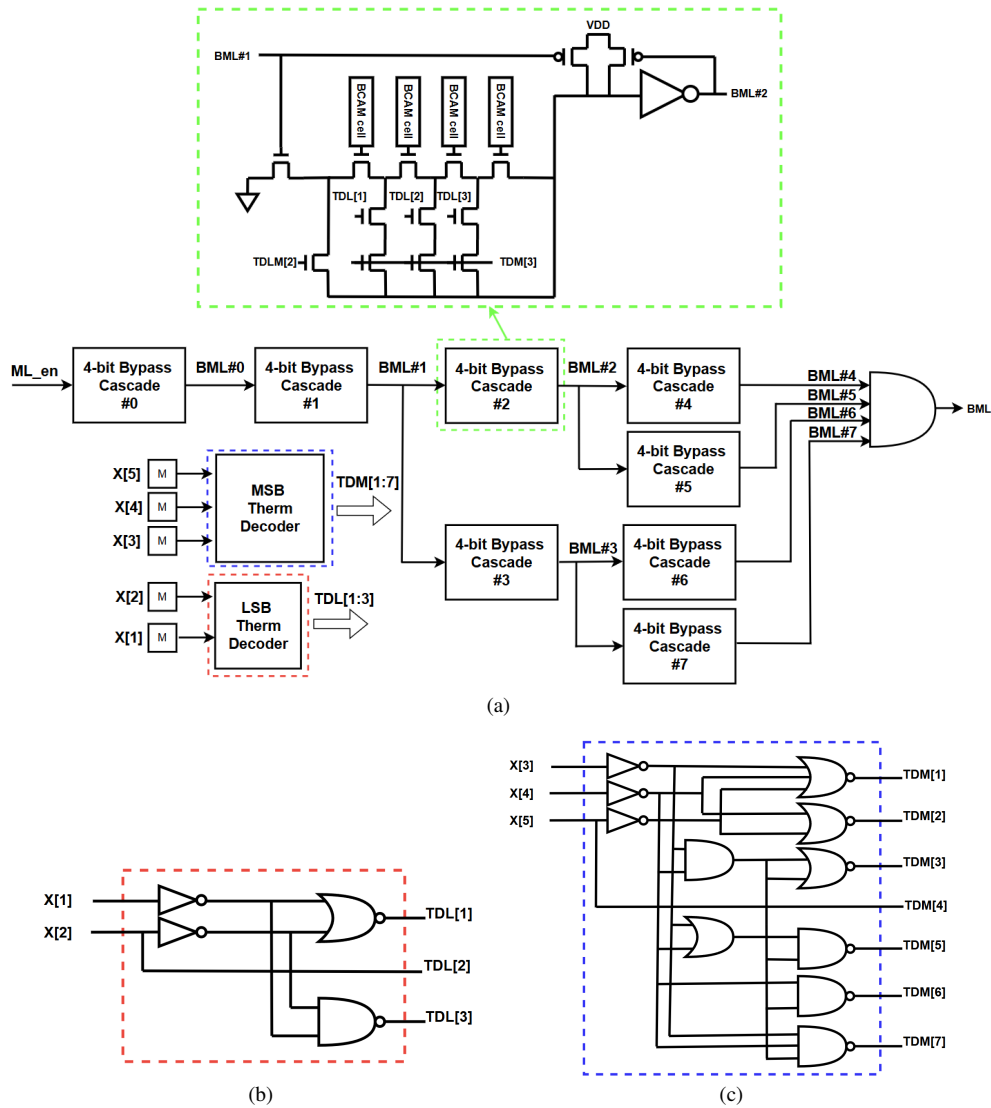


Fig. 5. 32-bit DCR-TCAM architecture with (a) 32-bit DCR-TCAM using the cascaded three-level tree (1_1-2-4) with the structure of 4-bit cascade sub-AND-ML; (b) LSB thermometer decoder logic design; (c) MSB thermometer decoder logic design.

Block#3) that manage the bit data from 24 to 17, and they use the signal BML#1 for ML_en signal (match line enable signal). For level 3 cascade, Block#4, #5, #6, #7, which manage the bit data in search and write operation from 16 to 1, will use the signal BML#2 and BML#3 for ML_en signals. The MSB and LSB thermometer of the 32-bit AND-ML bypass TCAM bank shown in Fig. 5(b), (c) is used to encode the first "X" position code for bypass enable signal for the 2-level bypass structure of each 4-bit bypass cascade sub AND-ML CAM.

2.3 The Specification of Proposed IPv6 Lookup Table

After applying two techniques to the lookup table design, Figure 6 shows the architecture of one memory bank with a size of 32×128 -bit consisting of 32×128 -bit memory component which is used to store the IPv6 address with the first "X" position data for each 32-bit IP stored address. The SL MUX component generates four suitable 32-bit local search data for each 32×32 -bit block memory in the memory bank (LSL_#n[32:1]) based on the global 128-bit input search data (GSL[128:1]) and the bank type data – bank selection data (BS[1:4]), which is decoded from the input bank selected register data (BSR[2:1]) by BANK CONTROL component (controlled by bank_control operation signals). The searching operation of this memory bank is controlled by WL signals (control the CAM's write and search operation) which are generated by the WL DECODER component (controlled by WL operation signal) and ML_en signal (control the AND-ML component in each 32-bit TCAM block) generated by BANK CONTROL component. The output searching ML of four 32×32 -bit memory blocks in a 32×128 memory bank (BML_#n[31:0]) are encoded by the ADDRESS ENCODER component with bank type-defined data (BS[1:4]) to output the memory address that stores the data matched with the input search data with low priority demand.

In Fig. 7, the lookup table 256×128 -bit TCAM design consists of 8 memory banks with size 32×128 -bit, the control unit which is used to generate the signals for WL decode, and the bank control unit for each bank, the input circuit is used to drive the input data into the lookup table and the address priority select component, which is used to select the address with low priority demand. The input circuit component takes global data inputs, which include 128-bit storing data (BL[128:1]), 128-bit global searching data (GSL[128:1]), 20-bit data for first "X" position code in each four 32-bit memory blocks (X_Data[20:1]) and 2-bit input bank selection data for identifying IP type stored in each 32×128 -bit memory bank. The output of this component (local data inputs) will transfer the data into 8 memory banks in the design sequentially from bank#1 to bank#8 due to the selected bank data input in the control unit component. The input operation signal of the control unit component (SW operation signals) detects the working operation of the design and sends control operation signals to every memory bank in

the design. During the searching process, each memory bank outputs the address data if there is any IP address matching with the input IP search data and the address priority select component collects this output (ADDRESS_1[10:0] to ADDRESS_8[10:0]) data and output the memory address (ADDRESS[10:0]) with low priority demand.

3. Result

In this section, the lookup table design is compared to the CV-TCAM with the Area-Cost and Capacity-Efficiency measurement. It consists of 3 main parameters including the Maximum number of stored words (no. of stored words - max), the maximum effective area per stored word compared to the CV-TCAM (effective area/word – max), and the maximum clock frequency (clock freq. – max (MHz)). In Tab. 6, the effect of the DCR scheme on the area cost is that the 32-bit memory bank with this technique can reduce 11.99% of the number of transistors. But with the complex architecture of enhanced capacity technique applied to the design, the SL MUX and address encoder component increase the number of transistors for classifying the IP type and input search data and encoding the output address, but this decrease is not significant as the design-scale of the lookup table which makes it use more memory bank to store data.

In Tab. 7, we compare the design with the comparison of the design with 3 different TCAM designs that are used in the IP lookup table, we can see that in the maximum number of stored IP addresses, the design has a 400% increase in memory capacity. With the reduction of transistor used in the design, the maximum effective area per word percentage parameter (compared to the CV-TCAM with the same capacity) of the design has the best effective value (28.53%) compared to others and the maximum clock frequency that the design can perform is nearly 300 MHz (298 MHz). The energy/bit/search parameter shows the design's power consumption in one period of searching time. In Tab. 7, the power consumption of this work is smaller than JSSC 2015 and JSSC 2018 with the same size and supply voltage (0.234 fJ compared to 0.41 fJ). This design reduces not only the area and search power consumption but also the leakage power due to the unactivated ML of the Type 0 bank in the search operation. For the effective energy/word/search (= energy/bit/search \times total CAM bits / no. of the stored word – max), it measures the energy per stored word in a search. The NAND-type TCAMs using the segmented SL scheme, which has smaller effective energy/word/search than the NOR-type TCAMs because the NAND-type ML consumes less power than the NOR-type ML and the segmented SL scheme blocks the signal propagation to the segmented SLs behind "X" cells [3]. As the maximum number of stored words in this work is higher compared to other designs, it consumes less power in searching operations (7.55 fJ) with a working frequency of 289 MHz.

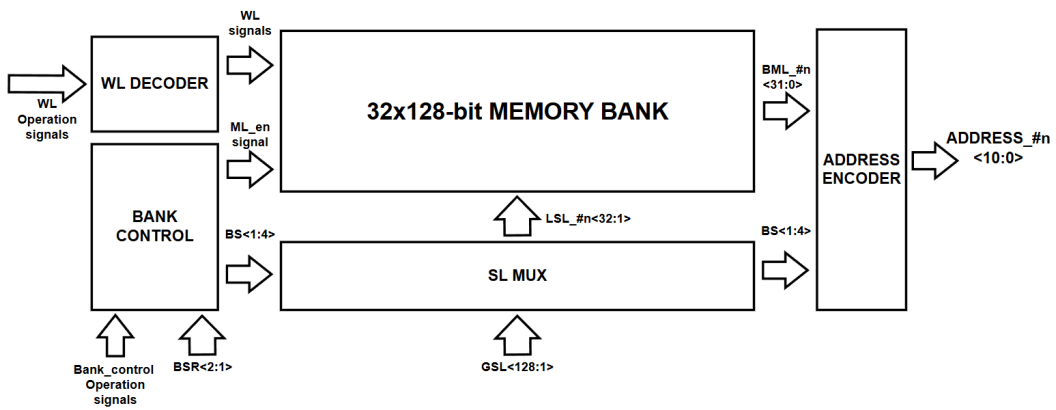


Fig. 6. 32×128 -bit memory bank design detailed block diagram.

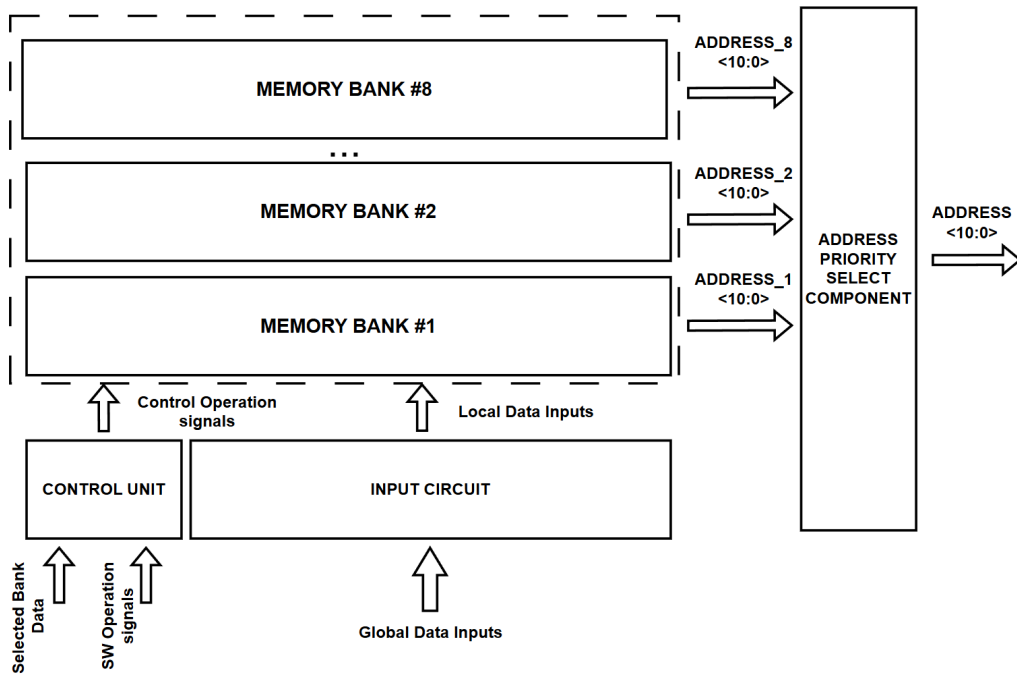


Fig. 7. 256×128 -bit lookup table design detailed block diagram.

		CV-TCAM	DCR-TCAM
Bit length		32-bit	
Number of transistors	TCAM	640	-
	BCAM	-	384
	First "X" position	-	40
	Series nMOS transistors	64	32
	2-level bypass	-	56
	Thermometer decoders	-	62
	MLSA	5	40
	AND_BML	-	10
Total		709	624 (88.01%)

Tab. 6. Number of transistors of CV-TCAM and the design with 32-bit memory bank.

	JSSC 2013 [22]	JSSC 2011 [23]	JSSC 2015 [8]	JSSC 2018 [3]	This work
Technology	65nm CMOS	65nm CMOS	65nm CMOS	65nm CMOS	65nm CMOS
Supply volage	1 (V)	1 (V)	1.2 (V)	1.2 (V)	1.2 (V)
CAM type	NOR TCAM	NAND TCAM	NAND TCAM	NAND TCAM	NAND TCAM
Configuration	128K × 128b	256 × 144b	256 × 128b	256 × 128b	256 × 128b
No. of stored words - max	128K (100%)	256 (100%)	672 (262.5%)	672 (262.5%)	1024 (400%)
Effective area/word - max	52.7%	150.27%	42.58%	39.21%	28.53%
Energy/bit/search [fJ]	1.98	0.165	0.41	0.41	0.234
Effective energy/word/search [fJ]	285	24	63	20	7.55
Max. clock freq. [MHz]	250	400	330	330	289

Tab. 7. Performance of the design compared to others TCAM.

4. Future Work

Numerous adaptations, tests, and experiments, including techniques aimed at reducing power consumption and specific test cases involving varied input storage and search data for IPv6 address lookup tables, have been deferred to future work due to time constraints. Experiments using real data are notably time-intensive, often requiring several days to complete a single run. Future research will focus on a more detailed exploration of power-saving techniques, particularly through modifications in the match line architecture, where selective match line discharge can enhance power efficiency. Testing such large-scale designs demands significant processing time in the Virtuoso tool to validate functionality and efficiency, given the constraints in input data signals. Additionally, memristor-based TCAMs [24–27] will be considered due to their potential for improved energy efficiency and reduced area footprint.

5. Conclusion

In this paper, the low-area and enhanced capacity TCAM for the IPv6 lookup table using the DCR scheme and DR technique is proposed. For the low-area technique, the conventional TCAM needs $2N$ -bit memory cells for N -bit stored data, which contains a core cell (used to store data) and a mask cell (used to identify Don't Care). However, the DCR scheme applied to the design (focus on the storing data component) encodes N -bit mask cells into $\log_2 N$ -bit memory cells for storing the first "X" position, which performs the function of "X"s in the TCAM by using additional decoders and bypass transistors. BCAM replaces the TCAM, reducing the number of transistors used in the design. For enhanced capacity technique, the proposed data-relocation technique increases the number of IP addresses stored in the TCAM lookup table by relocating the data in the prefix bits into "X" cells. There can be four types of banks according to the stored bits. The Type 0 bank is empty. The Type 1 and Type 2 banks store four 32-bit words and two 64-bit words instead of a 128-bit word in the Type 3 bank, respectively. Therefore, the Type 1 and Type 2 banks store four- and

two-times larger IP addresses than the conventional TCAM storing 128-bit words. A 256×128 -bit lookup table utilizes a 1.2 V, 65 nm CMOS process. The design improves the area cost (number of transistors) by reducing 5.98% for one memory bank (32×128 -bit) and 5.93% for the 256×128 -bit lookup table design. The effective Area-Cost per stored IP parameter is also improved with the increase of 71.62% with the average distribution of stored IPv6 in the lookup table for one memory bank (32×128 -bit) and 71.47% with an average distribution of stored IPv6 in lookup for the 256×128 -bit lookup table design.

Acknowledgments

We acknowledge the support of time and facilities from Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for this study.

References

- [1] AKHBARIZADEH, M. J., NOURANI, M. Hardware-based IP routing using partitioned lookup table. *IEEE/ACM Transactions on Networking*, 2005, vol. 13, no. 4, p. 769–781. DOI: 10.1109/TNET.2005.852885
- [2] PAGIAMTZIS, K., SHEIKHOESLAMI, A. Content-addressable memory (CAM) circuits and architectures: A tutorial and survey. *IEEE Journal of Solid-State Circuits*, 2006, vol. 41, no. 3, p. 712–727. DOI: 10.1109/JSSC.2005.864128
- [3] YANG, B.-D. Low-power effective memory-size expanded TCAM using data-relocation scheme. *IEEE Journal of Solid-State Circuits*, 2015, vol. 50, no. 10, p. 2441–2450. DOI: 10.1109/JSSC.2015.2457908
- [4] WOO, K.-C., YANG, B.-D. Low-area TCAM using a Don't Care reduction scheme. *IEEE Journal of Solid-State Circuits*, 2018, vol. 53, no. 8, p. 2427–2433. DOI: 10.1109/JSSC.2018.2822696
- [5] ARSOVSKI, I., HEBIG, T., DOBSON, D., et al. A 32 nm 0.58-fJ/bit/search 1-GHz ternary content addressable memory compiler using silicon-aware early-predict late-correct sensing with embedded deep-trench capacitor noise mitigation. *IEEE Journal of Solid-State Circuits*, 2013, vol. 48, no. 4, p. 932–939. DOI: 10.1109/JSSC.2013.2239092

- [6] HUANG, P.-T., HWANG, W. A 65 nm 0.165 fJ/bit/search 256×144 TCAM macro design for IPv6 lookup tables. *IEEE Journal of Solid-State Circuits*, 2011, vol. 46, no. 2, p. 507–519. DOI: 10.1109/JSSC.2010.2082270
- [7] MIYATAKE, H., TANAKA, M., MORI, Y. A design for high-speed low-power CMOS fully parallel content-addressable memory macros. *IEEE Journal of Solid-State Circuits*, 2001, vol. 36, no. 6, p. 956–968. DOI: 10.1109/4.924858
- [8] YANG, B.-D., LEE, Y.-K. A low-power CAM using pulsed NAND-NOR match-line and charge-recycling search-line driver. *IEEE Journal of Solid-State Circuits*, 2005, vol. 40, no. 8, p. 1736–1744. DOI: 10.1109/JSSC.2005.852028
- [9] YANG, B.-D., LEE, Y.-K., SUNG, S.-W., et al. A low power content addressable memory using low swing search lines. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2011, vol. 58, no. 12, p. 2849–2858. DOI: 10.1109/TCSI.2011.2158703
- [10] WANG, C.-C., CHENG, C.-J., CHEN, T.-F., et al. An adaptively dividable dual-port BITCAM for virus-detection processors in mobile devices. In *IEEE International Solid-State Circuits Conference – Digest of Technical Papers*. San Francisco (CA, USA), 2008, p. 390–622. DOI: 10.1109/ISSCC.2008.4523221
- [11] ARSOVSKI, I., CHANDLER, T., SHEIKHOLESAMI, A. A ternary content-addressable memory (TCAM) based on 4T static storage and including a current-race sensing scheme. *IEEE Journal of Solid-State Circuits*, 2003, vol. 38, no. 1, p. 155–158. DOI: 10.1109/JSSC.2002.806264
- [12] ARSOVSKI, I., CHANDLER, T., SHEIKHOLESAMI, A. A mismatch-dependent power allocation technique for match-line sensing in content-addressable memories. *IEEE Journal of Solid-State Circuits*, 2003, vol. 38, no. 11, p. 1958–1966. DOI: 10.1109/JSSC.2003.818139
- [13] MOHAN, N., SACHDEV, M. Low-capacitance and charge-shared match lines for low-energy high-performance TCAMs. *IEEE Journal of Solid-State Circuits*, 2007, vol. 42, no. 9, p. 2054–2060. DOI: 10.1109/JSSC.2007.903089
- [14] BAEG, S. Low-power ternary content-addressable memory design using a segmented match line. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2008, vol. 55, no. 6, p. 1485–1494. DOI: 10.1109/TCSI.2008.916624
- [15] LIN, C.-S., CHANG, J.-C., LIU, B.-D. A low-power precomputation-based fully parallel content-addressable memory. *IEEE Journal of Solid-State Circuits*, 2003, vol. 38, no. 4, p. 654–662. DOI: 10.1109/JSSC.2003.809515
- [16] PAGIAMTZIS, K., SHEIKHOLESAMI, A. A low-power content-addressable memory (CAM) using pipelined hierarchical search scheme. *IEEE Journal of Solid-State Circuits*, 2004, vol. 39, no. 9, p. 1512–1519. DOI: 10.1109/JSSC.2004.831433
- [17] CHANG, Y.-J. A high-performance and energy-efficient TCAM design for IP-address lookup. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2009, vol. 56, no. 6, p. 479–483. DOI: 10.1109/TCSII.2009.2020935
- [18] LI, H.-Y., CHEN, C.-C., WANG, J.-S., et al. An AND-type match-line scheme for high-performance energy-efficient content addressable memories. *IEEE Journal of Solid-State Circuits*, 2006, vol. 41, no. 5, p. 1108–1119. DOI: 10.1109/JSSC.2006.872719
- [19] KADIR, R. B., ANWAR, A. S., MUNTASIM-UL-HAQUE. Analysis of charge-shared matchline sensing schemes and current race scheme in high-speed ternary content addressable memory (TCAM). In *International Conference on Innovations in Science, Engineering and Technology (ICISSET)*. Dhaka (Bangladesh), 2016, p. 1–4. DOI: 10.1109/ICISSET.2016.7856490
- [20] SHAFAL, F., SCHULTZ, K. J., GIBSON, G. F. R., et al. Fully parallel 30-MHz, 2.5-Mb CAM. *IEEE Journal of Solid-State Circuits*, 1998, vol. 33, no. 11, p. 1690–1696. DOI: 10.1109/4.726560
- [21] HANZAWA, S., SAKATA, T., KAJIGAYA, K., et al. A large-scale and low-power CAM architecture featuring a one-hot-spot block code for IP-address lookup in a network router. *IEEE Journal of Solid-State Circuits*, 2005, vol. 40, no. 4, p. 853–861. DOI: 10.1109/JSSC.2005.845554
- [22] HAYASHI, I., AMANO, T., WATANABE, N., et al. A 250-MHz 18-Mb full ternary CAM with low-voltage matchline sensing scheme in 65-nm CMOS. *IEEE Journal of Solid-State Circuits*, 2013, vol. 48, no. 11, p. 2671–2680. DOI: 10.1109/JSSC.2013.2274888
- [23] WANG, C.-C., WANG, J.-S., YEH, C. High-speed and low-power design techniques for TCAM macros. *IEEE Journal of Solid-State Circuits*, 2008, vol. 43, no. 2, p. 530–540. DOI: 10.1109/JSSC.2007.914330
- [24] SALEH, S., GOOSSENS, A. H., BANERJEE, T., et al. TCAM-M-CogniGron: Energy efficient memristor-based TCAM for match-action processing. In *IEEE International Conference on Rebooting Computing (ICRC)*. San Francisco (CA, USA), 2022, p. 89–99. DOI: 10.1109/ICRC57508.2022.00013
- [25] CHOWDHURY, Z. I., RESCH, S., CILASUN, H., et al. CAMEleon: Reconfigurable B(T)CAM in computational RAM. In *Proceedings of the Great Lakes Symposium on VLSI (GLSVLSI)*. Virtual Event, 2021, p. 57–63. DOI: 10.1145/3453688.3461507
- [26] KHAN, M. R., RASHID, A. H. Memristor-transistor hybrid ternary content addressable memory using ternary memristive memory cell. *IET Circuits, Devices & Systems*, 2021, vol. 15, no. 7, p. 619–629. DOI: 10.1049/CDS2.12057
- [27] KIM, S. M., KIM, K. M., CHOI, J. H., et al. A digital processing in memory architecture using TCAM for rapid learning and inference based on a spike location dependent plasticity. *IEEE Access*, 2023, vol. 11, p. 3416–3430. DOI: 10.1109/ACCESS.2023.3234323

About the Authors . . .

Anh PHAM is currently pursuing a B.S. degree in Electrical and Electronics Engineering with Ho Chi Minh University of Technology (HCMUT), VNU-HCM. His research interests include digital circuit design and efficient algorithms. He is the first author of this paper.

Doanh BUI is a master student who also received his B.S. degree in Electronics and Telecommunications Engineering from Ho Chi Minh City University of Technology (HCMUT),

VNU-HCM, in 2021. His research interests include CMOS digital circuit design, physical design in IC design. He is the second author of this paper.

Phuc Thien Phan NGUYEN is a master student who also received his B.S. degree in Electronics and Telecommunications Engineering from Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, in 2023. His research interests include CMOS digital circuit design, transistor and memristor modeling, and neuro-inspired engineering. He can be contacted at email: nptphuc.sdh232@hcmut.edu.vn. He is the third author of this paper.

Linh TRAN received the B.S. degree in Electrical and Computer Engineering from University of Illinois, Urbana – Champaign (2005), M.S. and Ph.D. in Computer Engineering from Portland State University (2006, 2015). Currently, he is working as Lecturer at the Faculty of Electrical-Electronics Engineering, Ho Chi Minh City University of Technology – VNU-HCM. His research interests include quantum/reversible logic synthesis, computer architecture, hardware-software co-design, efficient algorithms and hardware design targeting FPGAs and deep learning. He can be contacted at email: linhtran@hcmut.edu.vn. He is a corresponding author in this paper.