A content-addressing memory (CAM) architecture and its reasoning algorithm

F. Wang & N. Helian
School of Computing, Communication and Mathematics
London Metropolitan University, U.K.

Abstract

In this paper we propose a CAM architecture and its corresponding reasoning algorithm, which is novel in three aspects compared with the conventional CAM structure requiring a considerable number of logic circuits for matching within its individual cells and in its peripherals. The architecture design issues and its corresponding reasoning algorithm have been discussed. In our opinion, this new associative memory presents one with a broad range of trade-offs based on cost requirements, speed requirements, and storage and logic technology parameters.

1. Introduction

Many data processing applications, like databases, data warehouses, data mining, OLAP (On-Line Analytical Processing), require the search of items in a table stored in memory[1][2]. The time required to find an item stored in memory can be reduced considerably if stored data can be identified for access by the content of the data itself rather than by an address. A memory unit accessed by content rather than location is called an associative memory or content addressable memory (CAM). This type of memory is accessed simultaneously and in parallel on the basis of data content rather than by specific address or location. A considerable amount of research activity has been conducted to include CAM in applications[3]. In this paper we propose a novel CAM architecture and its corresponding reasoning algorithm. The architecture design issues and its corresponding reasoning algorithm will be discussed.

2. Architecture design

We will use the magnetic core as an exemplified physical storage mechanism to demonstrate a new CAM operation in this paper. The principle can be apparently
applicable to other types of memories. The memory consists of a cross-point array of magnetic cores (ferrite toroids). As shown in Fig.1, a core is required for each bit of memory and has three wires passing through it, providing the means to select and detect the contents of each bit. The main property that makes core memory work is the hysteresis of the magnetic material used to make the toroid. Only a magnetic field over a certain intensity (generated by the wires through the core) will cause the core to change its magnetic polarity (or state from '0' to '1'). In a traditional random access manner, to select a memory location, one of the X and one of the Y lines are driven with half the current required to cause this change. Only the combined magnetic field generated at the intersection the driven X and Y lines is sufficient to change the state of the bit. All other memory locations will at worst, only have half the required magnetic pulse, and will not be affected. Unlike semiconductor memory, core memory is said to be non-volatile and will not lose its contents when the power is removed.

Fig.2: The 0th phase of the scan for “1” matching along the main diagonal of the 5x5 matrix.
For content-addressable purpose, we reverse the use of the traditional cross-point architecture by introducing a third line for each bit. This line is used to flow a pulse of exciting current to test if an argument matches the contents of the cell or not. When “0” is stored as shown in Fig.1(a), the magnetic field induced by the pulse current I (Lentz’ law) is in the same direction as the magnetic moment “0” and the moment stays still as before. No voltage response will be generated (Faraday’s law), which means no match for “1” occurs; when “1” is stored as shown in Fig.1(b), the magnetic field is in the opposite direction of the magnetic moment thereby rotating the magnetic moment “1” by an angle. The induced voltages by that change will appear at both X and Y terminals, meaning a match for stored “1” at the intersection occurs.

![Diagram](image)

Fig.3: The 1\textsuperscript{st} phase of the scan for “1” matching along the diagonals whose diagonal distance is 2.

The strength of the pulse current is carefully chosen at an appropriate level in order for the magnetic field H to rotate the magnetic moment “1” by an angle less than 90°, otherwise causing an unexpected destructive test (the memory contents is lost after the location is tested), which needs to be followed by a write procedure to restore its previous content. After the application of the magnetic field, the magnetic moment would spring back to its previous position, i.e. the circumferential easy axis. The above is an exemplified search for “1” and it is possible to realize a search for “0” by simply opposing the polarity of the exciting current.

The matching scan for the entire array is divided into several phases, which starts with the main diagonal test (0\textsuperscript{th} phase). The 0\textsuperscript{th} phase of the scan for “1” matching along the main diagonal of a 5x5 array is shown in Fig.2. The cells in the array are marked by the letter C with two subscripts. The first subscript i gives the X coordinate and the second j specifies the Y coordinate. Five bits are being tested simultaneously in the 0\textsuperscript{th} phase.
The “1” cell sends out a pulse of voltage at both X and Y terminals whereas the “0” cell does not respond. Logically, by monitoring the output of X and Y lines, we are able to know which cell contains “1” in the main diagonal of the matrix.

The 1st phase of the scan for “1” matching along the four diagonals is shown in Fig.3. Two pairs of diagonals have been chosen, each of which contains a sub-main diagonal and a vertex. The total number of the cells being scanned simultaneously at this stage is ten (in either pair is five). At first let’s define a concept of diagonal distance. The so-called diagonal distance DD is defined as the separation distance between the two diagonals in the same triangle of the matrix. As will be illustrated in the following sections, the diagonal distance will be monitored to decide if the scan procedure is finished or not. In Fig.3 the diagonal distance DD is 2. Similar to Fig.2, the “1” cell shows its existence by sending out a pulse of voltage at both X and Y terminals.

Fig.4 shows the 2nd phase along the diagonals whose diagonal distance is 1. 2x5 bits are being tested simultaneously at this stage. This is the last phase of the scan for “1” matching. There is a possibility, in the 1st and 2nd phase, that two (but not more than two) pulses from two cells may join together on the way propagating to either X or Y terminals thereby generating a double-amplitude resulting pulse (denoted as “2”) at the terminals. The overall architecture integrating the above 0th, 1st and 2nd phases is shown in Fig.5. The “comb”-shaped circuit carries the exciting currents to each cell. The delay lines are imposed to slightly shift different phases to distinguish them during a scan cycle. In this figure, 1D represents one unit of time delay and 2D represents two units. The corresponding time sequence of a scan against 5x5 cell matrix is shown in Fig.6. The single-amplitude pulse at the terminal contributed by only one cell is denoted as “1” and no pulse is denoted as “0”. As mentioned above, two (but not
more than two) pulses from two cells may join together on the way propagating to either X or Y terminals thereby generating a double-amplitude pulse at the terminals. This double-amplitude pulse is denoted as “2”. In the example of 5x5 array, three phases is needed to scan all the cells and the corresponding time required for a full scan is 3D. The time delay constant of the delay lines should be reduced to its minimum, as long as different phases can be distinguished against the time axis. Based on those obtained terminal voltages, we are in a position to determine which cell in the array matches the pre-set argument. This is the task of the reasoning algorithm.

3. Reasoning algorithm
The reasoning algorithm enables us to reason the internal contents of the cells in a black box (a CAM) based on the external terminal values. The reasoning procedure in either Fig.3 or Fig.4 to determine which cell matches the pre-set argument (“1” in our example) is a little complicated compared with the main diagonal scan in Fig.2. In algebraic, to solve n unknown variables needs n equations. Here in our example, each equation corresponds to a terminal output and is derived from the fact that the terminal voltage is equal to the sum of the contribution from those two cells in that terminal line. In the example of 5x5 array, we have 5x2=10 equations in each phase, that are just sufficient to find out the contents of those cells in the corresponding diagonals. The exception is 0th
phase, the number of the equations is 10, which is redundant to solve just 5 unknown cell contents in the main diagonal. To maintain the integrity of the algorithm, we just include 5 cells under test in the 0th phase.

We can avoid to solve equation group to find out the contents of the cell by a reasoning algorithm. Fig.7 shows the reasoning algorithm for matching based on the collective terminal values in an example of 5x5 array.

Logically, by monitoring and analyzing the output of X and Y lines, we are able to know which cell contains “1” in the main diagonal of the matrix. The reasoning procedure in the 0th phase, as shown in Fig.7(a), is straightforward and can be expressed by $C_{00}=X_0=Y_0=1$, $C_{11}=X_1=Y_1=1$, $C_{22}=X_2=Y_2=1$, $C_{33}=X_3=Y_3=0$ and $C_{44}=X_4=Y_4=0$.

In the 1st phase let’s start the reasoning from $C_{04}$. Since $X_0=C_{01}+C_{04}=0$, $C_{01}=C_{04}=0$; since $Y_1=C_{01}+C_{21}=1$, $C_{21}=1$; since $X_2=C_{21}+C_{23}=1$, $C_{23}=0$; since $Y_3=C_{23}+C_{43}=1$, $C_{43}=1$; since $X_4=C_{43}+C_{40}=1$, $C_{40}=0$; since $Y_0=C_{40}+C_{10}=0$, $C_{10}=0$; since $X_1=C_{10}+C_{12}=0$, $C_{12}=0$; since $Y_2=C_{12}+C_{32}=0$, $C_{32}=0$; since $X_3=C_{32}+C_{34}=1$, $C_{34}=1$;
Fig. 8: An example of guess failure.

C_{43}=1; since Y_3=C_{34}+C_{04}=1, C_{04}=0 (just confirmation). The detailed reasoning procedure has been shown in Fig. 7(b). Similarly, the 2nd phase can be deduced, as shown in Fig. 7(c), which is the last step of the scan.

Fig. 8 gives an example of starting the reasoning with wrong assumption resulting in a contradiction in the exploration. Since X_4=C_{40}+C_{34}=1, there are two possibilities: (1) C_{40}=0 and C_{34}=1 (the fact); (2) C_{40}=1 and C_{34}=0 (not the fact). Let’s deliberately start with an assumption C_{40}=1 and C_{34}=0. Since Y_3=1, C_{23}=1. Similarly C_{21}=0, C_{01}=1. However C_{01}=1 is contradictory with the precondition X_0=C_{01}+C_{04}=0, which implies the assumption of C_{01}=1 is incorrect.

Considering a NxN array as shown in Fig. 9, where N=2n+1 is an odd number (n is an integer for 0,1,2,..., \( \frac{N-1}{2} \)). Note that we take N as an odd number here because we manage to get pairs of diagonals except for the main diagonal. It does not matter if the size N of the array is an even number. In the latter case, the algorithm needs to be adjusted correspondingly.

The matching scan for the entire array is divided into n+1 phases, where n=\( \frac{N-1}{2} \), which starts with the main diagonal test (0th phase). The 0th phase of the scan for “1” matching is carried out along the main diagonal of a NxN array. N bits are being tested simultaneously at this stage. The “1” cell sends out a pulse of voltage at both X and Y terminals whereas the “0” cell does not respond. In other words, by monitoring and analyzing the output of X and Y lines, we are able to know which cell contains “1” in the main diagonal of the matrix. The reasoning procedure is straightforward and can be expressed by the below formula:

\[ C_i=Y_i \quad \text{for } i=0,1,2,...,N-1. \]  

(1)

The kth phase of the scan for “1” matching along the four diagonals is shown in Fig. 9. Two pairs of diagonals have been chosen: one pair contains a longer diagonal whose starting point is C_{0k} and a shorter diagonal whose starting point
is $C_{0, N-k}$, the other pair contains a longer diagonal whose starting point is $C_{k, 0}$ and a shorter diagonal whose starting point is $C_{N-k, 0}$. The so-called diagonal distance DD is defined as the separation distance between the two diagonals in the same triangle of the matrix. The diagonal distance will be monitored to decide if the scan procedure is finished. In the $k$'th phase the diagonal distance DD is $N-2k$. In the $n+1$'th phase, the diagonal distance DD is decremented to 1 and the whole $N \times N$ matrix has been fully scanned.

In each phase except for the $0$'th phase, $2N$ cells are being tested simultaneously. The “1” cell shows its existence by sending out a pulse of voltage at both X and Y terminals. Two (but not more than two) pulses from two cells may join together on the way propagating to either X or Y terminals thereby generating a double-amplitude resulting pulse at the terminals. Any X or Y terminal value is equal to the sum of the contributions from the two and just two cells in that X or Y line. That is, $X_i = C_{i, Y \text{-first}} + C_{i, Y \text{-second}}$ or $Y_j = C_{X \text{-first}, j} + C_{X \text{-second}, j}$. Concretely, either X or Y coordinate is divided into three zones, which implies different $C_{i, Y \text{-first}} + C_{i, Y \text{-second}}$ combination or $C_{X \text{-first}, j} + C_{X \text{-second}, j}$ combination.

In Zone 1, where $0 \leq i < k$, we have

$$X_i = C_{i, k+i} + C_{i, N-k+i}$$  \hspace{1cm} (2)

In Zone 2, where $k \leq i \leq N-k-1$, we have

$$X_i = C_{i, i-k} + C_{i, k+i}$$  \hspace{1cm} (3)

In Zone 3, where $N-k-1 < i \leq N-1$, we have

$$X_i = C_{i, i-N+k} + C_{i, k}$$  \hspace{1cm} (4)

Similar zone division and formula apply to Y coordinate.

The general reasoning algorithm is based on several deductions. Let’s state them one by one.

**Deduction 1**

Any X or Y terminal value, except for the 0th phase, is equal to the sum of the contributions from the two and just two cells in that X or Y line. That is, $X_i = C_{i, Y \text{-first}} + C_{i, Y \text{-second}}$ or $Y_j = C_{X \text{-first}, j} + C_{X \text{-second}, j}$.

In other words one cannot find an exception that just one cell or more than two cells contribute to the terminal value (including 0 contribution). As can be seen shortly, this deduction is important to obtain Deduction 2.

**Deduction 2**

Any X or Y terminal value is equal to 0, 1 or 2. That is, $X_i = 0, 1, 2$ and $Y_j = 0, 1, 2$. 
As mentioned above, a single-amplitude pulse at the terminal contributed by only one cell is denoted as “1” and no pulse is denoted as “0” and a double-amplitude pulse is denoted as “2” at the terminals. Since Deduction 1 states any X or Y terminal value is equal to the sum of the contributions from the two and just two cells in that X or Y line, this deduction is obvious.

Deduction 3
If \( X_i = 0 \), then \( C_{i,Y \text{-first}} = 0 \) and \( C_{i,Y \text{-second}} = 0 \). Similarly, if \( Y_j = 0 \), then \( C_{X \text{-first},j} = 0 \) and \( C_{X \text{-second},j} = 0 \).

Deduction 4
If \( X_i = 2 \), then \( C_{i,Y \text{-first}} = 1 \) and \( C_{i,Y \text{-second}} = 1 \). Similarly, if \( Y_j = 2 \), then \( C_{X \text{-first},j} = 1 \) and \( C_{X \text{-second},j} = 1 \).

Deduction 3 and 4 are quite obvious but very useful in reasoning. In the above two cases, the internal contents of the two corresponding cells have been certainly and uniquely determined by the external terminal output value. It is wise to start the reasoning procedure from a terminal with the output of “0” or “2”.

Deduction 5
If \( X_i = 1 \), then there are two possibilities: (1) \( C_{i,Y \text{-first}} = 0 \) and \( C_{i,Y \text{-second}} = 1 \); (2) \( C_{i,Y \text{-first}} = 1 \) and \( C_{i,Y \text{-second}} = 0 \). Similarly, if \( Y_j = 1 \), then there are two possibilities: (1) \( C_{X \text{-first},j} = 0 \) and \( C_{X \text{-second},j} = 1 \); (2) \( C_{X \text{-first},j} = 1 \) and \( C_{X \text{-second},j} = 0 \).

This is a little awkward situation to cope with. As illustrated in a 5x5 example shown in Fig.8, we can use a “contradiction” method to start the reasoning, i.e. randomly choosing a possibility to start, alternatively using X, Y conditions to explore ahead, if it is a wrong assumption, sooner or later, one would encounter a contradictory embarrassment, then giving up that attempt and re-start with another possibility. Since there are just two possibilities, the new attempt would be certainly successful.

Deduction 6
The total number of the cells in either pair of diagonals is a fixed number: N.

The significance of this fact is that we can store the words of N bits each diagonally, which would be a novel data storage method compared with traditional manner and bring a couple of advantages to us.

Fig.10 shows flowchart of the reasoning algorithm. This is a recursive procedure of probing along the X-direction and the Y-direction alternatively, which integrates all the above six deductions. Regarding to the box of “Choosing different X2(Y2) based on whether j(i) belongs to Zone1, Zone2 or Zone3”, see the above definitions and the corresponding equation (2), (3) and (4) for details. \( C_{i,Y2} = C_{0,N_k} \) and \( C_{X2,j} = C_{0,N_k} \) is used to judge if a reasoning procedure returns to its starting point \( C_{0,N_k} \) to finish the reasoning procedure in the k’th phase. The
diagonal distance will be monitored to decide if the scan procedure is finished or not. In the k'th phase the diagonal distance DD is N-2k. In the n+1'th phase when the diagonal distance DD is decremented to 1, the whole NxN matrix has been fully scanned.

4. CAM Organization
An exemplified block diagram of this new CAM is shown in Fig.11. It consists of a 5x5 memory array and a comparison logic for 5 words with 5 bits per word. The argument register A and key register K each have 5 bits of a word. The match register M has 5 bits, one for each memory word. The key register provides a mask for choosing a particular field or key in the argument word or database record. The match logic compares the content of the storage cell with the corresponding unmarked bit of the argument and provides an output for the decision logic that sets the bit in Mi. This CAM has a write capability for storing the information to be searched. This makes the device a random-access memory for writing and a content-addressable memory for reading.
5. Conclusion

In this paper we propose a CAM architecture and its corresponding reasoning algorithm, which is novel in three aspects compared with the conventional CAM structure[1][2]. First, the integration density is high because the memory cell matrix is simply structured with a conventional cross-point array. Secondly, a NxN cell matrix can be fully scanned for matching in just one memory cycle consisting of N/2 phases. Thirdly, this memory is of great flexibility in use since it is a mixture of RAM (Random Access Memory) and CAM.

References

[1] Zaki and Mannaty, The hardware software trends in data mining high-performace architectures, 4th European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD'00), Lyon, France, September 13-16, 2000

