A 1.2-Million Transistor, 33-MHz, 20-b Dictionary Search Processor (DISP) ULSI with a 160-kb CAM

MASATO MOTOMURA, JUN TOYOURA, KAZUMI HIRATA, HIDEYUKI OOKA, HACHIRO YAMADA, AND TADAYOSHI ENOMOTO, MEMBER, IEEE

Abstract—**A. 1.2-million transistor, 33-MHz, 20-b dictionary search processor (DISP) ULSI has been developed using a 0.8-μm, triple-layer-Al, CMOS fabrication technology. A 13.02×12.51-mm² chip contains both a specially developed 160-kb content addressable memory (CAM) and cellular automaton processor (CAP). The CAM capacity is 20 times larger than a previously reported CAM. A single DISP chip can store a maximum of 2048 words, and performs dictionary search in various search modes, including an "approximate word search." The character input rate for the dictionary search operation is 33-million characters per second. The DISP typically consumes 800 mW at a supply voltage of 5 V. A high-speed and functional 50000-word dictionary search system can be built with 25 DISP chips arranged in parallel, to play an important role in natural language processing.**

I. INTRODUCTION

DICTIONARY search is a key operation in natural language processing, including machine translation [1] and voice or character recognition [2]. For example, in the case of English to Japanese translation, the Japanese translation words for each English word in input texts must first be retrieved from a dictionary [1]. In such applications, the required dictionary size may grow to several tens of thousands of words. Thus, the most important issue in constructing a dictionary search system is both how to increase the vocabulary and how to speed up the search process for such a huge dictionary. In addition, character errors in input words due to misspellings or pattern recognition errors are inevitable in such applications. Thus, what is required is not only a search for a stored word that exactly matches an input word ("exact word search"), but also a search for stored words that approximately match an input word ("approximate word search") [3].

Conventional dictionary search systems are essentially software systems. For "exact word search," stored words in the dictionary are iteratively read out and compared with an input word. The iteration is performed according to some sorting algorithm, for example, a binary tree. When an "exact" match between the input word and a stored word is accomplished, the address of the matched stored word is used to obtain a set of translation words. For a practical dictionary containing 50000 words, the average number of required iterations is approximately 15, which would require several hundred microseconds. On the other hand, for the "approximate word search," a search must be executed through all the words stored in the dictionary for the word closest to the input word with character errors. This process requires several tens of milliseconds, which is too slow for any practical natural language processing. Furthermore, there are many other serious disadvantages to the use of conventional dictionary search systems, including the complicated softwares required and the difficulty of adding new stored words, since all words must be sorted according to a specified sorting rule.

Content addressable memories (CAM's) [4]–[8] are able to simultaneously compare an input character with all of their stored characters at once. Thus, CAM's are powerful tools for dictionary search systems, and can drastically reduce the time required for an "exact word search" to less than 1 μs. However, the memory capacity for CAM's reported to date, 20 kb at most [8], is far too small for dictionary storage. This is why the use of CAM's has been restricted to areas where large memory capacity is not required, such as text database retrieval [5], [7], [9], address filtering in local area networks [6], etc. Moreover, conventional CAM's do not have an "approximate word search" function, except in [5].

In an attempt to resolve these problems encountered in trying to construct practical dictionary search systems, we have developed the dictionary search processor (DISP) ULSI [10], which contains a 160-kb CAM and a high-speed and compact cellular automaton processor (CAP). CAM capacity is 10 to 20 times larger than that of previously reported CAM's, so that a 50000-word dictionary search system can be constructed with only 25 of them.

Manuscript received April 12, 1990; revised June 21, 1990.

M. Motomura, J. Toyoura, K. Hirata, H. Yamada, and T. Enomoto are with the System Ultra-LSI Research Laboratory, Microelectronics Research Laboratories, NEC Corporation, 1120 Shimokuzawa, Sagamihara-shi, Kanagawa-ken 229, Japan.

H. Ooka is with the VLSI Development Division, NEC Corporation, 1120 Shimokuzawa, Sagamihara-shi, Kanagawa-ken 229, Japan.

IEEE Log Number 9038258.

0018-9200/90/1000-1158$01.00

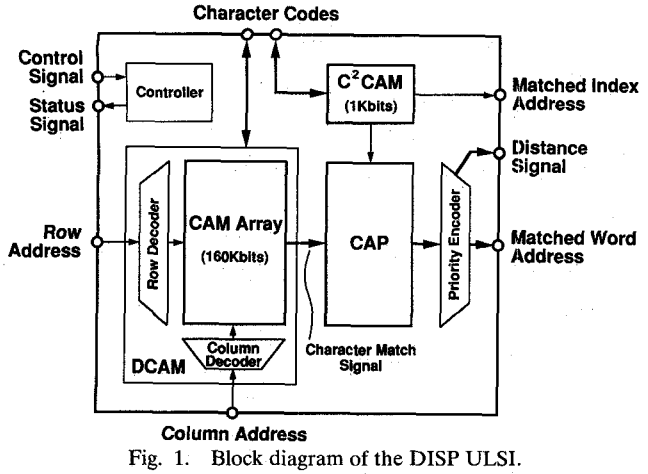


Fig. 1. Block diagram of the DISP ULSI.

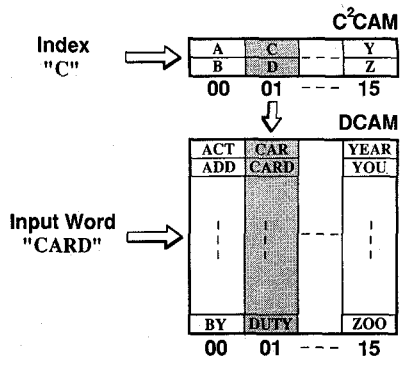


Fig. 2. New CAM architecture for the DISP. The DCAM stores words belonging to 16 different categories, and the C^2CAM stores indexes for each category. The C^2CAM classifies an input word into one of 16 categories. The DCAM compares an input word with words stored in the corresponding category.

In this paper, the architecture and performance for the DISP ULSI are presented. The architecture and circuits for producing extremely large capacity CAM's as well as the CAP architecture are also described in detail.

II. DISP Architecture

In order to perform the dictionary search, the DISP must at least execute the following steps:

1) store words,
2) compare input words with stored words character by character,
3) perform word searches either "exactly" or "approximately," and
4) produce the addresses of matched stored words.

As shown in Fig. 1, the DISP ULSI integrates a 160-kb data CAM (DCAM) and a 1-kb control-code CAM (C^2CAM) for steps 1) and 2), a cellular automaton processor (CAP) for step 3), a priority encoder for step 4), and a controller.

A maximum of 2048 words, which are stored in the DCAM, are classified into 16 different categories. An input word is first classified into one of those 16 categories by the C^2CAM. Then the input word is compared with all stored words in the selected category character by character. A character match signal for each character comparison is then produced. Either "exact" or "approximate" word searches are conducted by the CAP, using the character match signals obtained. Both the addresses of the matched words and their "distance" signal, i.e., the degree of their dissimilarity to the input word, are obtained from the priority encoder.

III. Content Addressable Memory (CAM)

For a dictionary search system, the larger the CAM capacity, the greater the number of storable words, but conventional CAM's are too small for dictionary search systems. In order to simultaneously compare each input character with all stored characters, single memory cells of conventional CAM's must be composed of a SRAM cell and a MATCH circuit, whose area is three times larger than that of the SRAM cell itself. Thus, conventional CAM cells are roughly four times larger than conventional SRAM cells alone.

A. New CAM Architecture

In order to reduce CAM cell area, thus increasing CAM capacity, the massive comparison work performed by conventional CAM's should be avoided. Thus emerges an important idea: "categorization." Let the stored words be classified into plural categories according to some rule. Then it will be sufficient to compare an input word only to stored words which belong to that same category. In this case, it will not be necessary for every CAM cell to have a MATCH circuit. Instead, a single MATCH circuit can be shared with plural SRAM cells, each of which corresponds to different categories, and used in a time-shared fashion. This allows a new CAM structure with greatly increased CAM capacity.

The 160-kb data CAM (DCAM) and the 1-kb control code CAM (C^2CAM) are designed according to the new CAM architecture described above. The DCAM stores a maximum of 2048 words, which belong to 16 different categories. The C^2CAM stores indexes for each category. These indexes are used for selecting the category appropriate for each input word. The category selection is executed in the DISP by entering the matched index address obtained from the C^2CAM as the column address of the DCAM (see Fig. 1). Fig. 2 shows an example of the operation of the DCAM and the C^2CAM. In this example, words stored in the DCAM are classified into categories 00 to 15 according to their first characters, e.g., words starting with characters *A* or *B* are classified into category 00, and so on. The C^2CAM stores two characters as indexes for each category. The index, i.e., the first character *C* of the input word "CARD," is first applied to the C^2CAM, where an "index search" isperformed. Parallel comparison between the input character *C* and all the stored indexes is carried out, and a match occurs for category 01, where index *C* is stored. Thus corre-

TABLE I
MEMORY CAPACITY COMPARISON BETWEEN A PREVIOUSLY REPORTED CAM [5] AND THE PRESENT DCAM

	PREVIOUSLY REPORTED CAM	DCAM	IMPROVING FACTOR
DESIGN RULE (μm)	1.6	0.8	4
ARCHITECTURE	CONVENTIONAL	NEWLY DEVELOPED	4
AREA (mm^2)	40	55	1.4
CAPACITY (Kb)	8	160	20

sponding category 01 in the DCAM is selected. Character comparison between the input word "CARD" and the stored words in category 01 is then executed.

B. CAM Capacity Comparison

Table I shows a memory capacity comparison between a previously reported CAM [5] and the present DCAM. The design rule for the CMOS fabrication process is improved from 1.6 to 0.8 μm, resulting in a fourfold increase in CAM capacity. The newly developed CAM architecture reduces cell area to about one-fourth of that of the conventional CAM's. This contributes a further fourfold increase in CAM capacity. By combining these two factors with a slightly increased CAM area, we were able to construct a 160-kb DCAM, with 20 times the capacity of the previously reported CAM.

C. CAM Structure

Fig. 3 is a block diagram of the DCAM, which consists of 20 CAM arrays, 00 to 19. Each CAM array has 512 rows and 16 columns. Individual rows consist of 16 SRAM cells, as well as single READ, MATCH, and WRITE circuits (the WRITE circuit is not shown here). Single columns are selected by means of word lines, and rows by means of row-select lines. Each column stores words belonging to a single category, so that each word stored in the DCAM is classified into one of 16 categories, as previously explained.

One character is stored in 20 SRAM cells, each of which has the same address in all the different arrays. For example, 20 black cells in Fig. 3 store one character. The 20 b for each character location are divided into a 16-b data field and a 4-b tag field. The 16-b data field can accommodate a single 2-byte character code. The 4-b tag field stores a garbage flag which shows whether that character location is used or not, a mask flag which shows whether the character is masked or not in the MATCH operation, and so on. A single column can store 512 characters. Four successive neighboring characters form one group, for a total of 128 groups. A single group can store a single word with a character length of four and less, so that a maximum of 128 words can be stored in one column, and the maximum number of stored words for a single DISP is 2048. Several groups can also be concatenated to store a word with a character length of more than four.

Fig. 4 shows a detailed view of the new CAM structure for a single row. It is almost the same as a conventional SRAM structure except for a transistor $M4$ and a match line. The MATCH circuit, shared by 16 SRAM cells, is constructed simply by adding only one transistor ($M4$) to a conventional READ circuit ($M1$–$M3$). The basic idea for producing such an extremely compact MATCH circuit is this: transistors $M1$ and $M2$, which are used as a differential pair in a conventional READ circuit, can operate as an EXCLUSIVE-OR gate for the MATCH circuit.

D. CAM Operation

The DCAM can perform three operations: WRITE, READ, and MATCH. For the WRITE and READ operations, 20 SRAM cells for a single character are selected by column and row addresses. Characters are then either read or written through data lines 00 to 19, as is usual in conventional SRAM's.

For the MATCH operation, a single column containing 512 characters is selected by a column address, which is produced by the C^{2}CAM as described in Section III-A. For example, a 4-b column address "0001" selects the 512×20 SRAM cells denoted by the hatched lines in Fig. 3. Then, at the MATCH circuits, the 512 stored characters are simultaneously compared with a single input character loaded from the data-line pairs 00 to 19.

Fig. 5 gives a detailed circuit diagram for the MATCH operation, where 20 CAM arrays and 20 MATCH peripheral circuits are shown. Hereafter, we assume a supply voltage of 5 V as logical value ONE, and 0 V as logical value ZERO. During the MATCH operation, transistors $M3$ are kept turned off by setting all row-select lines to ZERO. Let an input datum be ZERO and a stored datum in a selected cell (denoted by hatched lines) be ONE. First, all bit-line pairs and match lines are precharged, while all data-line pairs are discharged. After that, a single column is selected by the word lines, and the stored datum ONE causes the $\overline{\text{bit}}$ line to discharge. At the same time, by setting the $\overline{\text{mask}}$ signal and enable signal to ONE, and the discharge signal to ZERO, the input datum ZERO makes the $\overline{\text{data}}$ line charge up to ONE. Thus, $M1$ and $M4$ turn on and $M2$ turns off, so that a precharged match line discharges to ZERO. On the other hand, when an input datum ONE is applied, the match line maintains precharged level ONE because the low-level data line makes $M4$ turn off. In order to obtain a character match, each bit of an input character and a corresponding bit in the stored character must be the same to keep the match line at ONE. If even one pair of bits was mismatched, the precharged match line would discharge because the corresponding transistor $M4$ turns on, resulting in character mismatch.

In conventional SRAM's, the bit-line voltage swing required is limited to well below 1 V, because this small voltage difference is amplified by the READ circuit. How-

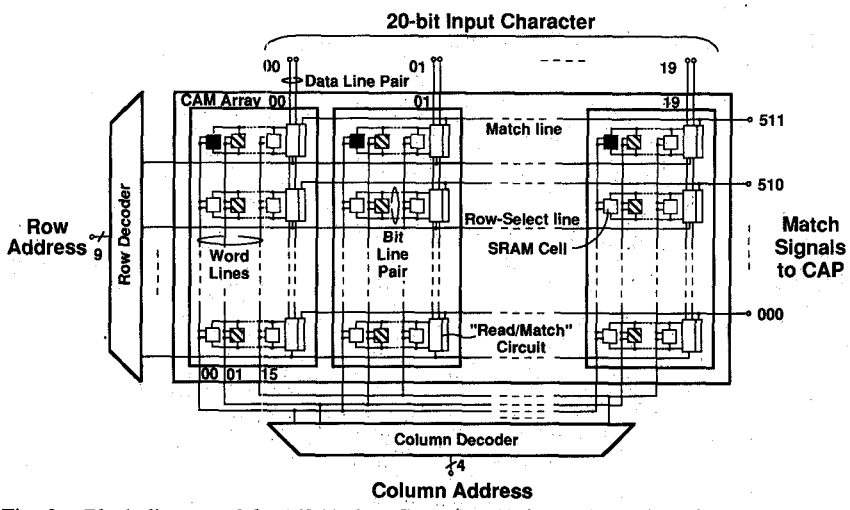


Fig. 3. Block diagram of the 160-kb data CAM (DCAM), which consists of 20 CAM arrays.

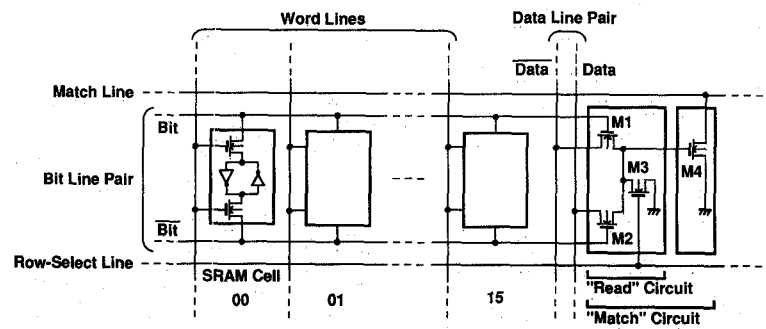


Fig. 4. Detailed CAM cells, a READ circuit, and a MATCH circuit for a single row.

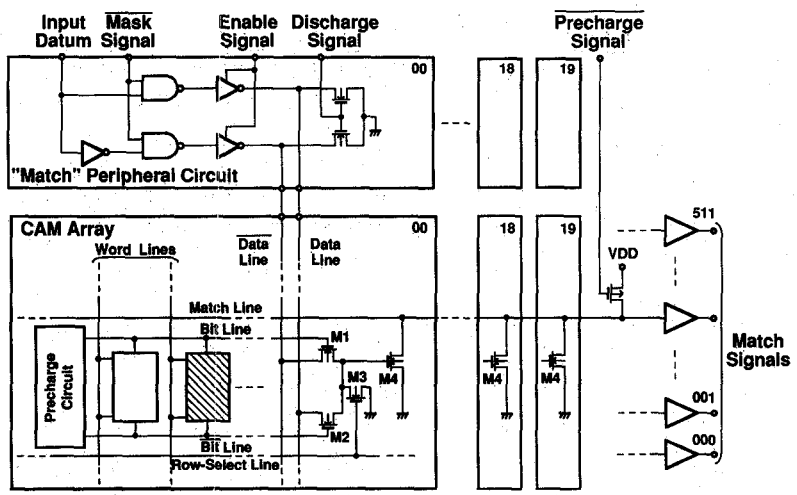


Fig. 5. Twenty CAM arrays and MATCH peripheral circuits. All data stored in a selected column are simultaneously compared with the input datum loaded from data lines, and the resulting match signals are obtained through match lines.

ever, in the MATCH operation explained above, the bit-(bit) line voltage must swing from the precharged level to the threshold voltage V_t of the nMOSFET in order to turn off $M1$ ($M2$). This large voltage swing takes place fast enough in the DCAM, because the number of cells connected to a bit-line pair, which is proportional to bit-line load capacitance, is far smaller than conventional SRAM's. Fig. 6 shows the simulation result for the MATCH operation (mismatched case). The bit-line voltage swing takes place within 2.9 ns after the word line rises, which results in the match-signal response time of 6.7 ns. If the number of cells per bit-line pair was not 16 but 64, which is a typical number for conventional SRAM's, the bit-line falling time would reach 10 ns (the fine line in Fig. 6), and then the response time of the match operation would exceed 10 ns.

Masking certain specified bits in every character location is an important function in the MATCH operation. For

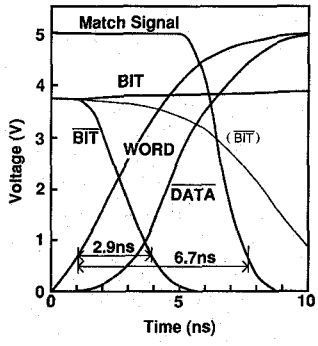


Fig. 6. Calculated match signal, bit and bit-line voltages, and data and data-line voltages. The match-signal response time is 6.7 ns.

example, the tag fields must be masked when the input 2-byte character codes are to be compared with the data fields alone at individual character locations. This function is performed simply with the MATCH peripheral circuit shown in Fig. 5. When a mask signal is ZERO, both data and data lines stay at ZERO. Thus $M4$ will be off regardless of the stored datum, which means the corresponding bit will be masked, because the input signal to the $M4$ gate is always ZERO.

In this way, a total of 512 match and mismatch signals is produced at the same time. This MATCH operation is continued for all the characters of the input word, starting from its first character, and the resulting match or mismatch signals are succeedingly sent to the CAP.

IV. Cellular Automaton Processor (CAP)

For dictionary search, the correct word search must be carried out even when input words contain character errors. This word search is called an "approximate word search." The number of character errors, called distance, between the input word and any stored word must be calculated, since stored words having the smallest distance value are determined to be the closest words to the input word. Distance is calculated as the required number of character substitutions, insertions, and deletions [3], [5]. For example, the distance of a wrong input word "San Fransisko" from a correct stored word "San Francisco" is 2 because two substitutions are required, "c" for "s" and "k."

A. Distance Calculation

A CAP calculates the distance between an input word and each stored word in a DCAM column using the match signals the DCAM produces. Hereafter, we define a match signal to be ZERO, and a mismatch signal to be ONE. Fig. 7 illustrates the concept of distance calculation in the CAP. As an example, the word being searched for is "MEMORY" and the input word is "MENORI." The CAP consists of an array of processor elements (PE's). Each PE performs the simple function, i.e., having and shifting a flag. A black PE has a flag, and a white PE does not have a flag. The solid line arrows in the PE array diagram show the flag shifts.

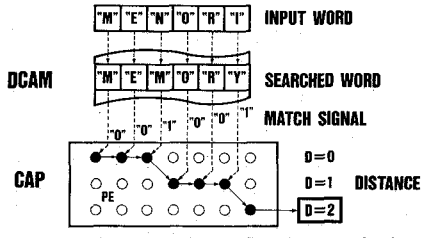


Fig. 7. Distance calculation in the CAP. The searched word is "MEMORY" and the input word is "MENORI." The CAP consists of a PE array, and executes distance calculation by the flag shifts between PE's. The distance between these two word is 2.

For each distance calculation, let the upper left corner PE have a flag as an initial condition. When the first character "M" is loaded, character comparison is performed in the DCAM as described in the previous section, and the match signal ZERO is sent to the upper left corner PE. In this case, the flag shifts to the right. When the next character "E" is loaded, the flag also shifts to the right, because the match signal ZERO is applied too. The third character "N" is wrong. Thus the mismatch signal ONE is sent to the PE having a flag. In this case, the flag shifts to the lower right. This shift means the distance value increases by 1 at this point. Similarly the flag shifts to the right two times for the input characters "O" and "R," and to the lower right for "I." Finally, the rightmost PE of the third row has the flag. This means that the calculated distance is 2, because the flag has shifted to the lower right PE two times. Similarly, when the rightmost PE of the first row or the second row has the flag, the calculated distance is defined to be 0 or 1, respectively. As can be seen from this explanation, the maximum distance calculated by the PE array is equal to the number of rows subtracted by one.

This is merely an explanation of the general concept of CAP distance calculation. Actual distance calculations performed in the CAP for character substitutions, insertions, and deletions are much more complicated, and will be discussed in detail elsewhere [10], [11].

B. Address Generation

The CAP in the present DISP consists of 128 blocks, each of which is a PE array of three rows by five columns, as shown in Fig. 8. Each block corresponds to each four-character group in a single column of the DCAM. That is, using the match signals supplied from the DCAM, a single block can calculate distance to a maximum value of 2 for a single word whose maximum character length is four. For a single word longer than four characters, several neighboring blocks can simultaneously be used to calculate distance. The calculated distance for each stored word is sent to the priority encoder. In the priority encoder, the stored words whose distance are equal to or less than 2 are treated as "approximately" matched words,

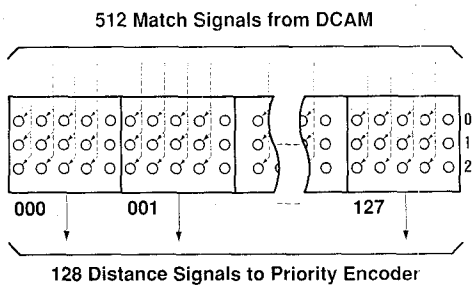


Fig. 8. The CAP architecture realized in the DISP ULSI. It consists of 128 blocks, each of which is a PE array of three rows by five columns. Each block can calculate a maximum distance of 2 for a single word whose maximum character length is four.

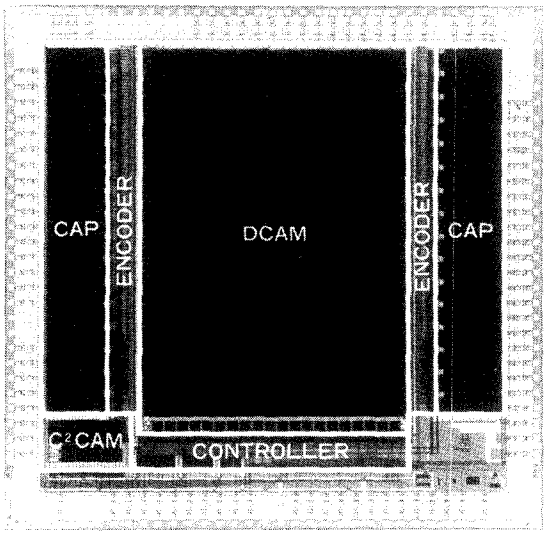


Fig. 9. Photograph of the 13.02-mm×12.51-mm DISP ULSI containing approximately 1.2-million transistors. The ULSI was fabricated using 0.8-μm CMOS technology.

and the stored words whose distance are larger than 2 are treated as mismatched words. The priority encoder serially produces the addresses of "approximately" matched words in the order of increasing distance. When there are several approximately matched words whose distance values are the same, their addresses are generated from the lower address value.

V. Characteristics

Fig. 9 shows a photograph of the DISP ULSI. Layouts for the DCAM, the CAP, and the C^2CAM were designed manually, while those for the priority encoder and the controller, and the interconnections among all the blocks, were designed using an automatic layout program. The DISP fabrication technology is summarized in Table II. It was fabricated with 0.8-μm, triple-layer-A1 interconnection, CMOS process technology. The third A1 level was used for the match lines and the row-select lines in the DCAM. This was useful for reducing CAM array area because the match lines and the row-select lines were allowed to run just above the bit-line pairs, which were drawn using the first A1 level.

TABLE II
0.8-μm CMOS Fabrication Process

FABRICATION TECHNOLOGY	0.8μmCMOS Single Polycide Triple-Layer-Al Twin Well
MOSFET STRUCTURE	LDD Strucutre
GATE LENGTH	n-ch : 0.8μm p-ch : 1.0μm
GATE OXIDE THICKNESS	150A°
LINE AND SPACE PITCH	1st -Al : 2.4μm 2nd-Al : 3.0μm 3rd-Al : 4.8μm

TABLE III
DISP ULSI Features

DCAM STRUCTURE	160Kb(=16 column×512row×20array)
CLOCK FREQUENCY	33 MHz for 20-bit character search
CHARACTER INPUT RATE	33 Million characters/sec
NUMBER OF SEARCH MODES	12 basic search modes : 3 approximate search modes ×4 anchor modes 3 "don't care" search modes: Fixed length "don't care" Variable length "don't care" Grammatical construction
POWER SUPPLY VOLTAGE	5V
POWER DISSIPATION	800mW
NUMBER OF MOSFETs	1,208,500
DIE SIZE	13.02×12.51mm²

Table III summarizes the characteristics of the DISP. A 13.02×12.51-mm^2 chip contains about 1208500 transistors, for which 1091000 are for the DCAM, 93000 are for both the CAP and the priority encoder, 17800 for the C^2CAM, and 6700 for the controller. The DISP ULSI operates at a clock frequency of 33 MHz with a typical power dissipation of 800 mW, at a 5-V power supply. The DISP ULSI can perform 12 basic search modes and three "don't care" search modes [11] at a character input rate of 33 million characters/second, so that all the words contained in 150 *Newsweek* magazines could be searched in one second.

VI. Experimental Results

Fig. 10 shows the measured operating waveforms for the DISP. Fig. 10(a) and (b) are two-phase 33-MHz clock pulses ($CK1$ and $CK2$). Fig. 10(c) is the character input to the DCAM, and Fig. 10(d) is the output signal of a matched word address from the priority encoder. The DCAM performs MATCH operations during the $CK1$ phase, while the CAP and the priority encoder operate during the $CK2$ phase. The response time of the address output is 25 ns.

VII. Discussion

For the classification of the stored words based on their first characters, a problem arises in the approximate word search operation when the first character happens to be mistaken. In this case, an approximate word search must be performed for all 16 categories. Therefore search time

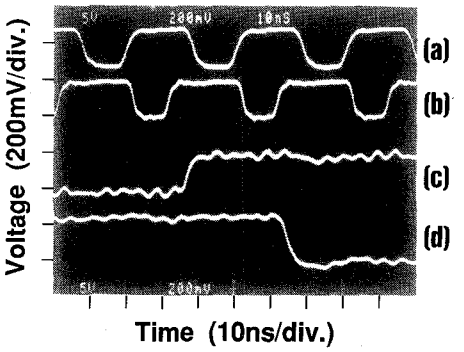


Fig. 10. Operation of the DISP ULSI at a clock frequency of 33 MHz and a supply voltage of 5 V. (a) and (b) are two-phase 33-MHz clock pulses, (c) is a character input to the DCAM, and (d) is an output match address from the priority encoder.

increases by a factor of 16. However, considering the fact that character errors are essentially only occasional, the effective overall degradation in search speed is insignificant.

In order to increase the vocabulary of the dictionary search system, several DISP chips can be arranged to operate in parallel. For example, 25 DISP's can make a 50000-word dictionary search system. By simultaneously applying a single input word to those DISP's in parallel, each DISP searches its stored words individually. Addresses produced from individual DISP's must be further encoded, so that an additional priority encoder is required.

VIII. Summary

A 1.2-million transistor dictionary search processor (DISP) has been described. In order to perform fast and highly functional dictionary search operations, a 160-kb data CAM (DCAM) and a cellular automaton processor (CAP) have been specially developed for the DISP. The DISP can store a maximum of 2048 words, and execute "exact" or "approximate" word searches at an input rate of 33M characters/second. This paper has focused not only on the overall operation but also on the architecture and circuits for the DCAM, which has a capacity 10 to 20 times greater than previously reported CAM's. We have shown that categorization of stored words is a key to successfully producing the large-capacity DCAM, along with the index search performed in the C^2CAM.

Acknowledgment

The authors are very grateful to all the people involved in developing the DISP ULSI and wish to express their thanks for all the generous help and cooperation they have received.

References

[1] H. Kaji, "Dictionary structure for flexible lexical transfer in machine translation," in *Proc. Int. Symp. Electron. Dictionaries*, Nov. 1988, pp. 36–38.

[2] H. Takahashi, N. Itoh, T. Amano, and A. Yamashita, "A spelling correction method and its application to an OCR system," *Pattern Recognition*, vol. 23, no. 3, pp. 363–377, 1990.

[3] P. A. Hall and G. R. Dowling, "Approximate string matching," *Computing Surveys*, vol. 12, no. 4, pp. 381–402, Dec. 1980.

[4] H. Kadota, J. Miyake, Y. Nishimichi, H. Kudoh, and K. Kagawa, "An 8-kbit content addressable and reentrant memory," *IEEE J. Solid-State Circuits*, vol. SC-20, no. 5, 951–957, Oct. 1985.

[5] H. Yamada, M. Hirata, H. Nagai, and K. Takahashi, "A high-speed string search engine," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 5, pp. 829–834, Oct. 1987.

[6] H. Yamada *et al.*, "Real-time string search engine LSI for 800-Mbit/sec LANs," in *Proc. IEEE Custom Integrated Circuit Conf.*, May 1988, pp. 21.6.1–21.6.4.

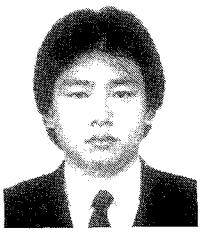
[7] J. P. Wade and C. G. Sodini, "A ternary content addressable search engine," *IEEE J. Solid-State Circuits*, vol. 24, no. 4, pp. 1003–1013, Aug. 1989.

[8] T. Ogura, J. Yamada, S. Yamada, and M. Tanno, "A 20-kbit associative memory LSI for artificial intelligence machines," *IEEE J. Solid-State Circuits*, vol. 24, no. 4, pp. 1014–1020, Aug. 1989.

[9] L. A. Hollaar, "Text retrieval computers," *IEEE Computer*, vol. 12, pp. 40–50, 1979.

[10] M. Motomura *et al.*, "A 1.2-M transistor, 33 MHz, 20-bit dictionary search processor ULSI for a machine translation system," in *ISSCC Dig. Tech. Papers*, Feb. 1990, pp. 90–91.

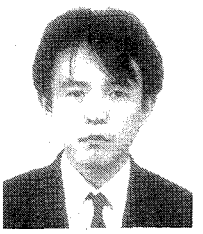
[11] M. Motomura *et al.*, "A 2-K word dictionary search processor with approximate word search capability," to be published.



Masato Motomura received the B.S. and M.S. degrees in physics from Kyoto University, Kyoto, Japan, in 1985 and 1987, respectively.

He joined the NEC Corporation, Kanagawa, Japan, in 1987, where he has been working in the System ULSI Research Laboratory of the Microelectronics Research Laboratories. His research interest includes associative memories, parallel processings, and neural networks.

Mr. Motomura is a member of the Institute of Electronics, Information and Communication Engineers of Japan.



Jun Toyoura was born in Tokyo, Japan, in 1962. He received the B.S. and M.S. degrees in physics from Tokyo University, Tokyo, Japan, in 1986 and 1988, respectively.

He joined the NEC Corporation, Kanagawa, Japan, in 1989, where he has been working in the System ULSI Research Laboratory of the Microelectronics Research Laboratories. Since then he has been involved in research of associative memories and neural networks.

Mr. Toyoura is a member of the Institute of Electronics, Information and Communication Engineers of Japan.

Kazumi Hirata was born in Yokohama, Japan, on March 7, 1957. He graduated from Turumi Technical School in 1975 and from the Nippon Electric Institute of Technology in 1980.

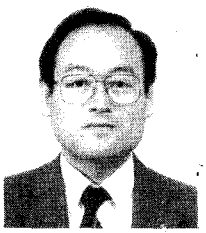
In 1975 he joined the NEC Corporation, Kawasaki, Japan, where he was engaged in the research and development of facsimile. Since 1978 he has been engaged in the development of high-voltage IC's.

Mr. Hirata is a member of the Institute of Electronics, Information and Communication Engineers of Japan.

Hideyuki Ooka was born in Hiroshima, Japan, in 1957. He received the B.E. degree in electronic engineering from Hiroshima Institute of Technology, Hiroshima, Japan, in 1980, and the M.E. degree in material engineering from Hiroshima University, Hiroshima, Japan, in 1983.

He joined the NEC Corporation, Kanagawa, Japan, in 1983, where he has been engaged in the development of MOS VLSI processing technology.

Mr. Ooka is a member of the Japan Society of Applied Physics.



Hachiro Yamada was born in Shizuoka, Japan, on June 6, 1949. He graduated from Shizuoka Technical High School in 1968, and from the Nippon Electric Institute of Technology in 1973.

In 1968 he joined NEC Corporation, Kawasaki, Japan, where he was engaged in the research and development of memory hierarchy systems and magnetic memory systems. Since 1985 he has been engaged in the development of VLSI architecture and special-purpose LSI. He is presently a Research Manager in the Microelectronics Research Laboratories, NEC Corporation.

Mr. Yamada is a member of the Institute of Electronics, Information and Communication Engineers of Japan.

Tadayoshi Enomoto (M'80) received the B.S.E.E. degree from Nihon University, Tokyo, Japan, in 1968, and the M.Sc. and Ph.D. degrees from the Department of Electrical Engineering, Ohio State University, Columbus, in 1972 and 1975, respectively.

He joined the NEC Corporation in 1968 and spent the first two and a half years as a Design Engineer of telephone exchanges. In 1970 he was awarded a four-year university fellowship by Ohio State University, where he studied solid-state devices and semiconductor physics for four and a half years. He returned to the NEC Corporation in 1975 and joined the Electron Device Research Laboratory of the Central Research Laboratories, where he worked on various MOS analog LSI's, including operational amplifiers, CCD delay lines, CCD filters, and switched-capacitor filters for telecommunication applications, as well as scaled MOSFET circuits for analog LSI's. While he was with the Ultra-LSI Research Laboratory of the Microelectronics Research Laboratories from 1982 to 1986, he worked on CMOS video signal processor (VSP) LSI's, stacked CMOS LSI's, and CMOS SOI LSI's. He was also involved in research to form recrystalized polysilicon-on-insulator (SOI) using CW Ar laser annealing. He is presently a Director of the System ULSI Research Laboratory of the Microelectronics Research Laboratories, Sagamihara-shi, Japan, and his current research interests concern development of various types of CMOS and BiCMOS LSI's including microprocessors, digital signal processors, DRAM's, SRAM's, CAM's, and SSE's, analog LSI's, and future ULSI's.

Dr. Enomoto is a member of the Institute of Electronics, Information and Communication Engineers of Japan.