A 20-kbit Associative Memory LSI for Artificial Intelligence Machines

TAKESHI OGURA, MEMBER, IEEE, JUNZO YAMADA, MEMBER, IEEE,
SHIN-ICHIRO YAMADA, MEMBER, IEEE, AND MASA-AKI TAN-NO

Abstract—**A 20-kbit (512 words×40 bits) CMOS associative-memory LSI is described. This LSI performs large-scale parallelism for highly efficient associative operations in artificial intelligence machines. Relational search, large-bit-length data treatment, and quick garbage collection are realized on the single-chip associative-memory LSI. A new cell array structure has been designed in order to reduce the chip area. A newly designed simple accelerator circuit allows for high-speed search operations. This LSI is fabricated using 1.2-μm and double-aluminum-layer CMOS process technology. A total of 284000 devices have been integrated on a 5.3×7.9-mm² chip. The measured minimum cycle time and power dissipation at 10-MHz operation are 85 ns and 250 mW, respectively. The associative memory described here, with its highly efficient associative operation capabilities, promises to be a large step toward the development of high-performance artificial intelligence machines.**

I. INTRODUCTION

RECENTLY, many studies have been performed on high-performance artificial intelligence machines based on associative memory applications. The representative studies are a Lisp machine [1], Prolog machines [2]–[4], MIT data-base accelerator [5], a text data-base accelerator [6], [7] and a knowledge-based super computer [8]. Data processing in these machines requires many associative operations. Furthermore, associative operations in sequence require an enormous number of comparisons and data I/O operations. These operations consume a huge amount of data processing time. Thus, a bottleneck exists here for high-speed data processing.

To deal with this problem, an associative-memory-based machine architecture is considered to be effective. An associative memory carries out a parallel data search based on the data content without extensive address handling. This memory is also referred to as a content addressable memory (CAM).

A fully integrated 1-kbit CAM LSI was first reported in 1983 [9]. CAM LSI's of 4 and 8 kbit were developed in 1985 [10], [11] and a 16-kbit CAM LSI was introduced in 1988 [12]. The 4-kbit CAM LSI has accomplished versatile functions to achieve a self-operative high-speed data search system. A high-performance Prolog machine using the 4-kbit CAM LSI was developed experimentally in 1986 [13].

There remain, however, serious problems in broadening CAM LSI applications for artificial intelligence machines. In actual data processing, various search functions, such as less-than and greater-than search, are necessary. Moreover, large-bit-length data cannot be treated at high throughput in conventional CAM LSI's. Furthermore, in order to broaden CAM LSI applications, it is necessary to develop a large-bit-capacity CAM LSI.

In order to resolve these problems, the present study is focused on the following two points. One is the CAM LSI architecture for carrying out various search functions and a large-bit-length data treatment. This large-bit-length data treatment is referred to here as wide-band data processing. The other point is the circuit design for a large-bit-capacity and high-speed CAM LSI.

As a result, a 20-kbit CAM LSI has been developed that integrates 284000 devices on a 5.3×7.9-mm² chip area using 1.2-μm and double-aluminum-layer CMOS process technology. The bit-serial relational search, such as less-than and greater-than search, and wide-band data processing capabilities have been incorporated into the chip. A quick garbage collection function is also realized in wide-band and non-wide-band data processing. A new CAM cell array structure is designed to reduce the cell array area and to speed up READ/WRITE operations. In addition, a simple accelerator circuit for discharging a match line is also designed to speed up search operations.

II. CAM ARCHITECTURE

A. Basic Architecture

The 20-kbit CAM LSI consists of five function blocks: cell array, blocks for word, bit, and address operations, and a control block. A block diagram of this LSI is shown in Fig. 1.

The cell array is composed of 512 words×40 bits of associative memory cells. Each constituent cell circuit is composed of a latch and EXCLUSIVE NOR circuit. A search

Manuscript received August 5, 1988; revised January 4, 1989.
The authors are with NTT LSI Laboratories, 3-1, Morinosato Wakamiya, Atsugi-shi, Kanagawa Pref. 243-01, Japan.
IEEE Log Number 8928783.

0018-9200/89/0800-1014$01.00

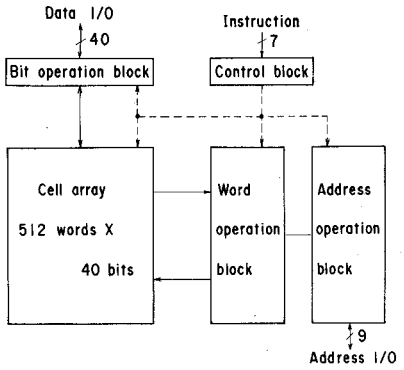


Fig. 1. Block diagram of the 20-kbit CAM LSI.

result flag signal for every word is generated by using the output signal of every EXCLUSIVE NOR circuit. In addition, all response flag signals are sent to the word operation block in a parallel fashion.

One of the outstanding architectural features lies in the cell array. In a data WRITE operation, these 40 bits in each word location are divided into a 32-bit data field and an 8-bit tag field. The 32-bit data field is divided into four parts of 8-bit data. The tag field is used for the relational search and wide-band data processing. Writing into arbitrary data parts and arbitrary tag bits is executed. This operation is referred to here as a partial-WRITE operation. The data written parts and bits are pointed by 12-bit write-mask data corresponding to the four data parts and eight tag bits. The write-mask and 40-bit search-mask data are latched in the bit operation block. The search-mask data specify search bit positions in 40-bit-length data.

The other outstanding architectural features lie in the word and address operation blocks. The word operation block has been designed to carry out the essential operations of the relational search, wide-band data processing, and garbage collection. In this block, two registers are provided for each word. One of the two registers is a master–slave type and receives a search result from the cell array. The other register is an *R-S* flip-flop type and is used for the garbage collection. These two registers for each word are referred to here as response and garbage flag registers. A high-speed multiple-response resolver has also been incorporated in the word operation block. The resolver selects a ONE signal from among the response or garbage flag registers in the same way as that already reported in [10].

The address operation block handles the address data when it is necessary to link one part of structural data to the following part stored in off-chip memories. Address handling is also performed while quick garbage collection is being executed in the wide-band data processing. Furthermore, the address data corresponding to matched and data-written words are retrieved by this address operation block.

This CAM LSI supports a powerful but simple instruction set for various CAM applications. The instruction set contains 26 instructions classified into four modes: search, READ, WRITE, and garbage collect, as listed in Table I. The

TABLE I
INSTRUCTION SET OF THE 20-KBIT CAM LSI

Mode	Operations
Search (5 instructions)	•fully parallel equal search •bit-serial relational search •wide band data search
Read (4 instructions)	•data & address read of matched words (to be garbage or not) •data read using address as same as RAM
Write (8 instructions)	•data write into matched words (in parallel or in sequence) •data write into unmatched words in parallel •data write into garbage words in sequence •data write using address as same as RAM •search-/write-mask-data write
Garbage collect (6 instructions)	•initialization (all word becomes garbage) •garbage collection for matched/unmatched words •garbage collection for contiguous 2/4/8 words
Nop & Test (3 instructions)	

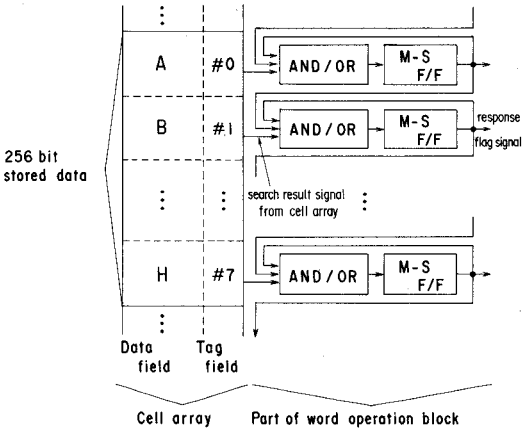


Fig. 2. Circuits for relational search and wide-band data processing.

relational search, wide-band data processing, and quick garbage collection can be carried out by a combination of these instructions. Various system configurations using the CAM LSI are available. A typical CAM system configuration, tuned to the relational search, is described in the Appendix.

B. Relational Search

The relational search operation is carried out by an iterative bit-serial operation for every word in parallel. The word operation block circuit for the relational search is composed of AND/OR logic gates and a response flag register in each word location, as shown in Fig. 2. These circuits are used to accumulate results of the iterative bit-serial operations.

The schematic diagram and operation flow of the less-than search, which is one of the typical relational search operations, are shown in Fig. 3(a) and (b). The search operation is iteratively executed from the most significant bit to the least significant bit in bit-serial and word-parallel manner. The iterative search operation is carried out by using a set of temporary key data. The set of temporary key data is generated from original key datum by an off-chip register and arithmetic logical unit (RALU) or a

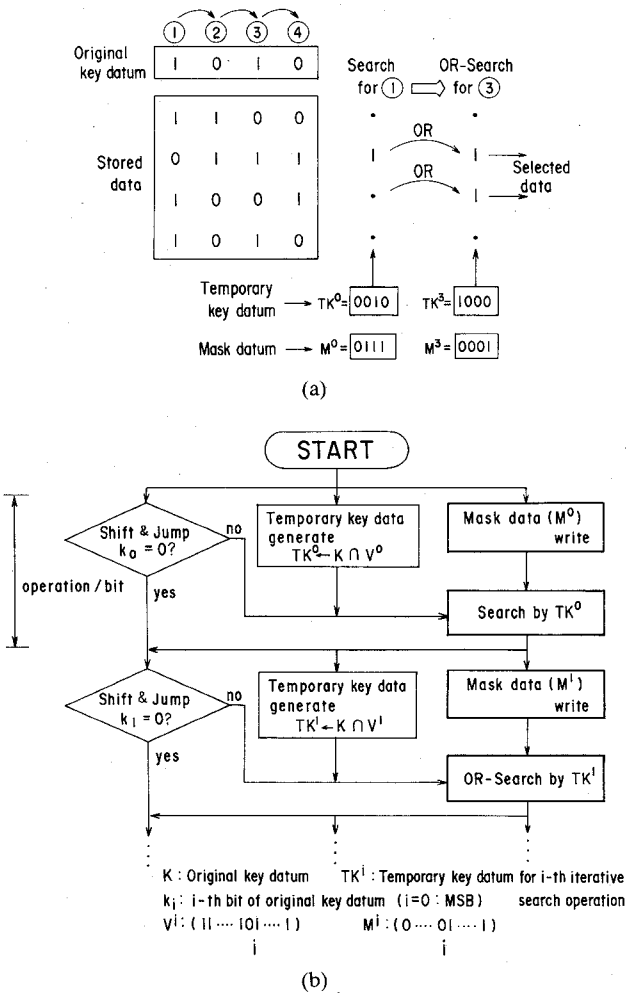


Fig. 3. Bit-serial less-than search: (a) schematic diagram and (b) operation flow.

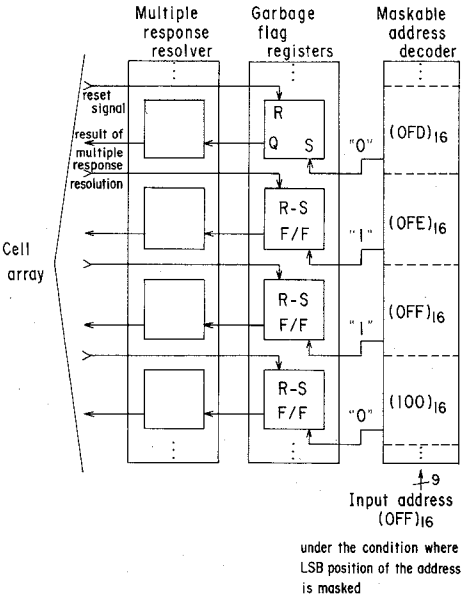


Fig. 4. Circuits for quick garbage collection.

processor. This is done simply on the basis of logic operations. In the less-than search operation, the temporary key datum TK^i for ith iterative search operation is generated by $TK^i = K \cap V^i$, as shown in Fig. 3. Here, K is the original key datum, and V^i is $(1 \cdots 101 \cdots 1)$.

At the search operation for the most significant bit, stored datum (0111) is selected, as shown in Fig. 3(a). This result indicates that the stored datum (0111) is less than the original key datum (1010). In this way, the less-than search operation is executed in a serial manner until all possible temporary key data have been compared. The average number of possible temporary key data is 16 for 32-bit-length data. In Fig. 3(a), the final result indicates that the stored data (0111) and (1001) are less than the original key datum (1010).

The CAM LSI "mask data write" operation can be executed in parallel to the other operations—"shift and jump" and "temporary key data generate"—under appropriate system configurations (see the Appendix). When the ith bit of original key datum ki is ONE, the ith bit-serial operation is completed in one cycle. If ki is ZERO, the ith bit-serial operation requires two cycles. As a result, a 32-bit less-than search can be executed in an average of 48 cycles, and 64 cycles are necessary in the worst case. The less-than search execution speed is estimated to be $O(m)$, where m is bit length, because the operation is executed in bit-serial and word-parallel manner. It should be noted that relational search execution speed is not affected by the data volume.

C. Wide-Band Data Processing

This CAM LSI has been designed to support associative operation on excessive bit-length data of more than 40 bits. The 8-bit tag field is used for the wide-band data processing. Data having bit length exceeding 40 bits can be stored in two or more successive word locations in a folded form by using the tag. The tag indicates the following part of the data. In Fig. 2, up to 256 bits of data are handled in eight successive word locations.

Wide-band data processing is accomplished by iteration of 32-bit data processing and communication of the processing results between neighboring word locations. This communication is executed at all word locations in parallel and performed without any overhead time. The 32-bit data processing, such as equal and relational search, is carried out by referencing the tag field. Consequently, $32 \times n$ bits of large-bit-length data can be processed in $k \times n$ cycles for a certain integer k. The integer k depends on the content of the 32-bit data processing. In the wide-band data processing, the number of cycles necessary for execution is not affected by the data volume.

For instance, for 256-bit data where $n = 8$, the equal search operation is completed in $1 \times 8 = 8$ cycles, where $k = 1$, and the less-than search operation is completed in an average of $48 \times 8 = 384$ cycles, where $k = 48$. This high throughput and the flexible expandability of the word configuration is an efficient way to process complicated structural data.

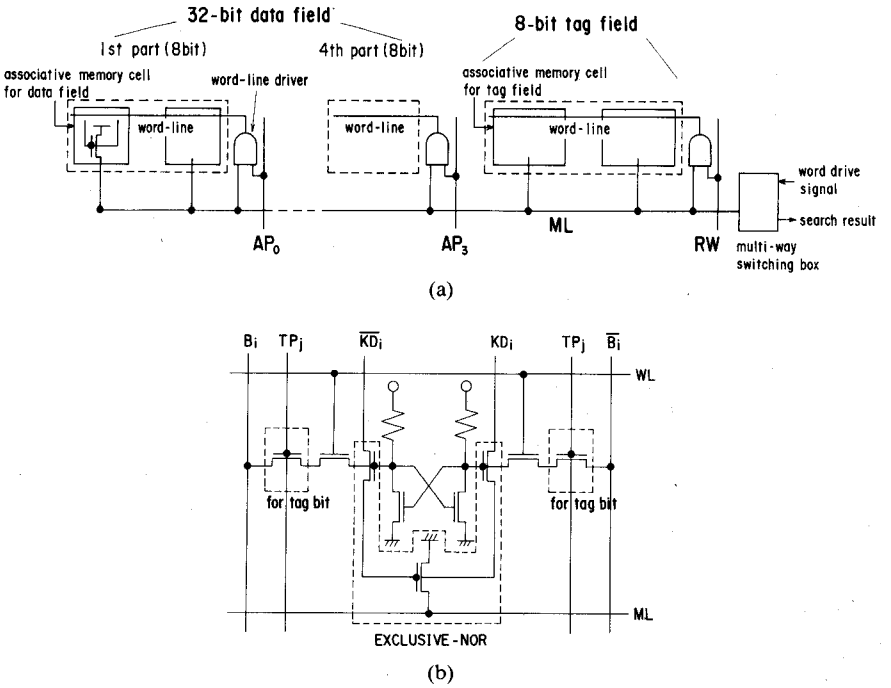


Fig. 5. New CAM cell array: (a) schematic diagram and (b) cell circuit. *B*: bit line, *KD*: key-data line, *ML*: match line, *TP*: partial-WRITE signal line for tag bit.

D. Quick Garbage Collection

Quick garbage collection in the wide-band data processing can be performed by using a garbage flag register, a multiple-response resolver, and a maskable address decoder, as shown in Fig. 4. The garbage flag register is used to indicate whether each word location is reserved or empty. Before a WRITE operation starts, the multiple-response resolver selects one possible word location from among empty word locations in the same way as that of the 4-kbit CAM LSI [10].

In the wide-band data processing, quick garbage collection is carried out by using the address data in a word serial manner. A word address for a datum with an excessive bit length, which occupies two or more word locations, is generated in response to search operations. The generated address data are sent to a maskable address decoder to identify a set of two or more contiguous word locations. Then, garbage flag registers located in successive words, whose contents should be garbage, are set to be ONE at the same time by the maskable address decoder output, as shown in Fig. 4. In Fig. 4, $(OFF)_{16}$ is sent to the maskable address decoder, which masks the LSB position of the address. Therefore, two contiguous word locations $(OFE)_{16}$ and $(OFF)_{16}$ are identified by the decoder.

The maskable address decoder can be realized by setting A_i and $\bar{A}_i$ at a high level, where A_i is a masked address bit. In this CAM LSI, the lower three bits of input address are maskable. Therefore, two, four, and eight contiguous word locations can be made garbage at the same time. In addition, the maskable address decoder is used as a conventional word address decoder under nonmaskable conditions.

III. LSI DESIGN

A. CAM Cell Array

In order to develop a large-bit-capacity CAM LSI and to realize a partial-WRITE operation, a new CAM cell array structure has been designed for the 20-kbit CAM LSI, as shown in Fig. 5. This CAM cell array consists of associative-memory cells, word-line drivers, and a multi-way switching box.

The associative-memory cell circuit is composed of seven/nine n-MOS transistors and two high-resistive poly-Si load devices, as shown in Fig. 5(b). In contrast to a conventional transistor-load cell circuit, this high-resistive poly-Si load cell circuit allows the parallel-WRITE operation for multiple word locations at the same time. The capability of a parallel-WRITE operation is essential for a CAM LSI, as pointed out in [10]. A search operation is achieved by detecting whether a precharged match line has discharged through the cell or not.

This cell circuit has two merits. One merit is the reduction of the number of transistors that constitute an EXCLUSIVE NOR circuit. In this cell circuit, the EXCLUSIVE NOR circuit is made up of three transistors. The conventional EXCLUSIVE NOR circuit is composed of four transistors [11]. The other merit is small stray capacitance of the match line because only one transistor is connected to the match line in the cell circuit. On the other hand, a drawback is slow discharging of the match line through the cell circuit. The gate voltage of the cell transistor discharging the match line is about $V_{cc}-2V'_{th}$, where V'_{th} is threshold voltage including the backgate bias effect, and V_{cc} is the supply voltage. In order to overcome this demerit, a simple

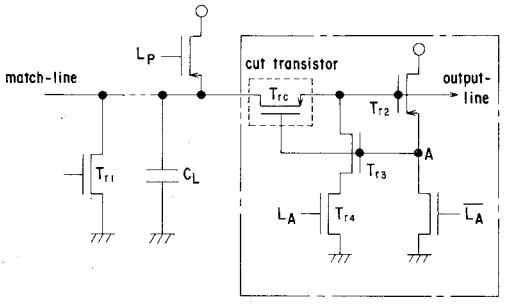


Fig. 6. Simple accelerator circuit.

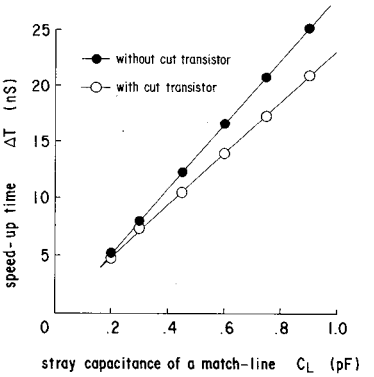


Fig. 7. Speed-up time ΔT dependence on stray capacitance of a match line C_L.

accelerator circuit for discharging the match line is applied (see Section III-B).

In this CAM cell array structure, the match line is also used to identify data written/read word locations. The partial-WRITE operation into arbitrary data parts is carried out by using the word-line drivers and the match line. Any additional horizontal signal line is not necessary, as shown in Fig. 5. Therefore, this CAM cell array structure is suitable for a large-bit-capacity CAM LSI.

In a search operation, the word lines are isolated electrically from the match line and set at a low level by setting signals AP_i ($i = 0$–3) and RW at a low level. Therefore, a search result propagates to the word operation block through the multi-way switching box. Here, signals AP_i and RW control whether the word line is isolated electrically from the match line or not.

In a data READ/WRITE operation, all-key-data line KD_i, $\overline{KD_i}$ is set at a low level to cut the current path from a match line to the ground. A word drive signal is sent to the match line through the multi-way switching box. In a READ operation, all signals AP_i and RW are set at a high level to connect the match line and word lines. Signals TP_j ($j = 0$–7) are also set at a high level to activate the tag bit cells. In a WRITE operation, RW is set at a high level, and each AP_i and TP_j are set at a low/high level corresponding to write-mask data. Writing into data parts and tag bits, where AP_i and TP_j are set at a low level, is prohibited.

This new cell array structure has two merits: one is cell array area reduction and the other is word-line delay reduction. The cell array area is reduced by 25 percent as compared with the structure in which a horizontal signal line is added to identify word locations, which is identical to a double-word-line structure in a static RAM [14]. The word-line delay reduction is achieved because each word line, which has small line capacity, is driven in parallel.

B. Accelerator Circuit for Discharging Match Line

A simple accelerator circuit for discharging the match line is designed as shown in Fig. 6. A search result is detected whether a precharged match line is discharged or not. Therefore, quick discharging of the match line is necessary to achieve a high-speed search operation.

The accelerator circuit is composed of four transistors. The match and output lines are precharged by setting L_P and L_A at a low level. A search operation starts in $L_P =$ $L_A =$ ONE. In the case of mismatch, the charge of the match line begins to flow through (a) cell transistor(s). Then, the match-line potential V_M falls down exponentially. When the V_M reaches the threshold voltage of transistor T_{r2}, the transistor goes to the ON condition. The node A potential V_A becomes higher by charging up through T_{r2}, then the T_{r3} goes to the ON condition. As a result, an additional current path through T_{r3} and T_{r4} is generated, and the discharging of the match line is accelerated. If the cut transistor T_{rc} is inserted, as shown in Fig. 6, the match line and the output line are cut electrically on the way. The stray capacitance of the output line is smaller than that of the match line. Therefore, the output line potential falls down more quickly. In order to estimate the speed-up time, the dependence of the speed-up time ΔT on stray capacitance C_L of a match line is calculated using a MOS circuit simulator.

The calculated results are shown in Fig. 7. Since the C_L of the match line is estimated at 0.3 pF, irrespectively of the cut transistor, ΔT is about 8 ns, which is 30 percent of total search operation time. The accelerator circuit without a cut transistor was adopted for this CMA LSI. When the C_L becomes larger, the cut transistor is more effective. This simple accelerator circuit contributes to realization of a high-speed search operation.

IV. FABRICATED LSI

The 20-kbit CAM LSI was fabricated using a 1.2-μm CMOS process technology with double-aluminum layers for interconnection. A photomicrograph of the LSI is shown in Fig. 8. A total of 284 000 devices have been integrated on a 5.3×7.9-mm^2 chip. The newly designed cell for data bit occupies 578 μm^2.

An example of the I/O signal waveforms is shown in Fig. 9. Fig. 9 shows the timing relationships between clock, data, address, and search-response signals. The search-response signal indicates whether a matched word exists. Waveforms are observed in a sequence of search-mask-data WRITE, search, and READ operations. The waveforms indicate that the search-response signal changes to ONE in a search-operation cycle, and the searched-out data and ad-

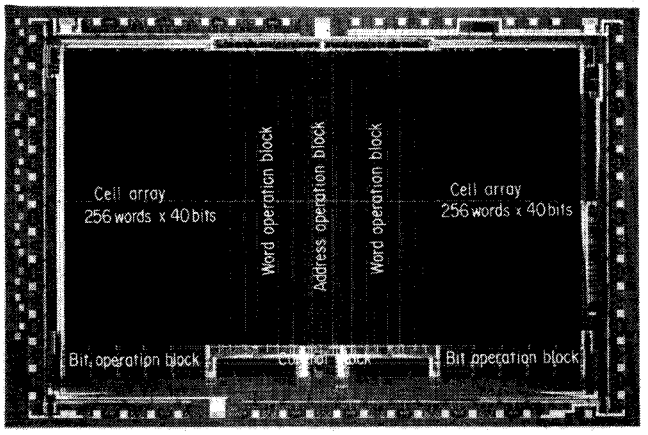


Fig. 8. Photomicrograph of a 20-kbit CAM LSI.

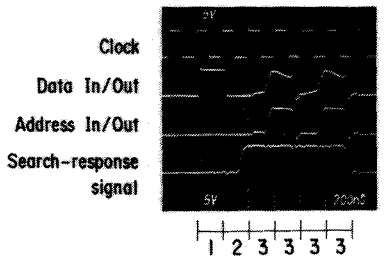


Fig. 9. I/O signal waveforms: 1: Search-mask-data WRITE; 2: "search"; 3: READ.

TABLE II
FEATURES OF THE 20-KBIT CAM LSI

Configuration	512 words × 40 bits
Instruction set	26 instructions
Cycle time	85 ns (minimum)
Supply voltage	5 V
Power dissipation	250 mW at 10-MHz operation
I/O interface	I/O common, TTL compatible
Number of pins	66
Package	84-pin PGA
LSI process technology	1.2-μm CMOS with double aluminum layers
Number of devices	284,000
Chip size	5.3mm × 7.9mm

dress are then retrieved sequentially during every cycle. In the fourth READ-operation cycle, the search-response signal falls down to ZERO because searched-out data and address are retrieved completely. A minimum cycle of 85 ns has been measured. Power consumption at 10-MHz operation is 250 mW. The features of the 20-kbit CAM LSI are summarized in Table II.

V. CONCLUSION

A 20-kbit CMOS CAM LSI has been developed. The functions of relational search, wide-band data processing, and quick garbage collection have achieved a highly parallel associative system. The functional features are summarized as follows:

1) relational search by iterative bit-serial operation,
2) wide-band data processing,
3) quick garbage collection in the wide-band data processing,
4) fully parallel equal search,
5) word-count extension capability,
6) multiple-response resolution,
7) simultaneous parallel writing on multiple-word locations,
8) partial writing on a specified field, and
9) address output corresponding to matched or data-written words.

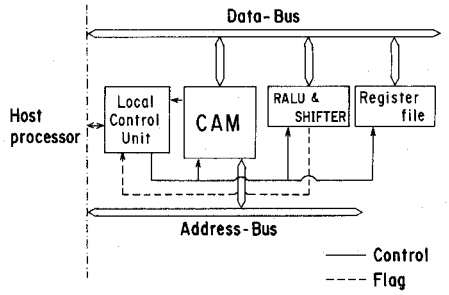


Fig. 10. Typical CAM system configuration.

A new CAM cell array structure was designed in order to reduce the chip area. A total of 284 000 devices have been integrated on a 5.3×7.9-mm^2 chip using 1.2-μm and double-aluminum-layer CMOS process technology. A newly designed simple accelerator circuit allows for high-speed search operations. The measured minimum cycle time and power dissipation were 85 ns and 250 mW (at 10 MHz), respectively. This associative memory, with its proven highly efficient associative operation capabilities, is expected to contribute significantly to the development of high-performance artificial intelligence machines.

APPENDIX
TYPICAL SYSTEM CONFIGURATION

Various system configurations using the CAM LSI's are available by the conventional TTL technology. In [13], a CAM system using the 4-kbit CAM LSI's tuned Prolog execution was reported. A typical CAM system configuration, tuned to the relational search operation, is shown in Fig. 10. This system consists of a CAM chip array, RALU&SHIFTER, a register file, a local control unit, a data bus, and an address bus.

In this CAM system, CAM chip arrays and other functional blocks can operate in parallel. The host processor sends key data and macroinstructions, such as less-than and greater-than search. The local control unit decodes the macroinstruction into a microinstruction sequence. The register file is used to store search-mask data for bit-serial operations. The RALU&SHIFTER executes key-data han-

dling, as shown in Fig. 3(b), and generates several flags for sequence control.

ACKNOWLEDGMENT

The authors wish to thank Dr. T. Sudo, Dr. N. Ieda, Dr. T. Nakashima, and Dr. T. Kimura for supporting this work. The authors also wish to express their indebtedness to J. Naganuma for the design of this LSI.

REFERENCES

[1] J. G. Bonar and S. P. Levitan, "Real-time LISP using content addressable memory," in *Proc. 10th Int. Conf. Parallel Processing*, Aug. 1981, pp. 112–117.

[2] J. Naganuma, T. Ogura, S. Yamada, and T. Kimura, "A high-speed CAM based architecture for a prolog machine (ASCA)," *IEEE Trans. Comput.*, vol. 37, pp. 1375–1383, Nov. 1988.

[3] W. Dilger and H. A. Schneider, "ASSIP-T: A theorem proving machine," in *Proc. Int. Conf. FGCS'84*, Nov. 1984, pp. 497–506.

[4] M. Ohkubo, H. Yasuura, N. Takagi, and S. Yajima, "A hardware oriented unification algorithm using a content addressable memory" (in Japanese), *J. IPSJ*, pp. 915–922, Sept. 1987.

[5] J. P. Wade, P.J. Osler, R. E. Zippel, and C. G. Sodini, "The MIT database accelerator: 2K-trit circuit design," in *Proc. Symp. VLSI Circuit*, May 1987, pp. 39–40.

[6] K. Takahashi, H. Yamada, H. Nagai, and M. Hirata, "Intelligent string search processor to accelerate text retrieval," in *Proc. IWDM'87*, Oct. 1987, pp. 440–453.

[7] H. Yamada, M. Hirata, H. Nagai, and K. Takahashi, "A high-speed string-search engine," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 829–834, Oct. 1987.

[8] T. Higuchi, T. Furuya, H. Kusumoto, K. Handa, and A. Kokubu, "The IX supercomputer for knowledge based systems," in *Proc. Fall Joint Comp. Conf.*, Nov. 1986, pp. 1041–1049.

[9] T. Nikaido, T. Ogura, S. Hamaguchi, and S. Muramoto, "A 1Kbit associative memory LSI," *Japan. J. Appl. Phys*, vol. 22, pp. 51–54, Jan. 1983.

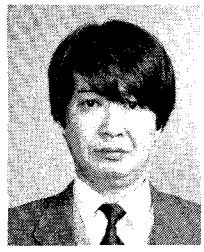
[10] T. Ogura, S. Yamada, and T. Nikaido, "A 4-kbit associative memory LSI," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 1277–1282, Dec. 1985.

[11] 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, pp. 951–963, Oct. 1985.

[12] H. Yamada *et al.*, "Real-time string search engine LSI for 800-Mbit/sec LANs," in *Proc. CICC'88*, May 1988, pp. 21-6.

[13] J. Naganuma, T. Ogura, and S. Yamada, "A high-speed CAM based prolog-machine (ASCA) and its farmware interpreter" (in Japanese), in *Dig. Pap. 32nd IPSJ Tech. Meeting*, Mar. 1986, pp. 55–56.

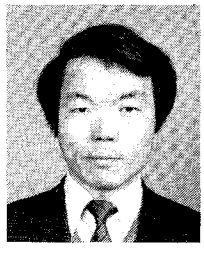
[14] T. Sakurai and T. Iizuka, "Double word line and bit line structure for VLSI RAM's," in *Proc. 15th Conf. Solid State Dev. Mater.*, Sept. 1983, pp. 269–272.



Takeshi Ogura (M'86) was born in Kobe, Japan, on October 1, 1953. He received the B.S. and M.S. degrees in electronic engineering from Osaka University, Osaka, Japan, in 1976 and 1978, respectively.

In 1978 he joined the Musashino Electrical Communication Laboratory, Nippon Telegraph and Telephone Public Corporation (NTT), Tokyo, Japan. He is now with NTT LSI Laboratories, Kanagawa, Japan. He is interested in logic-in-memory LSI's and their application.

Mr. Ogura is a member of the Institute of Electronics, Information and Communication Engineers of Japan, the Information Processing Society of Japan, and the IEEE Computer Society.



Junzo Yamada (M'86) was born in Nagoya, Japan, on April 3, 1951. He received the B.S. and M.S. degrees in electronic engineering from Tokyo Institute of Technology, Tokyo, Japan, in 1974 and 1976, respectively.

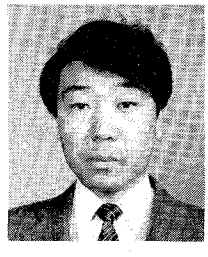
In 1976 he joined the Musashino Electrical Communication Laboratory, Nippon Telegraph and Telephone Public Corporation (NTT), Tokyo, Japan, where he worked on the design and testing of MOS memory. He is currently a Senior Research Engineer, NTT LSI Laboratories, Kanagawa, Japan. He has been engaged in research on the circuit design of dynamic MOS memory.

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

Shin-ichiro Yamada (M'86) was born in Hokkaido, Japan, on February 20, 1948. He received the B.S. degree in electrical engineering from Muroran Institute of Technology, Muroran, Japan, in 1970.

In 1970 he joined the Musashino Electrical Communication Laboratory, Nippon Telegraph and Telephone Public Corporation (NTT), Tokyo, Japan, where he was engaged in the research and development of memory LSI's and LSI-CAD systems. He is currently a Senior Research Engineer, NTT LSI Laboratories, Kanagawa, Japan.

Mr. Yamada is a member of the Institute of Electronics, Information and Communication Engineers of Japan, the Information Processing Society of Japan, and the IEEE Circuits and System Society.



Masa-aki Tan-no was born in Tokyo, Japan, on February 5, 1958. He received the B.S. degree in electrical engineering from Nippon University, Tokyo, Japan, in 1980.

In 1980 he joined the Musashino Electrical Communication Laboratory, Nippon Telegraph and Telephone Public Corporation (NTT), Tokyo, Japan. He is currently a Research Engineer at NTT LSI Laboratories, Kanagawa, Japan. He has been engaged in the testing of VLSI circuits.

Mr. Tan-no is a member of the Institute of Electronics, Information and Communication Engineers of Japan.