Associative Memories and Processors: An Overview and Selected Bibliography

BEHROOZ PARHAMI

Abstract—**A survey of associative processing techniques is presented, together with a guide to the published literature in this field. Some familiarity with the basic concepts of associative processing is assumed. The references have been divided into four groups dealing with architectural concepts, hardware implementation, software considerations, and application areas. The discussion of architectural concepts consists of a classification of associative devices into four major categories (fully parallel, bit-serial, word-serial, and block-oriented) and an enumeration of techniques for dealing with multiple responses and hardware faults. With respect to hardware implementation, considerations are given to the basic operations implemented, hardware elements used (e.g., cryoelectrics, magnetic elements, and semiconductors), and physical characteristics such as speed, size, and cost. The discussion of software aspects of associative devices deals with synthesis of algorithms, programming problems, and software simulation. The application areas discussed include solution of some mathematical systems, radar signal processing, information storage and retrieval, and performance of certain control functions in computer systems.**

Introduction

MORE THAN 15 YEARS have passed since the publication of the first paper dealing with the concept of associative processing [141]. Numerous implementation problems in the early years limited the applications of associative processing techniques to small and highly specialized systems. Recent advances in computer technology and development of new architectural concepts for associative devices have made the design of larger and more flexible systems possible.

The primary thesis of this paper is that associative processing is an important concept that can be employed to enhance the performance of special-purpose and general-purpose computers of the future. This claim is substantiated by the numerous applications suggested for associative processing techniques and the projected improvement in performance over conventional techniques. For example, it has been noted that the solution of certain data processing problems on a general-purpose computer using conventional procedures requires excessive running times. These problems include military data processing [7], applications involving large rapidly changing data bases [128], and information retrieval [132]. Associative processors can handle these problems much more efficiently than conventional systems.

The purpose of this paper is to provide a survey of associative processing techniques and a guide to the published literature in this field. This is not meant to be a tutorial paper and assumes some familiarity with the basic concepts of associative processing. These concepts have been covered in a number of surveys and introductory expositions which have appeared in the literature [17], [18], [65], [101], [153], [155], [164]. Murtha [105] surveys highly parallel information processing systems and provides insight into the relationship of associative devices to other highly parallel systems. The division of references into four groups dealing with architectural concepts, hardware implementation, software considerations, and application areas should help the reader put individual references in perspective.

Before proceeding any further, it is appropriate to define the class of systems we are dealing with. Most authors have defined associative processing as the performance of certain operations in parallel on a number of words. In this paper, we take the view that an associative device is a "black box" with certain functional capabilities. Whether or not operations are performed in parallel within the black box is immaterial. In fact, we will see that the concepts of serial and block-oriented associative processing are very important. Accordingly, we propose the following definitions.

Associative Memory: An associative memory is a storage device that stores data in a number of cells. The cells can be accessed or loaded on the basis of their contents.

Associative Processor: An associative processor is an associative memory in which more sophisticated data transformations can be performed on the contents of a number of cells selected according to the contents.

Associative Computer: An associative computer is a computer system that uses an associative memory or processor as an essential component for storage or processing, respectively.

As can be seen from these definitions, the distinction between an associative memory and an associative processor is not very clear. In fact, the two terms have been used interchangeably by some authors. Most associative memories can be programmed to perform logic and arithmetic operations and can therefore simulate associative processors. We accept the convention that an associative processor must be able to perform logic and/or arithmetic operations in hardware. Note that the definition given for associative processors excludes array computers, such as the Illiac IV,[1] from consideration. Also, the definition of associative computers does not embrace computers that use associative devices for performing executive control functions.

The terminology used for associative devices is by no means standardized. Associative memories [133] have been identified by a variety of names. The property that allows them to address the data by content has resulted in the names "content-addressable memory" [39], "content-addressed memory" [87], and "data-addressed memory" [108]. Their search capability has prompted the names "catalog memory"

Manuscript received July 21, 1972; revised January 19, 1973. This work was supported by the U. S. Office of Naval Research, Mathematical and Information Sciences Division, under Contracts N00014-69-A-0200-4027, NR 048-129; the U. S. Atomic Energy Commission, under Contract AT(11-1) Gen 10, Project 14; and the National Science Foundation, under Grant GJ 33007X.

The author is with the Computer Science Department, University of California, Los Angeles, Calif. 90024.

[1] G. H. Barnes *et al.*, "The ILLIAC IV computer," *IEEE Trans. Comput.*, vol. C-17, pp. 746–757, Aug. 1968. Also W. J. Bouknight *et al.*, "The Illiac IV system," *Proc. IEEE*, vol. 60, pp. 369–388, Apr. 1972.

[141], "multiple instantaneous response file" [111], "parallel search file" [49], "parallel search memory" [42], and "search memory" [74]. Finally, one of their main characteristics—namely, decentralized processing—has resulted in the name "distributed-logic memory" [83]. The term "associative processor" [41] has been used universally. Associative computers [124] have been called by at least two other names: "distributed-logic computer" [82] and "logic-in-memory computer" [151]. We will use the terms "associative memory," "associative processor," and "associative computer" because they have been widely used elsewhere and for consistency and brevity throughout this paper. The term "associative device" refers to an associative memory or processor.

Architectural Concepts

In this section, we will review some concepts in the design of associative devices which are independent of implementation techniques. Several computer systems with associative processing capabilities have been described [32], [137]. Associative devices can be used in a number of ways in a computer system, depending on the application: 1) as a special-purpose peripheral device [34], [76]; 2) as part of a storage hierarchy [33], [110], [151]; 3) as a subsystem integrated into the computer system [34], [56]; and 4) as an autonomous system within a multiprocessor [34], [137], [162].

In addition, an associative memory can be used for performing some executive control functions within a computer system. In what follows, we will consider possible organizations for associative devices without considering them in the context of their application.

From the point of view of organization, we can classify associative devices in four categories: fully parallel, bit-serial, word-serial, and block-oriented. Several extensions and modifications to the basic organizations discussed here have been proposed, e.g., multiaccess capability [107], hybrid associative memory [159], and read-only associative memory [88], [115]. In what follows, we will discuss these four basic organizations along with techniques for dealing with multiple responses and hardware faults in associative devices.

Fully Parallel Systems

Fully parallel associative memories and processors can be further divided into word-organized and distributed-logic systems. Since comparison logic must be associated with each bit of the memory in a fully parallel word-organized associative device, use of cryogenic techniques becomes attractive. This accounts for the large number of references dealing with cryogenic associative memories. Until very recently, only exact match operations had been proposed to be implemented in fully parallel associative devices. Recent developments in integrated-circuit technology have made possible the design of associative processors in which a variety of comparison and arithmetic operations are performed in parallel on each word. Since the amount of processing logic associated with each word is relatively high in such a system, very few applications can efficiently utilize its power.

The concept of a distributed-logic memory was proposed by Lee [82], [83], [84]. This organization is very desirable when dealing with variable-length data items. The distributed-logic memory consists of a control unit and a large number of identical cells, each of which stores one character of information (consisting of a symbol part and a state part) and can communicate with its two neighbors and with the control unit. The cells evaluate the input condition (given by the input and command buses) independently and in parallel, and depending on the outcome will perform or ignore the specified operation. Typical operations are changing state, transmitting state information to a neighbor, accepting data from the input bus, or putting data on the output bus. Using these commands, the distributed-logic memory can perform pattern matching operations, and is therefore well suited for information retrieval applications [37], [83], [148]. Several modifications of Lee's original design have been proposed [26], [55], and some consideration has been given to the economical implementation of such a system [26], [28].

Bit-Serial Systems

More than a decade ago, Shooman [138] introduced the concept of parallel processing with vertical data, which essentially consists of processing one bit of a large number of words in parallel. Since the number of words to be processed is usually larger than the number of bits in each word, in applications for which associative processing techniques have been proposed, this approach represents a compromise between fully parallel and word-serial processing. The introduction of this concept resulted in a large number of practically realizable proposals for associative memories and processors [10], [14], [41], [50], [52], [66], [76], [127], [129], [150] and also made the use of memory elements with destructive readout possible [22]. It should be noted that byte-serial associative devices may also be conceived which fall between bit-serial and fully parallel systems.

Word-Serial Systems

When an associative device is viewed as a "black box" with certain functional capabilities, we note that parallel operation on all the words is not essential. In fact, if the words are operated upon serially with a very high speed, a reasonable speed of operation may be obtained with moderate cost. This is the design philosophy for word-serial associative devices [30], [80], [130]. A word-serial associative device essentially represents hardware implementation of a simple program loop that is used for linear search. Two factors contribute to the relative efficiency of this approach as compared with programmed linear search. 1) Instruction decoding time is reduced since the search operation requires a single instruction on the associative device. 2) The data rates achieved by circulating memories are much higher than those of random access memories.

Block-Oriented Systems

For applications such as information storage and retrieval where a large storage capacity is required, neither bit-serial nor word-serial systems offer an acceptable solution. Bit-serial systems become very expensive while word-serial systems result in excessive processing time. As a result, several attempts have been made to provide associative capabilities for mass storage [67], [102], [113]. Slotnick [145] proposes a logic-per-track device that consists of a head-per-track disk memory with some logic associated with each track. His system exploits the very high potential data rates of head-per-track disks and is suitable for applications requiring quantity of storage which presently suffer from the high cost of random access memories or from performance degradation due to frequent transfers between primary and secondary storage. A system based on Slotnick's idea and Lee's distributed-logic

memory [82], [83] has been proposed by Parhami [112] for information storage and retrieval applications. The author's conclusion is that block-oriented associative processing constitutes the most promising architecture for applications such as information storage and retrieval.

The Problem of Multiple Responses

One of the unique problems encountered in associative devices is that of multiple responses. In general, a search operation may yield a set of responders. Obviously, the purpose of a search is to identify a set of arguments on which a specific operation is to be performed. If this operation can be performed in the associative memory or processor, no problem arises. However, if members of the set should be read out and transferred to other subsystems, some means should be provided to indicate the number of responders and to select them in some order.

The first problem is to identify the number of responders. The simplest scheme yields a binary indication: no responder or some responders. A more sophisticated scheme may provide a ternary indication: no responder, exactly one responder, or more than one responder. Such a ternary response is useful, for example, for ordered retrieval from an associative memory [136]. In some applications, an exact or approximate knowledge of the number of responders may be of some value. An estimate for the number of responders may be obtained by analog summing techniques [76]. Obtaining the exact number of responders amounts to finding the sum of a set of 1-b binary numbers [48].

The second problem is to select one member from the set of responding words. The selection may be at random or according to some priority scheme, e.g., according to physical location of members of the set in memory matrix. Software or hardware approaches may be used for this purpose. The most straightforward software solution is to examine the words one at a time (or in blocks) sequentially in some order and select the first responder. Another approach is to use ordered retrieval techniques, i.e., to select a responder with a maximum (or minimum) numerical value in a particular field. This method, of course, requires that the value stored in that particular field of each word be unique [49], [100]. A more detailed discussion of ordered retrieval will be given later.

The simplest hardware approach is to detect the first "1" in the response store by shifting [22]. A second approach is to arrange the elements of the response store into a two-dimensional array. A first interrogation of this array causes row and column indicators to be set which indicate active rows and columns. Then the intersections of such columns and rows are interrogated to find the actual responders [160]. Hilberg [70] proposes a modification of this method to reduce the number of interrogations. In the fully parallel approach, a combinational logic net is used which either selects a responding word directly or gives its address for subsequent retrieval [2], [47], [160].

Reliability and Fault Tolerance

A concept of fundamental importance in computing systems is that of fault tolerance, which may be defined as the proper execution of programs (possibly in a degraded mode) despite the presence of hardware faults or software errors. The problem of fault tolerance of associative devices has not been explored except for a few isolated attempts [41]. It has generally been recognized that associative devices possess the property of graceful degradation; i.e., the failure of a cell does not necessarily affect the operation of the whole system. Furthermore, such faulty cells can be effectively isolated from the others [124].

Except for the word-serial systems, the protection of associative devices against faults is complicated because of the high degree of internal complexity and the large number of concurrent operations that take place at the bit level. Performing operations at the bit level precludes the use of conventional coding techniques with low redundancy. An effective method for concurrent detection of failures is the utilization of 1-out-of-2 encoding for all logic variables (double-rail implementation). Component replication constitutes another hardware redundancy technique that is very effective for protection of simple but critical parts, such as the input and mask registers, against failures. The array of cells can be divided into modules that can be replaced by spares through switching in the event of a failure detection. Alternatives available range from replacement of one cell to the replacement of the entire array.

In some applications of associative devices [25], [128], it has been reported that only a small fraction of real time is needed for the performance of processing functions assigned to the system. This makes the use of software and time-redundancy techniques attractive. Software redundancy can be used in the form of periodic diagnostic routines and the addition of extra steps to algorithms for checking critical components. Time redundancy consists of repetition of operations because of the possibility of faults, or program restarts in the case of failures.

Hardware Implementation

So far, we have considered different architectures of associative devices without specific reference to the functions they perform and their implementation. In this section, we will discuss the basic functions of associative devices, hardware elements used to implement them, and their physical characteristics.

Basic Operations

Different operations that must be performed by an associative device depend on the application. Theoretically, only exact match operation with masking (i.e., marking all cells whose contents in a certain field match that of a given key) and multiwrite operation with masking (i.e., writing a certain field of a given key into corresponding fields of a set of marked cells) are sufficient for performing any logic operation. However, the inefficiency of some such operations may force one to implement more complex primitives directly in hardware. In this section, we will consider some operations that may be implemented as primitives.

The simplest search operations are exact match (equality) and its complement (inequality). An equality (inequality) search operation marks all cells whose contents match (do not match) the unmasked portion of a key. The marking may be done by setting or resetting a flip-flop that is associated with each cell. Another search operation that may be useful in some applications is approximate match (similarity). A similarity search operation marks all the cells whose contents approximately match the unmasked portion of a key, e.g., mismatch in at most k positions, where k is a specified integer.

Other search operations are applicable to numerical arguments and include the following [65]:

Less than	Greater than
Less than or equal to	Greater than or equal to
Maximum value	Minimum value
Between limits	Not between limits
Next higher	Next lower.

The search for maximum (minimum) value requires no external argument except possibly for a mask. This operation marks all cells for which the numerical value in an unmasked field is a maximum (minimum) and is useful for ordered retrieval applications [62]. The between-limits (not-between-limits) search operation is a composite search operation that may be performed by a sequence of simple searches such as greater-than and less-than searches [165]. Actual implementation of all numerical search operations depends on the representation chosen for negative numbers.

Other operations, which we will discuss, are read, write, and arithmetic operations. The read operation is performed either by using an address, as in conventional memories, or by enabling a cell directly by a multiple response resolver. There are two types of write operations: simple write, which is similar to read operation, and multiwrite, which is a very powerful operation used for modifying a given field of an arbitrary number of cells [116]. Arithmetic operations are usually performed as sequences of other basic operations. Estrin and Fuller [39] give algorithms for arithmetic operations in terms of elementary operations that they define. These algorithms are performed in a bit-serial word-parallel fashion and assume 2's complement representation of negative numbers. A fully parallel addition capability can be provided at the expense of more hardware. Then simple add and multiadd operations can be defined in a manner similar to simple write and multiwrite.

Hardware Elements

Early advances in the design and implementation of associative devices resulted from the development of cryogenic or superconductive circuits. The idea was first presented by Slade and McMahon [141], who described a cryogenic catalog memory. Recent developments in magnetic-film and integrated-circuit technologies and the recognition of fabrication problems and high refrigeration and maintenance costs for cryogenic devices have diverted the attention of researchers from superconductive associative devices. In this section, we will discuss the use of various elements for the implementation of associative devices, without being concerned with their principles of operation. Reviews of various implementations have appeared in the literature [24], [65]. In addition to cryoelectrics, magnetic elements, and semiconductors, which will be discussed subsequently, other techniques such as the use of glass delay lines and holography have been proposed for the implementation of associative capabilities [54], [88], [120], [130], [131], [146], [147].

With respect to cryoelectric elements, we have already referred to the early contribution of Slade and McMahon [141]. Their work was later expanded in subsequent reports [142]–[144]. A large number of other researchers have since investigated various problems in the design of cryogenic associative devices [1], [8], [28], [31], [75], [92], [108], [109], [117], [118], [124], [133], [167], [168]. The main advantage of cryogenic elements for the implementation of associative devices is that they can be economically mass produced. The main disadvantages of cryogenic elements that make them unsuitable for implementing associative devices (small ones in particular) are their high refrigeration cost and maintenance problems.

The problems with cryoelectric components and the highly advanced magnetic memory technology have resulted in an interest in magnetic associative devices. Both discrete and continuous magnetic memory elements have been proposed for the implementation of associative devices. With respect to discrete components, consideration has been given to magnetic cores, transfluxors, biax cores, and multiaperture logic elements [3], [43], [66], [81], [91], [94], [95], [123]. Recently, magnetic thin-film, and in particular plated-wire, memories have become popular for realizing bit-serial associative devices. The following advantages have been stated for plated-wire associative devices over magnetic devices using discrete components: 1) reduction of the cost of memory array through batch fabrication; 2) improvement of operating speed; and 3) achievement of multiwrite capability with small cost in circuitry and power, due to small word currents. Many associative devices have been proposed which use continuous magnetic elements [20], [23], [50], [51], [98], [106], [121], [125], [128], [158].

The advent of large-scale integration (LSI) has resulted in a large number of proposals for solid-state associative devices. Lee [85] states the following advantages for solid-state associative cells. 1) High signal-to-noise ratio allows long associative memory words. 2) The equivalence of input and output energy levels minimizes crosstalk problems. 3) The input energy requirement can be made very small. 4) Output signals are compatible with the external logic. 5) Components with very loose tolerances can be used.

Most proposals deal with the implementation of simple associative memories [6], [9], [13], [16], [69], [71], [72], [79], [86], [156], [157]. It has been realized recently that with LSI, more complex operations required for associative processors can be implemented in hardware.

Physical Characteristics

In this section, we briefly review the speed, size, and cost of associative devices with respect to different architectures and implementations. Clearly, the parallel processing capability of associative devices can provide orders-of-magnitude increases in data processing speeds in some applications that are limited by the sequential character of the conventional von Neumann computer organization [126], [127]. The actual speed gain, however, depends on the application. Here we discuss the speed of associative devices without any comparison with conventional systems.

Speed of fully parallel associative devices depends on the operations implemented in hardware and on the hardware elements used. Wald [157] reports that MOS associative memories may be constructed with a cycle time of 300 ns for read, write, and equality search operations. The cycle time of such devices increases as more complex search and arithmetic operations are implemented because of the requirement for carry or borrow propagation. The time required for an operation in bit-serial associative devices is a linear function of the number of bits involved in the operation, except possibly for read and write operations. Rudolph *et al.* [128] report a plated-wire mechanization that requires from 100 to 300

ns/b for search operations. Of course, some overhead (of the order of 500 ns) is also required which is independent of the number of bits. Operation time in word-serial associative devices depends on the size of their memory array. Rux [130] reports a 2048-word associative memory using glass delay lines with a cycle time of 100 μs. The speed of block-oriented systems has not yet been evaluated. The RAPID system proposed by Parhami [112] is capable of searching a disk file in a few seconds, depending on the search criteria.

Two factors limit the size of associative devices: economic considerations and technical constraints. Economic limitations may be overcome by considering different organizations that provide tradeoffs between speed, size, and cost. Technical limitations include the half-select noise (limiting the word length) and interrogation drive problems (limiting the number of words). Cryogenic techniques were believed to allow large-capacity associative devices of fully parallel type (10^7 to 10^9 b). However, this conjecture has not yet materialized. Only very small semiconductor associative devices are practical today. The magnetic associative devices that have been proposed typically contain a few thousand words. These are mainly bit-serial devices. Large capacities are not practical with the word-serial approach, either. Block-oriented associative devices allow very large capacities (10^9 b) at the expense of a reduction in operation speed. The physical size of associative devices can be considerably reduced by achieving higher circuit densities [104].

The major drawback to the use of associative devices has been their relatively high cost. Even with such high costs, associative devices have been more economical than conventional systems for some applications. Rudolph *et al.* [128] project that for the air traffic control application considered, an associative processor will cost less than 20 percent of a conventional system. More economical realizations of associative devices are possible through batch fabrication techniques [78], [166].

Software Considerations

The concepts of associative processing, discussed in the preceding sections, are very important. Equally important, however, are methods that enable a user to make efficient use of these capabilities. In this section, we will discuss these methods. We will also examine software simulation of associative processing capabilities.

Algorithms

A problem that arises frequently in using most computing systems is that of synthesizing desired operations from a set of primitive operations implemented in hardware. Using Iverson's notation,[2] Falkoff [42] has described algorithms for associative memories. Estrin and Fuller [39] have developed algorithms for arithmetic and complex search operations for an associative memory that has as basic commands exact-match search and some read and write operations. They give execution times for these algorithms in terms of the execution time for basic commands and estimate the latter for several hardware implementations. Other researchers have also described search algorithms [45], [165], addition algorithms [96], [169], and algorithms for floating-point operations [154].

[2] K. E. Iverson, *A Programming Language*. New York: Wiley, 1962.

An interesting set of algorithms for associative devices includes those dealing with ordered retrieval. Ordered retrieval may be defined as the retrieval of the members of a set of words in ascending or descending order of the numerical value stored in a certain field. It is assumed that this value is unique for each word. Seeber and Lindquist [136] describe an associative memory with ordered retrieval. Lewin [87] has developed a technique that requires $2m-1$ cycles for complete readout of m words. Wolinsky [163] has developed a new proof for Lewin's result. Other relevant work appears in the literature [1], [19], [44], [73].

Programming Aspects

So far, the programming of associative devices has been done mainly at the machine language or microprogram level. At the machine language level, a set of instructions is defined which can be combined to form a program [39], [129]. These instructions vary in complexity and many of them may actually be executed as a sequence of more primitive commands. At the microprogram level, each field of a microinstruction controls a subsystem or transfer path within the associative device [41].

Emphasis on programming aspects of associative devices has been mixed. Some researchers emphasize the fact that programming for associative devices is difficult to learn because of their conceptual differences with more conventional systems [34]. Others point out that a large percentage of programming time and effort, for conventional systems, is usually spent in assigning and keeping track of addresses and that addressing by contents frees us to a large extent from considerations such as scanning and searching [1], [83].

Software Simulation

Software simulation of associative processing capabilities has been used for at least two reasons— to verify a hardware design and/or develop programs for it, and to replace hardware associative devices that have been prohibitively expensive. Functional behavior of associative memories with exact match as the only operation can be simulated by hash coding techniques. Several systems use software simulation of associative processing capabilities. The multilist system [119] uses a simulated associative memory for applications such as information retrieval and man–machine communication. The association storing processor [132] is suitable for information storage and retrieval applications. The TRAMPS question-answering system [4] also uses a software-simulated associative memory.

Applications

Numerous applications have been suggested for associative devices. These applications have one or more of the following properties: 1) require fast search of a large data base; 2) require the performance of arithmetic and logic operations on large sets of data; or 3) deal with dynamically changing data bases. Hanlon [65] notes that, for most suggested applications, the superiority of the associative processing approach has not been proved analytically or experimentally. However, even the small set of applications for which associative devices have been clearly shown to be superior is enough to justify such organizations for special-purpose use. These applications include solution of some mathematical systems, radar signal processing, information storage and retrieval,

and performance of control functions in computer systems. The first two are primarily numerical applications while the last two are nonnumerical (symbolic) in nature.

In addition to these four applications, associative devices have been suggested for areas such as space applications [63], artificial intelligence [4], [52], [53], [140], [170], language and code translation [15], [103], symbol manipulation [135], interactive communication [139], [149], logic design [93], and sorting [134]. These are mainly nonnumerical applications.

Solution of Mathematical Systems

Solution of mathematical systems refers to numerical applications such as function optimization [40], solution of differential equations [40], [58], matrix computations [77], and graph manipulation [29]. Estrin and Fuller [40] compare the solution time for a particular partial differential equation in three computer organizations: an associative computer, a conventional computer (IBM 7090), and the Solomon computer.[3] They conclude that an associative computer is considerably faster than a conventional one, but somewhat slower than the Solomon computer, which has a considerably more complex structure. Matrix computations are usually characterized by operations that involve doing the same thing to a number of matrix elements, e.g., an entire row or column. Using a number of simplifying assumptions, Katz [77] shows that for most operations on nth-order matrices, for large n, a speedup factor proportional to n can be obtained by using an associative processor.

Radar Signal Processing

Many researchers have investigated the applications of associative processing techniques to radar signal processing [14], [25], [35], [36], [59], [74], [97], [98], [99], [152], [161]. It is expected that much of the work in this area has not been reported because of its classified nature. Eddey [35] compares an associative processor to a general-purpose computer for radar tracking and correlation. The significant speed advantage of associative devices for a large number of targets leads him to conclude that their use will lead to automatic tracking systems of greatly increased capability and reliability, and reduced cost and size. Other researchers draw similar conclusions [14], [59].

A problem for which associative processors seem to be ideally suited is air traffic control [36], [97], [98], [99], [152]. Rudolph *et al.* [128] estimate a more than 200:1 speed advantage over a conventional system for solving and updating the collision detection problem for 128 aircrafts.

Information Storage and Retrieval

Suitability of the fast search capability of associative memories for information storage and retrieval applications was recognized over 10 years ago [61], [114]. Among the early proposals were the distributed-logic memory [82], [83], [90] and the multiple instantaneous response file [111], [171]. Other pertinent work has also been reported [115], [122].

With the state-of-the-art technology, it is obviously impractical to have an associative memory large enough to store all of the desired information [170]. The use of small associative memories in conjunction with secondary storage [34] results in the expenditure of considerable amounts of time in loading and unloading the associative memory. A possible solution that permits a reasonable tradeoff between speed and cost is the provision of associative capabilities for mass storage [67], [102], [112], [145].

Control Functions in Computer Systems

Many control and executive functions in computer systems can benefit from the use of associative devices. Foremost among these is the control of dynamic storage allocation schemes [21]. Time-sharing systems can use associative memories to control the page transfers [5], [68], [89]. Similarly, computer systems using a fast buffer storage between the main memory and the processing unit may use associative control for block transfers.[4] Associative memories may also be used for resource allocation [11], [12] and other executive functions within computer systems [38], [64].

Conclusion

We have provided an introduction to the basic concepts of associative processing and a guide to the literature in this field. Our conclusion is that the concept of associative processing is an important one and will be used extensively in special-purpose and general-purpose computers of the future. Many practical systems have already been built. A partial listing of delivered associative memories has been given by Rudolph *et al.* [127]. Among the special-purpose associative computers built to date, the most notable is the parallel element processing ensemble [60] which is designed for radar data processing applications. Goodyear Aerospace Corporation is marketing a general-purpose associative computer, called Staran [10], [129], which consists of a bit-serial associative processor and appropriate controls.

Despite the large volume of published literature, many aspects of associative processing remain to be explored. To assure reliable and efficient operation of associative devices, these areas need to be investigated.

1) The architectural properties of different organizations for associative devices (i.e., fully parallel, bit-serial, word-serial, and block-oriented) must be studied. The tradeoffs, in speed and cost, available to the designer must be identified.

2) Techniques need to be devised for the introduction of fault tolerance in associative devices to allow correct program execution despite the presence of hardware faults. The problem of testing and maintenance of such devices must also be examined.

3) More concrete performance evaluation results should be obtained for the proposed applications and new ones. Such results will aid the designers in deciding whether associative processing techniques are suitable for the applications being considered.

4) Software techniques must be introduced to simplify the programming aspects of associative devices. An associative computer is of little value for general-purpose use unless it can be programmed easily and efficiently.

To summarize, two properties of associative devices distinguish them from conventional systems. 1) Stored items can

[3] D. L. Slotnick, W. C. Borck, and R. C. McReynolds, "The Solomon computer," in *1962 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Spartan, 1962, pp. 97–107.

[4] J. S. Liptay, "Structural aspects of the system/360 model 85: II—the Cache," *IBM Syst. J.*, vol. 7, pp. 15–21, 1968.

be retrieved without a need for knowing their physical location. 2) Data transformation operations can be performed over many sets of arguments (possibly in parallel) with a single instruction.

The advantages of such devices for the proposed applications include increased speed, simplified software, structural regularity, simple growth, and graceful degradation. In particular, the structural regularity of associative devices justifies the replacement of logic circuits by functional memory organizations [46], [57].

Acknowledgment

The author wishes to thank Prof. A. Avizienis and Prof. W. W. Chu, under whose direction this work was performed, for their guidance and encouragement.

References

[1] R. W. Ahrons, "Superconductive associative memories," *RCA Rev.*, vol. 24, pp. 325–354, Sept. 1963.
[2] ——, "Calculations of speed of ladder network for superconductive associative memories," *IEEE Trans. Electron. Comput.* (Short Notes), vol. EC-14, pp. 267–270, Apr. 1965.
[3] A. Apicella and J. Franks, "BILOC—A high-speed NDRO one-core-per-bit associative element," presented at Int. Conf. on Magnetics, Apr. 1965.
[4] W. L. Ash and E. H. Sibley, "TRAMP: An interpretive associative processor with deductive capabilities," in *ACM Nat. Conf. Proc.*, pp. 143–156, 1968.
[5] D. Aspinall, D. J. Kinniment, and D. B. G. Edwards, "Associative memories in large computer systems," *Inform. Process.*, pp. 796–800, 1968.
[6] ——, "An integrated associative memory matrix," *Inform. Process.*, pp. 800a–800f, 1968.
[7] J. Atkin and N. B. Marple, "Information processing by data interrogation," *IEEE Trans. Electron. Comput.*, vol. EC-11, pp. 181–187, Apr. 1962.
[8] J. D. Barnard, F. A. Behnke, A. B. Lindquist, and R. R. Seeber, "Structure of a cryogenic associative processor," *Proc. IEEE*, vol. 52, pp. 1182–1190, Oct. 1964.
[9] J. Bartlett *et al.*, "Associative memory chips: Fast, versatile—and here," *Electronics*, pp. 96–100, Aug. 17, 1970.
[10] K. E. Batcher, "Flexible parallel processing and STARAN," in *WESCON Tech. Papers* (session 1), Sept. 1972.
[11] R. O. Berg and M. D. Johnson, "An associative memory for executive control functions in an advanced avionics computer system," in *IEEE Int. Computer Group Conf. Proc.*, pp. 336–342, June 1970.
[12] R. O. Berg and K. J. Thurber. "A hardware executive control for the advanced avionic digital computer system," in *NAECON Rec.*, pp. 206–213, May 1971.
[13] A. W. Bidwell and W. D. Pricer, "A high-speed associative memory," in *1967 Int. Solid-State Circuits Conf. Proc.*, pp. 78–79.
[14] R. M. Bird, "An associative memory parallel deltic realization for active sonar signal processing," in *Parallel Processor Systems, Technologies, and Applications*, L. C. Hobbs *et al.*, Eds. New York: Spartan, 1970, pp. 107–129.
[15] J. R. Brown, Jr., "A semi-permanent magnetic associative memory and code converter," presented at Special Tech. Conf. on Nonlinear Magnetics, Los Angeles, Calif., Nov. 1961.
[16] J. R. Burns and J. H. Scott, "Silicon-on-sapphire complementary MOS circuits for high-speed associative memory," in *1969 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 469–477.
[17] A. V. Campi, R. M. Dunn, and B. H. Gray, "Content addressable memory system concepts," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-1, pp. 168–173, Oct. 1965.
[18] V. Chlouba, "The use of codes 'm-out-of-n' in associative memories," *Inform. Process. Mach.*, vol. 13, pp. 113–138, 1967.
[19] ——, "Ordered retrieval from a decimal associative memory," *Inform. Process. Mach.*, vol. 13, pp. 139–155, 1967.
[20] W. F. Chow, "Plated wire content-addressable memories with bit-steering technique," *IEEE Trans. Electron. Comput.*, vol. EC-16, pp. 642–652, Oct. 1967.
[21] Y. Chu, "Application of content-addressed memory for dynamic storage allocation," *RCA Rev.*, vol. 26; pp. 140–152, Mar. 1965.
[22] ——, "A destructive-readout associative memory," *IEEE Trans. Electron. Comput.*, vol. EC-14, pp. 600–605, Aug. 1965.
[23] A. Corneretto, "3-K bit associative memory works at room temperature," *Electron. Des.*, vol. 10, pp. 8–11, July 5, 1962.
[24] ——, "Associative memories, a many-pronged design effort," *Electron. Des.*, vol. 11, pp. 40–55, Feb. 1, 1963.
[25] A. Costanzo and J. Garrett, "Application of an associative processor to an interceptor radar system," in *Nat. Aerospace Electronics Conf. Proc.*, pp. 107–112, 1969.
[26] B. A. Crane and J. A. Githens, "Bulk processing in distributed logic memory," *IEEE Trans. Electron. Comput.*, vol. EC-14, pp. 186–196, Apr. 1965.
[27] B. A. Crane, "Economics of the DLM, a batch-fabricatable parallel computer," presented at Conf. on the Impact of Batch Fabrication on Future Computers, Apr. 1965.
[28] B. A. Crane and R. R. Laane, "A cryoelectronic distributed logic memory," in *1967 Spring Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 517–524.
[29] B. A. Crane, "Path finding with associative memory," *IEEE Trans. Comput.* (Short Notes), vol. C-17, pp. 691–693, July 1968.
[30] W. A. Crofut and M. R. Sottile, "Design techniques of a delay-line content-addressed memory," *IEEE Trans. Electron. Comput.*, vol. EC-15, pp. 529–534, Aug. 1966.
[31] P. M. Davies, "A superconductive associative memory," in *1962 Spring Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 79–88.
[32] ——, "Design of an associative computer," in *IEEE Pacific Computer Conf. Proc.*, Mar. 1963, pp. 109–117.
[33] R. B. Derickson, "A proposed associative push down memory," *Comput. Des.*, vol. 7, pp. 60–66, Mar. 1968.
[34] J. A. Dugan, R. S. Green, J. Minker, and W. E. Shindle, "A study of the utility of associative memory processors," in *ACM Nat. Conf. Proc.*, pp. 347–360, Aug. 1966.
[35] E. E. Eddey, "The use of associative processors in radar tracking and correlation," in *Nat. Aerospace Electronics Conf. Proc.*, pp. 39–42, 1967.
[36] ——, "Ground-based aircraft collision avoidance using an associative processor," in *Nat. Aerospace Electronics Conf. Proc.*, pp. 302–306, 1970.
[37] R. P. Edwards, "Content-addressable distributed-logic memories," *Proc. IEEE* (Corresp.), vol. 52, pp. 83–84, Jan. 1964.
[38] J. D. Erwin and D. Jensen, "Interrupt processing with queued content-addressable memories," in *1970 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 621–627.
[39] G. Estrin and R. Fuller, "Algorithms for content-addressable memories," in *IEEE Pacific Computer Conf. Proc.*, pp. 118–128, Mar. 1963.
[40] ——, "Some applications for content-addressable memories," in *1963 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Spartan, pp. 495–508.
[41] R. G. Ewing and P. M. Davies, "An associative processor," in *1964 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 147–158.
[42] A. D. Falkoff, "Algorithms for parallel search memories," *J. Ass. Comput. Mach.*, vol. 9, pp. 488–511, Oct. 1962.
[43] T. Y. Feng, "A magnetic associative memory," in *1968 Spring Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 275–281.
[44] ——, "The structure of a high-speed associative processor," in *Nat. Electronics Conf. Proc.*, pp. 257–262, 1968.
[45] ——, "Search algorithms for associative memories," in *Proc. 4th Annu. Princeton Conf. on Information Sciences and Systems*, pp. 422–426, 1970.
[46] M. Flinders, P. L. Gardner, R. J. Llewelyn, and J. F. Minshull, "Functional memory as a general purpose systems technology," in *IEEE Int. Computer Group Conf. Proc.*, pp. 314–324, June 1970.
[47] C. C. Foster, "Determination of priority in associative memories," *IEEE Trans. Comput.* (Short Notes), vol. C-17, pp. 788–789, Aug. 1968.
[48] C. C. Foster and F. D. Stockton, "Counting responders in an associative memory," *IEEE Trans. Comput.*, vol. C-20, pp. 1580–1583, Dec. 1971.
[49] E. H. Frei and J. Goldberg, "A method for resolving multiple responses in a parallel search file," *IRE Trans. Electron. Comput.*, vol. EC-10, pp. 718–722, Dec. 1961.
[50] L. C. Fulmer and W. C. Meilander, "A modular plated wire associative processor," in *IEEE Int. Computer Group Conf. Proc.*, pp. 325–335, June 1970.
[51] R. H. Fuller, J. C. Tu, and R. M. Bird, "A woven plated-wire associative memory," in *Nat. Aerospace Electronics Conf. Proc.* (Abstract), p. 24, 1965.
[52] R. H. Fuller and R. M. Bird, "An associative parallel processor with application to picture processing," in *1965 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 105–116.
[53] R. H. Fuller, "Associative parallel processing," *Comput. Des.*, vol. 6, pp. 43–46, Dec. 1967.
[54] D. Gabor, "Associative holographic memories," *IBM J. Res. Devel.* vol. 13, pp. 156–159, Mar. 1969.
[55] R. S. Gains, and C. Y. Lee, "An improved cell memory," *IEEE Trans. Electron. Comput.* (Short Notes), vol. EC-14, pp. 72–75, Feb. 1965.

[56] R. G. Gall, "A hardware-integrated GPC/search memory," in *1964 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson; pp. 159–173.
[57] P. L. Gardner, "Functional memory and its microprogramming implications," *IEEE Trans. Comput.*, vol. C-20, pp. 764–775, July 1971.
[58] P. A. Gilmore, "Numerical solution of partial differential equations by associative processing," in *1971 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 411–418.
[59] J. A. Githens, "An associative, highly-parallel computer for radar data processing," in *Parallel Processor Systems, Technologies, and Applications*, L. C. Hobbs *et al.*, Eds. New York: Spartan, 1970, pp. 71–86.
[60] ——, "PEPE: A parallel processor," in *IEEE Int. Computer Society Conf. Proc.*, pp. 57–72, Sept. 1972.
[61] J. Goldberg and M. W. Green, "Large files for information retrieval based on simultaneous interrogation of all items," in *Large-Capacity Memory Techniques for Computing Systems*. M. C. Yovits, Ed. New York: Macmillan, 1962, pp. 63–67.
[62] E. J. Guass, "Locating the largest word in a file using a modified memory," *J. Ass. Comput. Mach.*, vol. 8, pp. 418–425, July 1961.
[63] D. C. Gunderson *et al.*, "Associative memories in space applications," in *WESCON Tech. Papers* (session 9), 1966.
[64] D. C. Gunderson, W. L. Heimerdinger, and J. P. Francis, "A multiprocessor with associative control," in *Prospects for Simulation and Simulators of Dynamic Systems*. New York: Spartan, 1967, pp. 183–200.
[65] A. G. Hanlon, "Content-addressable and associative memory systems: A survey," *IEEE Trans. Electron. Comput.*, vol. EC-15, pp. 509–521, Aug. 1966.
[66] P. A. Harding and M. W. Rolund, "A 2-1/2D core search memory," in *1968 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, 1968, pp. 1213–1218.
[67] L. D. Healy, G. J. Lipovski, and K. L. Doty, "The architecture of a context addressed segment-sequential storage," in *1972 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 691–701.
[68] L. Hellerman and G. E. Hoernes, "Control storage use in implementing an associative memory for a time-shared processor," *IEEE Trans. Comput.*, vol. C-17, pp. 1144–1151, Dec. 1968.
[69] R. F. Herlein and A. V. Thompson, "An integrated associative memory element," in *Int. Solid-State Circuits Conf. Proc.*, pp. 42–43, 1969.
[70] W. Hilberg, "Simultaneous multiple response in associative memories and readout of the detector matrix," *IEEE Trans. Electron. Comput.*, vol. EC-15, pp. 117–118, Feb. 1966.
[71] R. Igarashi, T. Kurosawa, and T. Yaita, "A 150-nanosecond associative memory using integrated MOS transistors," in *Int. Solid-State Circuits Conf. Proc.*, pp. 104–105, Feb. 1966.
[72] R. Igarashi and T. Yaita, "An integrated MOS transistor associative memory system with 100 ns cycle time," in *1967 Spring Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.; AFIPS Press, pp. 499–506.
[73] L. R. Johnson and M. H. McAndrew, "On ordered retrieval from an associative memory," *IBM J. Res. Devel.*, vol. 8, pp. 189–193, Apr. 1964.
[74] E. C. Josef and A. Kaplan, "Target-track correlation with a search memory," in *Proc. Nat. Conv. on Military Electronics* (Washington, D. C., June 1962), pp. 255–261.
[75] H. G. Kadereit and K. Goser, "A continuous film memory cell for superconductive associative memories," *Proc. IEEE* (Lett.), vol. 56, pp. 121–122, Jan. 1968.
[76] A. Kaplan, "A search memory subsystem for a general-purpose computer," in *1963 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 193–200.
[77] J. H. Katz, "Matrix computations on an associative processor," in *Parallel Processor Systems, Technologies and Applications*, L. C. Hobbs *et al.*, Eds. New York: Spartan, 1970, pp. 131–149.
[78] W. H. Kautz, "An augmented content-addressed memory array for implementation with large-scale integration," *J. Ass. Comput. Mach.*, vol. 18, pp. 19–33, Jan. 1971.
[79] D. J. Kinniment, A. E. Knowles, and D. B. G. Edwards, "An integrated associative storage system," in *Proc. Int. Conf. on Microelectronics* (London, June 1969), pp. 37–38.
[80] W. K. King, "Design of an associative memory," *IEEE Trans. Comput.* (Short Notes), vol. C-20, pp. 671–674, June 1971.
[81] J. R. Kiseda, H. E. Petersen, W. C. Seelbach, and M. Teig, "A magnetic associative memory," *IBM J. Res. Devel.*, vol. 5, pp. 106–121, Apr. 1961.
[82] C. Y. Lee, "Intercommunicating cells, basis for a distributed logic computer," in *1962 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 130–136.
[83] C. Y. Lee and M. C. Paull, "A content addressable distributed logic memory with applications to information retrieval," *Proc. IEEE*, vol. 51, pp. 924–932, June 1963.
[84] C. Y. Lee, "Content-addressable and distributed logic memories," in *Applied Automata Theory*, J. T. Tou, Ed. New York: Academic Press, 1968.
[85] E. S. Lee, "Semiconductor circuits in associative memories," in *IEEE Pacific Computer Conf. Proc.*, pp. 96–108, Mar. 1963.
[86] ——, "Associative techniques with complementing flip-flops," in *1963 Spring Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 381–394.
[87] M. H. Lewin, "Retrieval of ordered lists from a content-addressed memory," *RCA Rev.*, vol. 23, pp. 215–229, June 1962.
[88] M. H. Lewin, H. R. Beelitz, and J. A. Rajchman, "Fixed associative memory using evaporated organic diode arrays," in *1963 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 101–106.
[89] A. B. Lindquist, R. R. Seeber, and L. W. Comeau, "A time-sharing system using an associative memory," *Proc. IEEE*, vol. 54, pp. 1774–1779, Dec. 1966.
[90] G. J. Lipovski, "The architecture of a large associative processor," in *1970 Spring Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 385–396.
[91] R. R. Lussier and R. P. Schneider, "All-magnetic content addressed memory, *Electron. Ind.*, vol. 22, pp. 92–98, Mar. 1963.
[92] H. T. Mann and J. L. Rogers, "A cryogenic between-limits associative memory," in *Nat. Aerospace Electronics Conf. Proc.*, pp. 359–362, May 1962.
[93] R. M. Matney and C. H. Roth, "Associative computing structures for the solution of logic design problems," in *Proc. Southwestern IEEE Conf.*, pp. 4E1–4E8, 1969.
[94] J. E. McAteer, J. A. Capobianco, and R. L. Koppel, "Associative memory system implementation and characteristics," in *1964 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 81–92.
[95] W. L. McDermid and H. E. Petersen, "A magnetic associative memory system," *IBM J. Res. Devel.*, vol. 5, pp. 59–62, Jan. 1961.
[96] B. T. McKeever, "The associative memory structure," in *1965 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 371–388.
[97] W. C. Meilander, "The associative processor in aircraft conflict detection," in *Nat. Aerospace Electronics Conf. Proc.*, pp. 57–62, 1968.
[98] W. C. Meilander, M. Bialer, and J. Garrett, "A mission oriented associative processor using plated wire," in *Parallel Processor Systems, Technologies, and Applications*, L. C. Hobbs *et al.*, Eds. New York: Spartan, 1970, pp. 153–163.
[99] W. C. Meilander and R. G. Gall, "Evaluation of the Goodyear associative processor in an operational ATC environment," presented in IEEE Computer Society Conf., Boston, Mass., Sept. 1971.
[100] H. S. Miller, "Resolving multiple responses in an associative memory," *IEEE Trans. Electron. Comput.* (Short Notes), vol. EC-13, pp. 614–616, Oct. 1964.
[101] J. Minker, "An overview of associative or content-addressable memory systems and a KWIC index to the literature," *Comput. Rev.*, vol. 12, pp. 453–504, Oct. 1971.
[102] N. Minsky, "Rotating storage devices as partially associative memories," in *1972 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 587–595.
[103] R. C. Minnick, "Magnetic comparators and code converters," in *Proc. Symp. Application of Switching Theory in Space Technology*, pp. 193–204, Feb. 1962.
[104] J. L. Mundy, "A high density associative memory," presented at IEEE Computer Society Conf., Boston, Mass., Sept. 1971.
[105] J. C. Murtha, "Highly parallel information processing systems," in *Advances in Computers*, vol. 7. New York: Academic Press, 1966, pp. 1–116.
[106] M. Naiman, "Content-addressed memory using magnetoresistive readout of magnetic thin films," presented at the INTERMAG Conf., 1965.
[107] N. K. Natarajan and P. A. V. Thomas, "A multiaccess associative memory," *IEEE Trans. Comput.*, vol. C-18, pp. 424–428, May 1969.
[108] V. L. Newhouse and R. E. Fruin, "A cryogenic data addressed memory," in *1962 Spring Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 89–99.
[109] ——, "Data addressed memory using thin-film cryotrons," *Electronics*, vol. 35, pp. 31–36, May 4, 1962.
[110] S. Nissim, "Organizing the nanophile computers," *Electron. Des.*, vol. 11, pp. 44–53, Mar. 1, 1963.
[111] J. D. Noe, "MIRF (multiple instantaneous response file)," *Current Res. Devel. Sci. Document.*, vol. 9, Nov. 1961.
[112] B. Parhami, "A highly parallel computing system for information retrieval," in *1972 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 681–690.
[113] A. M. Peskin, "Associative capabilities for mass storage through array organization," in *1970 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 615–620.
[114] H. E. Petersen, "Content addressing and information retrieval," presented at IFIPS Congr., Aug. 1962.
[115] G. G. Pick, "A read-only multi-megabit parallel search associative

memory," presented at the 26th Annu. Meet. of American Documentation, Oct. 1963.
[116] S. N. Porter, "Use of multiwrite for general programmability of search memories," *J. Ass. Comput. Mach.*, vol. 13, pp. 369–373, July 1966.
[117] J. P. Pritchard and L. D. Wald, "Design of a fully associative cryogenic data processor," presented at the INTERMAG Conf., Apr. 1964.
[118] ——, "Design of a fully associative cryogenic data processor," *IEEE Trans. Magn.*, vol. MAG-1, pp. 68–71, Mar. 1965.
[119] N. S. Prywes and H. J. Gray, "The organization of a multilist-type associative memory," *IEEE Trans. Commun. Electron.*, vol. 82, pp. 488–492, Sept. 1963.
[120] C. F. Pulvari, "An improved field-controlled polarization-transfer device and the operating features of the exploratory content addressable memory system," *IEEE Trans. Electron Devices*, vol. ED-16, pp. 580–587, June 1969.
[121] J. I. Raffel, and T. S. Crowther, "A proposal for an associative memory using magnetic films," *IEEE Trans. Electron Comput.* (Short Notes), vol. EC-13, p. 611, Oct. 1964.
[122] D. L. Reich, "Associative memories and information retrieval," in *Some Problems in Information Science*, M. Kochen, Ed. New York: Scarecrow Press, 1965, pp. 217–224.
[123] A. D. Robbi and R. Ricci, "Transfluxor contest-addressable memory," in *Proc. Int. Conf. Nonlinear Magnetics*, pp. 831–837, Apr. 1964.
[124] R. F. Rosin, "An organization of an associative cryogenic computer," in *1962 Spring Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 203–212.
[125] C. A. Rowland and W. O. Berge, "A 300-nanosecond search memory," in *1963 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 59–65.
[126] J. A. Rudolph, "The associative processor—A new computer resource," in *IEEE Region 6 Conf. Proc.*, pp. 179–183, Apr. 1969.
[127] J. A. Rudolph, L. C. Fulmer, and W. C. Meilander, "With associative memory, speed limit is no barrier," *Electronics*, vol. 43, pp. 96–101, June 22, 1970.
[128] ——, "The coming of age of the associative processor," *Electronics*, vol. 44, pp. 91–96, Feb. 15, 1971.
[129] J. A. Rudolph, "A production implementation of an associative array processor: STARAN," in *1972 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 229–241.
[130] P. T. Rux, "A glass delay line content-addressable memory system," *IEEE Trans. Comput.*, vol. C-18, pp. 512–520, June 1969.
[131] M. Sakaguchi, N. Nishida, and T. Nemoto, "A new associative memory system utilizing holography," *IEEE Trans. Comput.*, vol. C-19, pp. 1174–1181, Dec. 1970.
[132] D. A. Savitt, H. H. Love, and R. E. Troop, "ASP: A new concept in language and machine organization," in *1967 Spring Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 87–102.
[133] R. R. Seeber, "Cryogenic associative memory," presented at ACM Nat. Conf., Aug. 1960.
[134] ——, "Associative self-sorting memory," in *Proc. 1960 EJCC*, pp. 179–187.
[135] ——, "Symbol manipulation with an associative memory," in *ACM Nat. Conf. Proc.*, pp. 5B-4-1–5B-4-4, 1961.
[136] R. R. Seeber and A. B. Lindquist, "Associative memory with ordered retrieval," *IBM J. Res. Devel.*, vol. 6, pp. 126–136, Jan. 1962.
[137] ——, "Associative logic for highly parallel systems," in *1963 Fall Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 489–493.
[138] W. Shooman, "Parallel computing with vertical data," in *Proc. 1960 EJCC*, pp. 111–115.
[139] E. H. Sibley, D. G. Gordon, and R. W. Taylor, "Graphical systems communications—An associative memory approach," in *1968 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 545–555.
[140] G. J. Simmons, "Application of an associatively addressed, distributed memory," in *1964 Spring Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 493–513.
[141] A. Slade and H. O. McMahon, "A cryotron catalog memory system," in *Proc. 1956 EJCC*, pp. 115–120.
[142] A. E. Slade and C. R. Smallman, "Thin film cryotron catalog memory," *Automat. Contr.*, pp. 48–50, Aug. 1960.
[143] ——, "Thin film cryotron catalogue memory," *Solid-State Electron.*, vol. 11, pp. 357–362, Sept. 1960.
[144] A. E. Slade and H. O. McMahon, "A cryotron catalog memory system," in *Proc. 1960 EJCC*, p. 120.
[145] D. L. Slotnick, "Logic per track devices," in *Advances in Computers*, vol. 10. New York: Academic Press, 1970, pp. 291–296.
[146] D. O. Smith and K. J. Harte, "Content-addressed memory using magneto- or electro-optical interrogation," *IEEE Trans. Electron. Comput.* (Corresp.), vol. EC-15, p. 123, Feb. 1966.
[147] ——, "Comment on 'Content-addressed memory using magneto- or electro-optical interrogation'," *IEEE Trans. Electron. Comput.* (Corresp.), vol. EC-16, p. 372, June 1967.
[148] E. S. Spiegelthal, "A content addressable distributed logic memory with applications to information retrieval," *Proc. IEEE* (Corresp.), vol. 52, p. 74, Jan. 1964.
[149] N. J. Stillman, C. R. Defiore, and P. B. Berra, "Associative processing of line drawings," in *1971 Spring Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 557–562.
[150] H. S. Stone, "Associative processing for general purpose computers through the use of modified memories," in *1968 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 949–955.
[151] ——, "A logic-in-memory computer," *IEEE Trans. Comput.* (Short Notes), vol. C-19, pp. 73–78, Jan. 1970.
[152] K. J. Thurber, "An associative processor for air traffic control," in *1971 Spring Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 49–59.
[153] K. J. Thurber and R. O. Berge, "Applications of associative processors," *Comput. Des.*, vol. 10, pp. 103–110, Nov. 1971.
[154] K. J. Thurber and P. C. Patton, "Hardware floating point arithmetic on an associative processor," in *IEEE Int. Computer Society Conf. Proc.*, pp. 275–278, Sept. 1972.
[155] G. T. Tuttle, "How to quiz a whole memory at once," *Electronics*, vol. 36, pp. 43–46, Nov. 15, 1963.
[156] L. D. Wald, "An associative memory using large-scale integration," in *Proc. Nat. Aerospace Electronics Conf.*, pp. 277–281, May 1970.
[157] ——, "MOS associative memories," *Electron. Eng.*, vol. 29, pp. 54–56, Aug. 1970.
[158] C. P. Wang, "A coupled magnetic film device for associative memories," *J. Appl. Phys.*, vol. 39, pp. 1220–1221, Feb. 1, 1968.
[159] A. Weinberger, "The hybrid associative memory concept," *Comput. Des.*, pp. 77–85, Jan. 1971.
[160] H. Weinstein, "Proposals for ordered sequential detection of simultaneous multiple responses," *IEEE Trans. Electron. Comput.* (Corresp.), vol. EC-12, pp. 564–567, Oct. 1963.
[161] M. A. Wesley, "Associative parallel processing for the fast Fourier transform," *IEEE Trans. Audio Electroacoust.*, vol. AU-17, pp. 162–165, June 1969.
[162] M. A. Wesley, S. K. Chang, and J. H. Mommens, "A design for an auxiliary associative parallel processor," in *1972 Fall Joint Computer Conf., AFIPS Conf. Proc.* Montvale, N. J.: AFIPS Press, pp. 461–472.
[163] A. Wolinsky, "A simple proof of Lewin's ordered-retrieval theorem for associative memories," *Comm. Ass. Comput. Mach.*, vol. 11, pp. 488–490, July 1968.
[164] ——, "Principles and applications of associative memories," presented at 3rd Annu. Symp. on the Interface of Computer Science and Statistics, Los Angeles, Calif, Jan. 1969.
[165] ——, "Unified interval classification and unified 3-classification for associative memories," *IEEE Trans. Comput.*, vol. C-18, pp. 899–911, Oct. 1969.
[166] C. C. Yang and S. S. Yau, "A cutpoint cellular associative memory," *IEEE Trans. Electron. Comput.*, vol. EC-15, pp. 522–528, Aug. 1966.
[167] C. C. Yang and J. T. Tou, "A cryogenic associative memory," *J. Franklin Inst.*, vol. 284, pp. 109–121, Aug. 1967.
[168] S. S. Yau and C. C. Yang, "A cryogenic associative memory system for information retrieval," in *Nat. Electronics Conf. Proc.*, pp. 764–769, Oct. 1966.
[169] ——, "A nonbulk addition technique for associative processors," *IEEE Trans. Electron. Comput.* (Short Notes), vol. EC-15, pp. 938–941, Dec. 1966.
[170] ——, "Pattern recognition by using an associative memory," *IEEE Trans. Electron. Comput.* (Short Notes), vol. EC-15, pp. 944–947, Dec. 1966.
[171] E. L. Younker, C. H. Heckler, D. P. Masher, and J. M. Yarbourough, "Design of an experimental multiple instantaneous response file," in *1964 Spring Joint Computer Conf., AFIPS Conf. Proc.* Washington, D. C.: Thompson, pp. 515–528.