W. F. Beausoleil
D. T. Brown
B. E. Phelps

Magnetic Bubble Memory Organization

Abstract: Magnetic bubbles offer promise of data storage at a lower cost than is possible using semiconductor technologies. However, a memory using magnetic bubbles is most economically implemented in the form of long shift registers, rather than as a random access storage device. This plus a lower shift rate than is possible with semiconductors suggests relatively long average access times. This shortcoming is largely overcome by an organizational technique called dynamic ordering, which can reduce the average number of shifts needed to access data. A dynamically ordered magnetic bubble memory, when combined with one or more smaller semiconductor memories in a hierarchy, should provide both large capacity and good performance with reasonable economy.

Introduction

Magnetic bubble technology offers considerable promise for large, low-cost, medium-speed, nonvolatile memories [1–3]. But presently at least they are not suitable for the random access type of organization and must be configured as shift registers. Since the number of shift registers per bubble device (chip) is limited, and since sense amplifiers are relatively expensive, the larger the shift register, the lower the cost per bit. Unfortunately, a shift register with a single sensing station and with randomly stored data requires, on the average, one half as many shift cycles to access a given data bit as the total bit capacity of the register (e.g., a 16,000-bit register averages 8,000 shifts per access).

This paper describes dynamically ordered magnetic bubble shift register memories [4], which keep the data likely to be referred to next near the sensing station, thus reducing the average number of access shifts. Also presented are two improvements to dynamic ordering: double dynamic ordering and two-dimensional ordering. While applicable to any shift register memory capable of shifting in both directions, these techniques are of particular interest for bubble memories in which the economics dictate long shift registers.

After a brief review of the elements of a bubble register, we give some simple illustrations of dynamic ordering, double dynamic ordering and two-dimensional ordering.

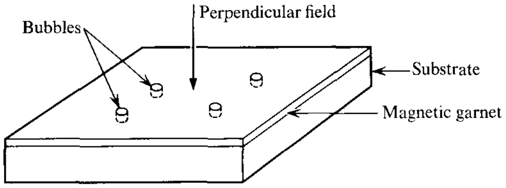


Figure 1 Elements of a magnetic bubble chip.

Magnetic bubble shift registers

In magnetic bubble shift registers [1], the magnetic bubbles are stored in a very thin single-crystal magnetic material such as synthetic garnet. The magnetic material is supported on a heavier nonmagnetic substance and exposed to a steady magnetic field perpendicular to the plane of the magnetic material, as shown in Fig. 1. The size of the bubbles is controlled by the strength of the perpendicular field and by the thickness of the layer of magnetic material. A practical bubble diameter might be less than three micrometers permitting more than one million bubbles per square inch. Bubbles created in the magnetic substance remain as long as the perpendicular field remains constant; thus, the use of a suitable permanent magnet makes the memory nonvolatile. Bubbles do not move unless acted upon by other magnetic fields, but they can be moved by controlled magnetic fields in the

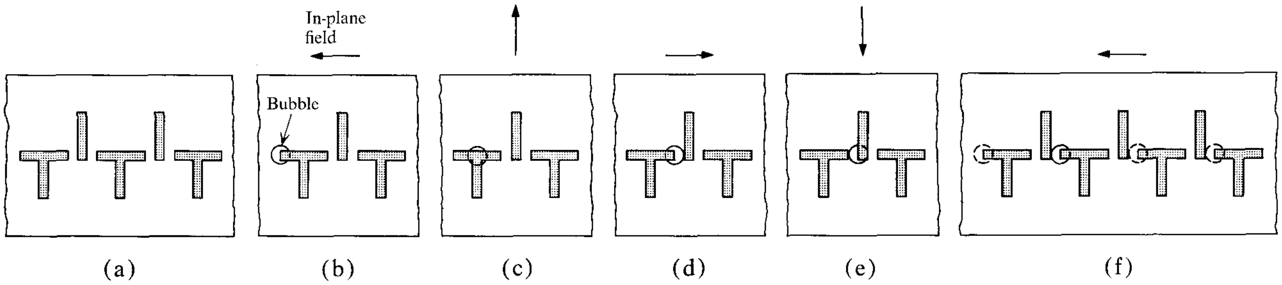
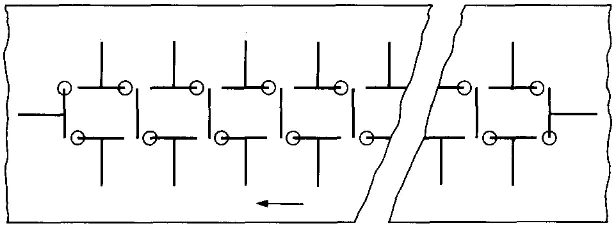


Figure 2 Typical T- and I-bar configuration.

Figure 3 T and I bars combined in a shift register.



plane of the magnetic material. Techniques exist for creating and erasing bubbles when required.

One technique suitable for constructing a bubble shift register uses T and I bars on top of the magnetic material (garnet), as shown in Fig. 2(a). The T and I bars are thin deposited layers of permalloy, normally unmagnetized. If a second, weak magnetic field is applied in the plane of the garnet, the permalloy bars become temporarily magnetized. Any bubble in the vicinity of a bar lying in the direction of this second field is attracted to one end of the bar, which end depending on the directions of the two fields (perpendicular and in-plane).

Figure 2(b) shows the convention for representing such an in-plane field by an arrow at the right and also the location assumed by a bubble on or near a permalloy bar. Figures 2(c),(d),(e), and (f) show the bubble movement as the in-plane field is changed. Note that the in-plane field is effectively rotating clockwise and that the bubble is moved from left to right. The field may actually rotate, or it may be stepped, for example, in 90-degree increments. Shift rates of one million per second have been demonstrated [1].

The in-plane field may be stopped at any position, and the bubbles will not move until a different in-plane field is applied. If the field is either rotated or stepped in the opposite direction, the bubble is moved from right to left. In a working shift register, there may be one bubble for each combination of T and I bar [shown dashed in Fig. 2(f)]. Conventionally, a 1 bit is represented by the presence of a bubble; a 0 bit by the absence of a bubble. Figure 3 shows the combination of such T and I bars into a complete shift register. It is further possible to use control lines and other techniques to cause bubbles to change paths, as from one shift register to another, and, in short, to duplicate the simplified patterns shown in the following illustrations of ordering techniques.

Dynamic ordering

A memory consisting of many long shift registers operating in unison would have, in common with rotating mechanical storage devices, a relatively long storage access time if the data contained therein were assigned permanent addresses. Magnetic bubble shift registers differ from a rotating mechanical device in that shifting (which corresponds to rotation) can be stopped and started almost instantaneously, and, as has been noted, that the shifting direction of a bubble shift register controlled by a rotating magnetic field can be reversed by changing the direction of field rotation. This bidirectional shifting ability is exploited in the dynamic ordering technique.

It is well known that the addresses referred to during program execution are not random, but that each address has a greater likelihood of being in the same locality as those recently referred to [5]. This fact makes feasible memory hierarchies, such as those in the IBM System/360 Model 85 [6] and IBM System/370 Models 155, 165, and 195. In these systems, a small, fast cache or buffer contains recently used data. The processor refers to the cache; if the item requested is not present, a small block of data that includes the item is brought in from a slower main memory. The block replaces a block already in the cache, which is typically chosen to be one not recently used. Due to the locality-of-reference principle, the fraction of items not found in the cache is small, and good performance is achieved.

A dynamically ordered shift register memory also relies on the locality-of-reference principle to achieve good performance. As noted, the bubble memory consists of many registers shifting in unison. A page of data is comprised of one bit from the corresponding position in each data shift register. In addition to the data shift registers, identical registers contain the address of each page. (Bytes within a page have sequential addresses, and the page size is a power of two, so that all byte addresses in a page have common high order bits that define a unique page address.) The addresses shift with the pages, making every page self-labeled and removing the requirement of any predefined page order. By decoupling the page currently referred to from the shift registers and reverse-shifting, the pages are continually reordered. Recently referred to pages stay near the I/O position (i.e., the position from which a page can be read or written into). The following simple example describes the reordering process.

Figure 4 shows symbolically a collection of shift registers, all of which shift in unison. Three registers contain the address bits; the others contain the data bits. There are 8 positions in each shift register, so this illustrative memory unit contains a total of 8 pages. The bottom positions of the registers can be logically separated from the rest of the registers and serve as the I/O positions. To simplify the illustration of the ordering process, Fig. 5 shows only a single shift register.

In Fig. 5(a) the pages in the shift register are numbered to indicate the order of use at the time this example begins. The highest number (8) indicates the most recently used page. In Fig. 5(b) a new page is requested from the memory and the register contents are shifted down until the desired address is located in page 5. This requires a down shift of 3 positions. While the system is reading or writing data from page 5, the I/O position (see Fig. 5(c)) is decoupled from the rest of the register and the rest of the register contents are shifted up as many positions as they had been shifted down (i.e., 3). Page 5 is now in the position of the most recently used page, with the rest of the register ordered to correspond.

Now, in Fig. 5(d), page 7 is requested. Shifting the register down two steps places page 7 in the I/O section. Shifting up two positions as in Fig. 5(e), with the I/O position decoupled, reflects the latest change in use. As shown in Fig. 5(f) the most recently used page (7) now rests in the I/O position; the next most recently used page (5) is only one shift cycle away, and so on. Page 7 is available without any lost time for shifting, page 5 is available with just one shift, etc.

Performance can be further enhanced by a modification to dynamic ordering called double ordering. Figure 6 shows a double-ordered shift register. Even-numbered and odd-numbered pages are ordered in opposite directions, so that recently used even-numbered pages are near the bottom and odd-numbered pages near the top. Less recently used pages are nearer to the center. Note the arrows in Fig. 6, which indicate that shifting can occur in both directions on both shift paths. Even-numbered pages are accessed by shifting down and odd-

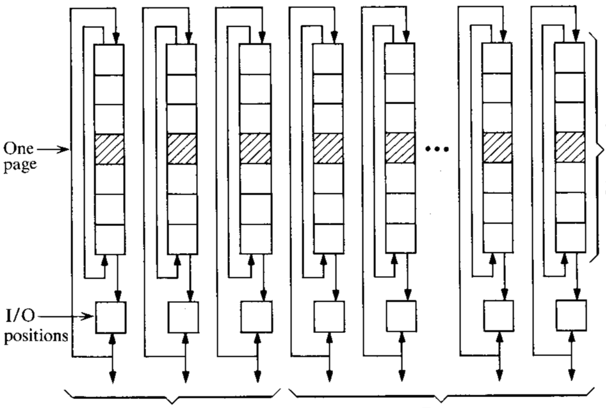


Figure 4 Shift register memory.

Figure 5 Shifting process during search and reordering.

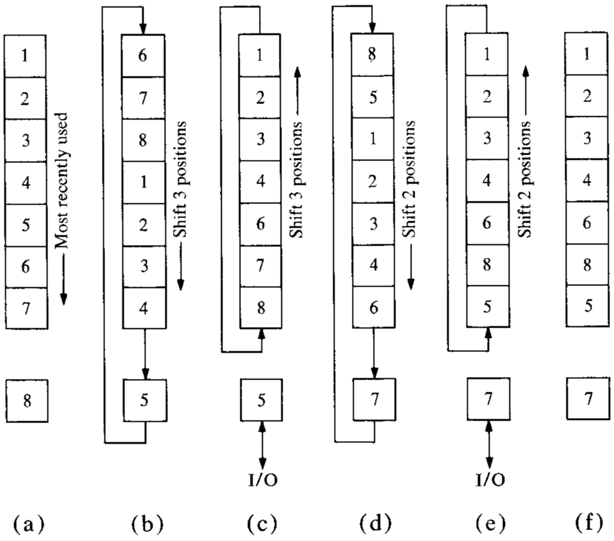
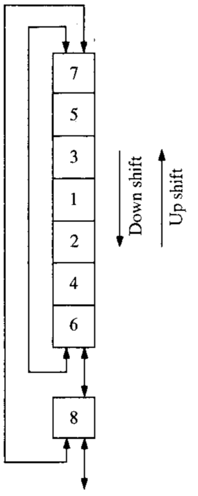


Figure 6 Double ordering scheme.



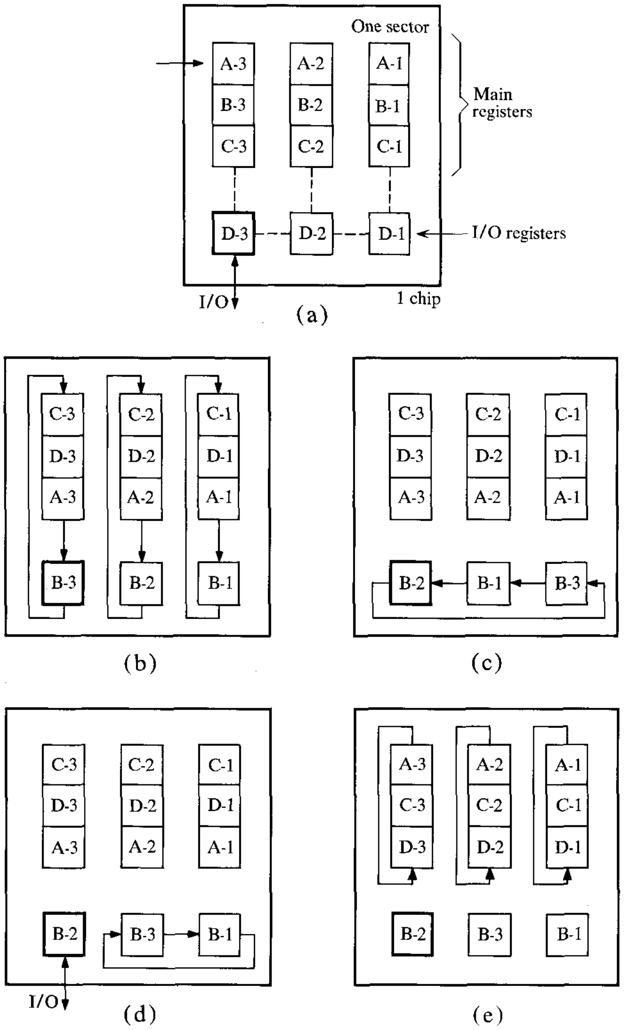


Figure 7 Two-dimensional ordering.

numbered pages by shifting up. In either case, the outside shift path is used for access. In the example in Fig. 6, pages 6 and 7 can be accessed in one shift, pages 4 and 5 in two shifts, etc. After a page has been accessed, the registers are reordered by shifting pages via the inside path in the direction opposite to that of the access shifts. This shifting continues until the page at the top of the main registers is odd-numbered and the page at the bottom is even-numbered, The number of reordering shifts can be determined as follows. If the accessed page and the page previously in the I/O position are either both odd or both even, the number of reordering shifts is equal to the number of access shifts. However, if the accessed page and the page previously in the I/O position are an odd-even or an even-odd combination, the number of reordering shifts is one fewer than the number of access shifts.

Double ordering has another advantage that is not too obvious in our simple illustration; it effectively halves worst-case access time (to the least recently used page). In our Fig. 5(a), it would take 7 down shifts to place page 1 in the I/O register, whereas in Fig. 6 double ordering makes page 1 available in 4 shifts. In a full-sized memory with hundreds or thousands of positions per register, a 50-percent reduction can be significant, particularly considering that limited additional controls are required.

Two-dimensional shift register ordering can further reduce the worst-case access time, as well as improve the average access time. Figure 7(a) is a very simple illustration of such an organization. This is effectively our 8-position shift register enlarged to 12 positions and divided into three 3-position shift registers plus a 3-position I/O shift register. Here we need to use another memory term, *sector.* A sector is a block or group of pages with contiguous page addresses; in our example, a sector is 3 pages in a given position in the 3 registers or in the I/O register.

In two-dimensional ordering, sectors are ordered according to use and pages within sectors are also ordered according to use. In our example, sectors have been given alphabetic designations and pages given numeric designations (D represents the most recently used sector, 3 the most recently used page within each sector at the time the example starts). Single ordering is shown for simplicity, but double ordering can be used both for sectors and for pages within sectors.

The shift registers are interconnected and controlled in such fashion that we have four possible modes of operation. In Fig. 7(b) sectors can be shifted into and out from the I/O register. In Fig. 7(c) the sector in the I/O register can be shifted to access a desired page from that sector. In Fig. 7(d) the remainder of the contents of the I/O register are shifted to enable re-ordering of the pages within that sector. In Fig. 7(e) the contents of the main registers are shifted to enable reordering of the sectors.

For illustration, we have assumed that page B-2 was requested [Fig. 7(a)]. Two shifts down place its sector in the I/O register [Fig. 7(b)]. One left shift of the I/O register places it in the I/O position [Fig. 7(c)]. One right shift of the I/O register reorders the pages within that sector [Fig. 7(d)]. Two upshifts of the main registers puts the sectors in proper order [Fig. 7(e)]. Now the most recently used sector is in the I/O register with the most recently used page in the I/O position.

In a two-dimensional organization, the worst-case access time can be reduced significantly. For example, if the number of pages equals the number of sectors, the maximum number of shifts is only $2\sqrt{n}$, where n is the maximum number of shifts that would be required in an equal-capacity, one-dimensional organization with the same page size.

Conclusions

Dynamic ordering, double dynamic ordering, and two-dimensional ordering are organization techniques that use the bidirectional shift capability of magnetic bubble shift registers to overcome the performance disadvantage of a memory constructed of shift registers. Further work is underway in order to determine the magnitude of the improvements possible.

Acknowledgments

Some of the concepts discussed here were developed jointly by W. A. Clark, IV, and the authors. E. L. Walker and R. Levi participated in the dynamically ordered magnetic bubble shift register design.

References

1. A. H. Bobeck and H. E. D. Scovil, "Magnetic Bubbles," *Scientific American*, **224,** No. 6, 78–90 (1971).
2. A. H. Bobeck, R. F. Fischer, and A. J. Perneski, "Application of Orthoferrites to Domain-Wall Devices," *IEEE Trans. Magnetics*, **5,** No. 3, 544–553 (Sept. 1969).
3. A. H. Bobeck, R. F. Fischer, and J. L. Smith, "An Overview of Magnetic Bubble Domains—Material-Device Interface," *Proc. Conf. on Magnetism and Magnetic Materials*, Amer. Inst. of Physics, Chicago (Nov. 1971).
4. Patent number 3670313 filed March 22, 1971, and issued June 13, 1972, to W. F. Beausoleil, D. T. Brown, and E. L. Walker.
5. M. V. Wilkes, "Slave Memories and Dynamic Storage Allocation," *IEEE Trans. on Electronic Computers*, **EC-14,** No. 2, 270–271 (1965).
6. C. J. Conti, "Concepts for Buffer Storage," *IEEE Computer Group News* **2,** No. 8, 9–13 (1969).

Received April 26, 1972

W. F. Beausoleil is in the IBM System Product Division, Poughkeepsie, New York, and D. T. Brown and B. E. Phelps are in the IBM System Development Division, Poughkeepsie, New York 12602.