EC83－48

バッファループ付加型磁気バブルメモリを用いたGRACEメモリモジュール

Memory Module of Relational Algebra Machine GRACE
Employing Efficient Access Mechanism for Magnetic Bubble Devices with
On-chip Memory Hierarchy

喜連川 優○　荻野 正□　田中 英彦□　元岡 達□

Masaru Kitsuregawa　Tadashi Ogino　Hidehiko Tanaka　Tohru Moto-oka

○東京大学 生産技術研究所　□東京大学 工学部

Institute of Industrial Science
University of Tokyo

Faculty of Engineering
University of Tokyo

1. Introduction

1.1 Control mechanism for magnetic bubble memory

Magnetic bubble memory has received attention for its high capacity and nonvolatility. It always attains 4 or more times larger capacity than semiconductor memory. On the other hand, the relatively low transfer rate makes the access time deteriorate as its capacity increases. This is because the magnetic bubble memory is basically a sequential memory and its access time becomes longer for larger minor loop. Whereas the chip fabricated in Bell Laboratory in 1973 has minor loops of 293 bits, the current 1 Mbit chip has those of 4096 bits long. The superiority to the disk in access time is being lost. There exists a tradeoff between large capacity and fast access time. Commercial bubble memory systems produced by such companies as Intel and Fujitsu adopt a conventional addressing mechanism and do not necessarily take any measures for this problem. The bubble controller by Hitachi[1] slightly modifies the addressing control mechanism. It uses the skipped address assignment so as to maximize the transfer rate in a multiple page operation.

On the other hand, there are several proposals to realize higher functions such as dynamic reordering on bubble ladder[2] and rebound sorting[3]. While these present various interesting features, many switches are required and its implementation has not been addressed. To improve the performance, we can find another approach easier to implement, memory hierarchy on chip level, which exploits the locality of references. Access time can be easily altered by arranging the length of the minor loop. The bubble chip with two level hierarchy was proposed[4]. Two kinds of minor loops, short and long one connected by common major line, are incorporated on a single chip. In [5], single bit buffering circuit design and dynamic reordering control are proposed, where short loop of two level hierarchy is reduced to be one bit long. Later another type of chip, quite similar to that by Anacker, appeared[6][7][8], where minor loop is segmented into two loops with a switch, which also shows the same effects. Recently this type of chip has been actually fabricated[9]. But its control mechanism is rather conservative. Records in a short loop is regarded as a segment and data exchange is performed at segment level. The segment has physically sequential order, where a segment in short loop must be swapped out to fixed position in long loop and thereafter requested segment is swapped in. There is no reordering even at segment level. If the records frequently referenced are spread beyond the segment boundary, we cannot expect high performance. The net effect is the expansion of the unit of swap, compared with the one record buffering assembly very early proposed by P.I.Bonyhard[5].

In this paper presented are the efficient control mechanism for the bubble memory system with on-chip hierarchy and the organization of the pilot system with sixteen 1 Mbit bubble chips. In our system, swap control at record level is realized and the frequently referenced records spread over large storage space are automatically converged into the buffer space. The logical record address is not fixed to the physical address but their mapping is dynamically managed. Bubble memory system incorporating these mechanisms are expected to

exhibit high performance. The potentials of this chip could be exploited.

1.2 Memory module of GRACE

We have developed the relational algebra machine GRACE[11,12]. GRACE adopts the novel relational algebra processing algorithm based on hash and sort, and can execute join quickly in O((N+M)/K) time, where N and M are the cardinalities of two relations and K is the number of memory banks to be involved. In GRACE the relations are partitioned dynamically into several buckets using hash function and the buckets are processed in parallel by multiple processors. She is expected to exhibit high performance even under the join dominant environment.

GRACE consists of four kinds of fundamental modules: processing module(PM), memory module(MM), disk module(DM) and control module(CM). These modules are connected with two inter-module networks. The processing modules and memory modules are connected by a processing network while memory modules and disk modules are connected by a staging network. The relations stored in disk modules are staged into memory modules through the staging network and then the data streams generated by memory modules flow into processing modules through the processing network.

The role of the memory module is to generate the bucket-serial data stream and also to provide the large intermediate storage space in which clustered relations are produced. Magnetic bubble memory and semiconductor RAM are the candidates of the storage medium. At present the advanced semiconductor technology can provide relatively large random access space at reasonable cost and seems to surpass the bubble technology. Contiguous disk bubble memory, however, is expected to attain the capacity of 4 ~ 16 Mbit per chip. The Bloch line memory seems to be promising because of its very large capacity (1 Gbits per chip) and much higher data transfer rate. Since these storage domain rotating devices are considered to employ Major/minor organization and the organization with RAM is relatively straightforward, the pilot module is implemented with bubble memory. The bubble memory system plays a key role as the intermediate storage of GRACE. Instead of the ordinary bubble chip, the improved one with on-chip hierarchy as described above is employed. The bubble control unit controls this device so that continuous data stream could be formed.

All the accesses to the bubble are performed based on the hashed values of the tuples. The tuples are not accessed by their physical locations. Instead we provide record descriptor memory(RDM) constructed by RAM which contains hashed values of the tuples and memory management tags. Synchronized with the magnetic field rotation, this memory is used to access the bubble associatively. On staging the arbitrary empty block denoted by the entry of RDM is dynamically allocated. For data stream generation, the tuples with the same hashed value which belong to a bucket are taken into the buffer loops dynamically to produce the continuous bucket-serial data stream with little inter-record gap time. Look ahead buffering control is employed so that the tuples of next bucket is brought up in buffer loops while those of the current bucket is being output. The number of tuples of each bucket is maintained and their output sequence is scheduled.

The design details of the controller for the improved bubble chip with buffer loops are described in this paper.

2. Improved Bubble Chip Organization and Its Control Mechanism

Here we examine, from the different points of view, the performance improvement by incorporating the two kinds of loops, short and long loops, onto the bubble chip.

2.1 On-chip Memory Hierarchy

Introducing the memory hierarchy on the chip is a effective approach to improve the performance of the bubble memory characterized by its large capacity and low access time. For the magnetic bubble memory, the hierarchical structure can be easily involved in a chip. Partitioning the minor loop into two loops of different length L1 and L2(L1<L2) through switch implements the two level hierarchy:short loop with the average access time L1/2 unit time and the capacity L1 records and long loop with (L1 + L2)/2 access time and capacity of L2 records. Multiple level hierarchy can be also designed in the same way. Fig.1 shows the chip organization fabricated by NEC[9]. We call the short loop buffer loop and the long loop main loop. Buffer loop is connected to major line through

block replication/transfer(BR/T) gate. The buffer loop and main loop is segmented by swap gate, through which the records in two loops can be exchanged. The requested record in main loop is swapped into the buffer loop and at the same time the counterpart record in buffer loop is swapped out into the main loop. It is clear that mapping between the logical record number and physical record address needs to be maintained dynamically as is found in virtual storage or cash memory management. With the adequate control mechanism, frequently referenced records over main loop automatically gather into the buffer loop, which results in much higher access time[7].

2.2 Minimization of Inter-record Gap Time for Set Oriented Access

For the applications such as data base processing, instead of individual record access by an address, set-oriented access for the records which satisfy some condition is performed frequently, where the output sequence of the records often does not matter. It is not the access time of a single record but the total output time of a set of records that is a key to the high performance. Namely the effective transfer rate is much more influential. Although unnecessary record can be skipped very quickly compared with the disk, namely during one bit field rotation period, inter-record gap time becomes dominant for conventional M/m bubble chip as the length of the minor loop becomes long for large capacity. It is difficult to output the records spread over long minor loop efficiently, i.e. continuously. Some mechanism to reduce the inter-record gap time is required to improve the performance. The bubble chip organization described above also works effectively. The short loop inserted between main loop and major line provides the record buffering mechanism to increase the effective transfer rate. Necessary records can be buffered in the added short loop while a record is being output bit serially. Exploiting the high bandwidth between buffer loop and main loop, the buffer loop is filled with the concerned records, which results in the continuous record output almost without the inter-record gap time. Performance of buffered bubble chip for set oriented access was examined in[12]. This tuple buffering also requires the dynamic address mapping mechanism as same as that of the memory hierarchical control. Using this technique, the hash-based relational algebra machine GRACE realizes the efficient set oriented access for the records with the given hash code of the record key. The details are beyond the scope of this paper[11,12].

3. Record Representation and Continuous Read/Write Operation

3.1 Record representation

Before discussing the details about the addressing mechanism, the record representation on bubble chips needs to be described here. This also contributes to the gap-less read/write operations. Fig.1 shows the structure of the

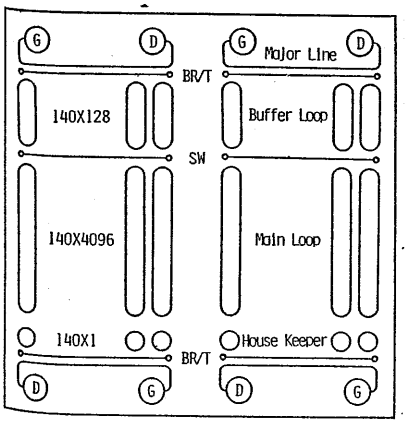

Fig.1 Chip Organization of Magnetic Bubble Memory With Two Level Hierarchy

Short loop(buffer loop) and long loop(main loop) are connected through swap gate

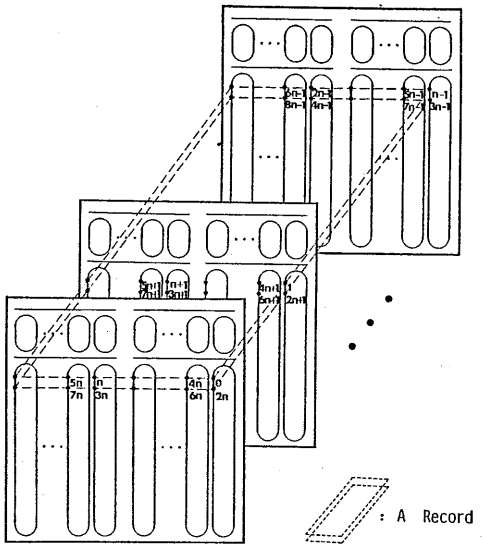

Figure 2 Record Representation on Multiple Bubble Chips

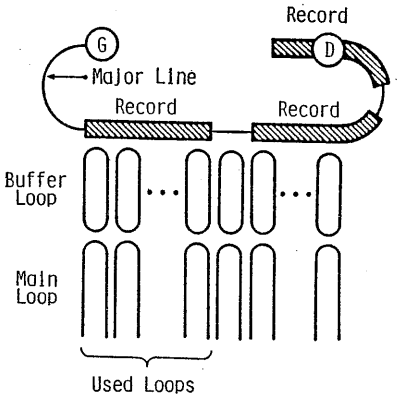

Fig.3 Continuous Read Operation

bubble chip with two level hierarchy. A chip contains two subchips. These are activated in parallel to attain higher transfer rate. Here we consider the memory system composed by multiple chips and representation scheme of a record on chips. A record is represented by the two adjacent rows of each subchip. When N chips are to be used in parallel, bits of a record are distributed over 4N rows of all the subchips. Fig.2 shows the bit assignment of a record. The handling for defect loops is described later in section 5. The length of a record is determined by the number of effective loops. Here we call these effective bits in 4N rows a physical record. We discriminate a logical record from a physical one. The former is the actual record that a user is to store.

3.2 Continuous read/write operation

Conventional controller cannot handle the continuous read/write operation efficiently. A controller serially treats the access for an physical record. After the current record is detected completely, the subsequent record is controlled to be transferred onto the major line. Therefore the inter-record gap time from the leftmost loop to the detector is at least produced. The length of a logical record is not always the same as that of a physical record. When the logical record is short, the last portion of the physical record is usually packed with zero. This increase the inter-record gap time and makes the effective transfer rate even worse.

This problem can be solved by introducing the adequate representation scheme of a record and a flexible control mechanism. As shown in Fig.3, a next record can be transferred onto the major line when the present record passed away the portion of major line over the used loops. This achieves contiguous arrangement on the major line and results in gapless record output. Our controller supports this contiguous transfer mechanism and realizes the higher effective data transfer rate. For continuous write operation, similar transfer mechanism is used. The timing phases for block replication/transfer and detection/generation are slightly shifted.

To realize the above control scheme, records are stored on chips adjusted to be at the side of the generator as shown in Fig.3. By this record representation scheme, records can be written continuously.

4. Dynamic Address Mapping Mechanism and Record Descriptor Memory

4.1 Record Descriptor Memory

Record Descriptor Memory attached with magnetic bubble memory has the entry for each record. It stores the description of a record and is synchronized with the movement of the bubbles. For the chips described above, two RDMs for buffer loop and main loop are used. When the records are swapped between these two loops, the descriptions for these records are also swapped. All the access to the bubble chip is performed based on the contents of RDM. The bubble has no fixed address physically. We can store several kinds of information related to each record in RDM to work this two-loop bubble chip efficiently.

As described in section 2.1, to exploit the benefits of hierarchical feature of the chip, mapping mechanism from logical record address to its physical location is required. The logical address of a record is stored in RDM. An address is allocated to an arbitrary location on-the-fly. The related records can be clustered into the shorter loop dynamically using swap gates. RDM shows which records are stored in the buffer loop. Due to the fine address mapping mechanism between two loops at record level, a larger performance improvement can be expected exploiting the benefits of on-chip hierarchy. The transfer gate between the major line and buffer loop is opened, when the requested record whose entry in RDM matches the given address is under the gate.

In data base application, as explained in 2.2, several kinds of information such as keys and hash codes of a record would be stored in RDM. The associative access based on such partial information stored in the RDM can be realized. RDM is also used as the working store for temporary results of data base processing. In relational algebra machine GRACE, the relation is hashed into several buckets dynamically and the hash code of the attributes which participates in join or projection operation is set into the RDM. The records with the same code are output continuously. This bucket-serial data stream generation is key function to GRACE. The records cannot be clustered on

the hash code along with the loading of the relation into the bubble memory, since the physical location is assigned on-the-fly following the data stream being loaded. The records having the identical hash value are dynamically clustered to generate bucket-serial data stream by buffering the related records into the buffer loop under the adequate control of gates using RDM.

Thus RDM could be utilized in several way. Other examples can be found in [10], where the binary data structure is represented. It is not the objective of this paper to present the various usage examples of RDM for several applications. The following sections do not discuss such problems. We investigate the control mechanisms for the gates, detector, and generator of the two-loop bubble device on the read/write operation using this record descriptor memory. RDM is fully utilized by the controller. Read and write operation is efficiently performed by this RDM-based control mechanisms and the potentials of the improved chip with on-chip memory hierarchy can be exploited effectively.

4.2 Read/Write Operation Based on Dynamic Address Mapping Mechanism

Read operation, described in section 2, is performed by examining the contents of RDM. The necessary records in main loops are dynamically swapped into the buffer loop, where the gates are controlled using the RDM. The physical locations of logical records are maintained solely in this memory.

For write operation, neither used is the conventional fixed addressing mechanism, since there is no physically fixed address. When a record is to be written, an arbitrary empty physical record which is denoted in RDM is dynamically allocated and the specified logical address is stored in the corresponding entry of RDM. Since any place can be assigned to the address only if it is empty, the nearest empty area is used for the new record. Thus for write operation as well as read operation, we can expect higher performance. Control mechanism to locate empty area on bubble generation, however, is not so simple for the improved bubble chip with buffer loops, which is discussed in the following section.

For data base processing, rapid back-out mechanism is essential. Several checks such as integrity control is performed to ensure the validity of the storage operation. If the operation is shown to be incorrect, the data base have to be restored to the previous consistent state. RDM is used as a tag for memory management. The updated records are stored in different area from that of the original record. The operation can be rapidly backed out by solely manipulating the RDM. Several tags can be defined on RDM. This is also used to maintain checkpoints.

Thus the RDM is used effectively for memory management.

4.3 Control Mechanism for Gates, Generator, and Detector

4.3.1 Control for Gates and Detector on Read Operation

(1) Control of BR/T gate between major line and buffer loops

The gate is activated when the RDM entry of the record under the gate has an appropriate value and the major line is empty.

The first condition is obvious. For the read operation, the value of the entry which is under the gate is checked every field rotation. The RDM plays like a shift register, and in real is constructed with semiconductor RAM and counters since several points need to be examined. Associated address register which is used to control this transfer is incremented to synchronize the magnetic field. If the record is transferred to the major line, record descriptor needs to be cleared. When the record is replicated, some control field of the entry is modified to denote that it is already output.

The second condition is used not to overwrite the record previously transferred on major line. But we can overwrite the data of buffer loops onto the major line if that data is all 0, namely, no bubble. While the length of the record must be equal to that of physical block of the bubble in the conventional bubble memory, the length can be programmable in our design. The record length counter which has the record length as initial value begins to be decremented when the record is transferred on major line. When it comes to zero, which means that the record has passed the portion of the major line above the currently used loops, next record can be transferred. This control realizes high data transfer rate regardless of the

length of the record.

(2) Control for detection of bubbles

There is some delay between data replication on major line and data detection on the detector, since bubbles have to travel along the major line. If the record length is short and the major line length to the detector is long, number of records could be on the line to improve the effective data transfer rate. The information when the next record arrives at the detector must be kept in the controller. A shift register is used in our design. The head of each record on the line is marked.

Control mechanism for defect loops is described in section 5.

(3) Control for swap gates between buffer loops and main loops

Swap gates, if activated, exchange the record in buffer loops and its counterpart in main loops. The records are swapped when the record in the main loops has higher output priority than that in the buffer loops. A sophisticated control logic which performs multiple comparisons allows the look ahead buffering. That is, the next set of required records with lower priority is swapped into the buffer loop while the current set of records with higher priority is being detected. This also helps the reduction of inter-record gap time. When the records of both loops have the same priority they are not swapped. Thus both of RDMs are read and compared based on their priority. Their contents are exchanged when the records are swapped. Two address pointers are used on this swap control: one for RDM of buffer loop and the other for that of main loop.

4.3.2 Control for Gates and Generator on Write Operation

(1) Control for the generator

Control mechanism for the write operation is much more complicated than that for the read one. We must control the generator so that the corresponding physical record block is to be empty when the generated bubbles arrive at the appropriate point on the major line just above the used buffer loops. The correct anticipation mechanism accounting for the propagation delay on the major line is required. The row in the buffer loops corresponding to the currently generated record is checked whether it is empty or not. If it is empty we can generate the bubbles of a record at that time since the row is guaranteed to be vacant when the record reached the appropriate position on the major line. Here we call this corresponding position the shadow of the generator. There are several shadows besides that on the buffer loops. Fig.4 shows all the shadows on buffer loops and main loops.

(i) Shadow on buffer loops

To sum up, we have one shadow SB on buffer loops. This position from the major line can be calculated by MOD(GL + RL,BL), where GL is the length from the generator to the left most loop, RL is the length of the record and BL is the buffer loop length.

(ii) Shadow on main loops

Even though the shadow on buffer loops is occupied, that record might be swapped into main loops and the shadow may become empty when it comes under the major line. It is not sufficient to examine the shadow on the buffer loop solely. We introduce the shadows on main loops. Fig.4 also shows these shadows. There are three shadows: SMØ,SM1,SM2. The number of shadows depends on the length of a record.

Let SG be the shadow of a generator, that is, the shadow of a record with 1 bit length. BLØ be the length from swap gate to major line in rotating direction. The length from major line to SG, SGL, can be expressed in

SGL = MOD(GL , BL)

Then the number of shadows changes as follows

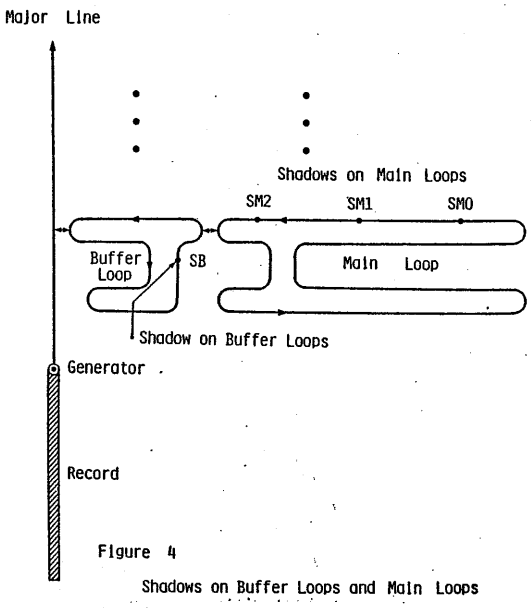

Figure 4
Shadows on Buffer Loops and Main Loops

depending on the length of a record.

i) SGL < BLØ

$$RL < BL\emptyset - SGL \quad \cdots \quad n$$
$$BL\emptyset - SGL \leq RL < BL + BL\emptyset - SGL \quad \cdots \quad n+1$$
$$\vdots$$
$$(k-1)BL + BL\emptyset - SGL \leq RL < kBL + BL\emptyset - SGL \quad \cdots \quad n+k$$

ii) SGL ≥ BLØ

$$RL < BL - (SGL-BL\emptyset) \quad \cdots \quad n+1$$
$$BL - (SGL-BL\emptyset) \leq RL < 2BL - (SGL-BL\emptyset) \quad \cdots \quad n+2$$
$$\vdots$$
$$(k-1)BL - (SGL-BL\emptyset) \leq RL < kBL - (SGL-BL\emptyset) \quad \cdots \quad n+k$$

where $n = [GL/BL]$

As the length of the record becomes longer, the short buffer loops rotate multiple times. This generates several shadows on main loops. Here we assume that the main loop be long enough for shadows to be overlapped each other.

In our chip GL is set equal to BLØ. BL is 128 and BLØ is 4Ø for NEC's chip, which means that the buffer loop is folded and the gates are positioned asymmetrically. The number of shadows depends on the length of a record.

$$RL < BL \quad --- \quad 1 \text{ shadow}$$
$$BL \leq RL < 2BL \quad --- \quad 2 \text{ shadows}$$
$$2BL \leq RL < RLmax \quad --- \quad 3 \text{ shadows}$$
$$(RLmax < 3BL) \quad .... \quad \{1\}$$

For the record whose length is less than 128, SB passes through the swap gate only once. Therefore one shadow SMØ is solely generated. If the length is in the range from 128 to 256 bits, SB passes through the gate twice. In this case two shadows are generated. When it is longer than 256, SB passes through the gate three times. Then SM2 is introduced. Remark that the record here is not a logical one but a physical one. That is, the record is expanded because the defect loops must be skipped. The record whose logical record length is less than 256 might become longer due to defect loops.

Even though the shadow on the buffer loop is not vacant, the record could be absorbed into the shadows on the main loop. Therefore the anticipated control must be given.

(iii) Shadow on major line

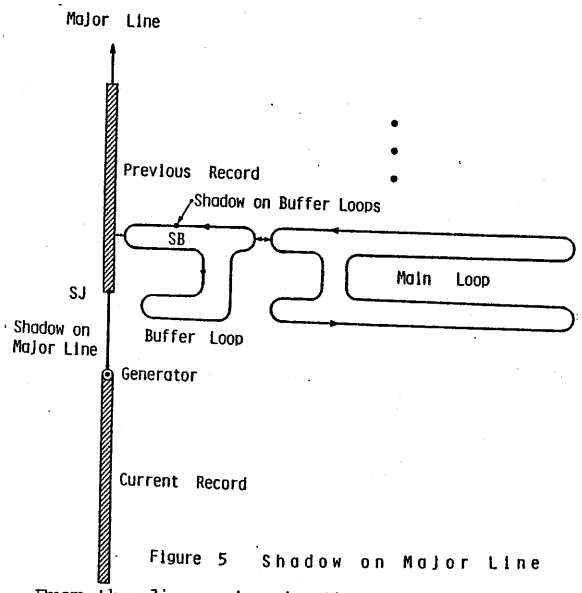

Figure 5 Shadow on Major Line

From the discussion in (i), we generate bubbles only if SB is empty. There is a certain case, however, in which this rule does not hold. Fig.5 shows this situation. The record on the major line which has been already produced might fall into the shadow. Therefore we have to consider the shadow on major line(SJ) as well as the other shadows. If SB is vacant but SJ is full, bubbles may not be generated. The shadows on major line, SJLi, are the points as follows.

$$SJLi = MOD(GL + RL, BL) + iBL$$
$$SJLi < GL, \; i > \emptyset$$

SJLi denotes length from the leftmost transfer gate in direction to the generator. If GL is long, several shadows are generated on major line. In our chip, however, at most only one SJ could be produced since GL < BL. When the record length RL is in the following range, we can find a shadow.

$$BL - GL \leq RL < BL$$
$$\cdots\cdots \quad \{2\}$$
$$2BL - GL \leq RL < 2BL$$

As described above there are three kinds of shadows, by which generation of bubbles is controlled. The generator is activated only if the appropriate empty shadow can be acquired. The record will be written into the shadow on buffer loop after RL + GL unit time. It is necessary to estimate the state of shadow. If that shadow is to be empty after RL + GL unit time, bubbles can be generated. We can find several cases.

a) SB is empty at present

a-1) SJ is set and SB will be occupied by it after RL + GL time

a-2) SJ is set but SJ will pass through SB into SM and SB will be empty

a-3) SJ is empty. Therefore SB is also empty.

b) SB is occupied at present

b-1) SMs are occupied and SB will be still occupied after RL + GL time.

b-2) SJ is empty and a record at SB will be transferred into SM and SB will be empty.

b-3) SJ is set. SB will move to SM but SJ will go into SB. SB will be occupied.

b-4) SJ is set. Both SJ and SB will go into SM.

Thus we can specify the generation condition under a certain design parameter. Cases of b-3) and b-4) cannot occur in our chip design. Among the above cases, generator is activated for a-2), a-3) and b-2) cases. The condition also depends on the length of a record. Here each case is examined in our chip.

a-2) :

From {1} and {2}, there are two cases for major line shadow and one or two main loop shadows are generated for each case. This case of a-2) can be expressed as follows, where SB denotes the logical variable and is true when its place is occupied. Other variables such as SJ, SMØ, SM1, SM2 also have same meaning.

$$\overline{SB} \wedge \{(BL - GL \leq RL < BL) \wedge (SJ \wedge \overline{SM\emptyset}) \vee (2BL-GL \leq RL < 2BL) \wedge (SJ \wedge (\overline{SM\emptyset} \vee \overline{SM1}))\} \quad \ldots\ldots \{3\}$$

a-3) :

This is apparent.

$$\overline{SB} \wedge \overline{SJ} \quad \ldots\ldots \{4\}$$

b-2) :

From {1}, there could be at most three shadows depending on the length of the record. The condition is

$$\overline{SJ} \wedge SB \wedge \{(RL < BL) \wedge \overline{SM\emptyset} \vee (BL \leq RL < 2BL) \wedge (\overline{SM\emptyset} \vee \overline{SM1}) \vee (2BL \leq RL) \wedge (\overline{SM\emptyset} \vee \overline{SM1} \vee \overline{SM2})\} \quad \ldots\ldots \{5\}$$

The generator condition is obtained by summing the above three expressions. Under RL < BL-GL or BL ≤ RL < 2BL - GL or RL ≥ 2BL, the major line shadow is not produced and SJ is false.

The total condition is

$$\{3\} \vee \{4\} \vee \{5\} = \overline{SB} \wedge \overline{SJ} \vee \overline{SM\emptyset} \vee \overline{SM1} \wedge (RL \geq BL) \vee \overline{SM2} \wedge (RL \geq 2BL) \quad \ldots\ldots \{6\}$$

If this is true, bubbles for a entering record starts to be generated. Several memory accesses to RDM for SB, SMØ, SM1 and SM2 are required to confirm that the shadow on buffer loop is empty after GL + RL unit time. These accesses are performed at every field rotation.

(2) Control for BR/T gate

Several records may be on the major line. The delay line is used to catch the heads of records. This is constructed by a shift register with the length GL. When the record arrives at the appropriate position, the gate between major line and buffer loops can be activated unconditionally since the location where we want to insert the record is guaranteed to be empty. The entry of RDM is not changed until actual record insertion is performed.

(3) Control for swap gate

The control mechanism of the swap gates for the write operation is much easier than for the read operation. Gates are activated when the block on main loops is empty so that empty area is prepared for the subsequent entering records. This improves the performance for continuous write operation to minimize the inter-record gap time. The records in buffer loops are controlled to be transferred into main loops. The entry value in RDM is also transferred.

5. Defect loop control

The defect loop control is straightforward for the memory system consisting of only one bubble chip. When multiple chips are activated in parallel to attain higher data transfer rate, their control is rather complex since the positions of defect loops are different from chip to chip. Fig.2 shows the distribution of the bits of the record over multiple chips. Two bits from adjacent rows in storage loops construct the two bits adjacent in the record. Since these two bits are in the same storage loop, both of them are discarded when that loop is wrong. Fig.6 shows the defect loop handling mechanism in write operation, where the memory system consists of two chips. The bit stream is handled in the unit of two bits, as shown in the figure. If there is a defect loop, then the two bits corresponding to that loop are skipped. The data stream is suspended and Ø's are controlled

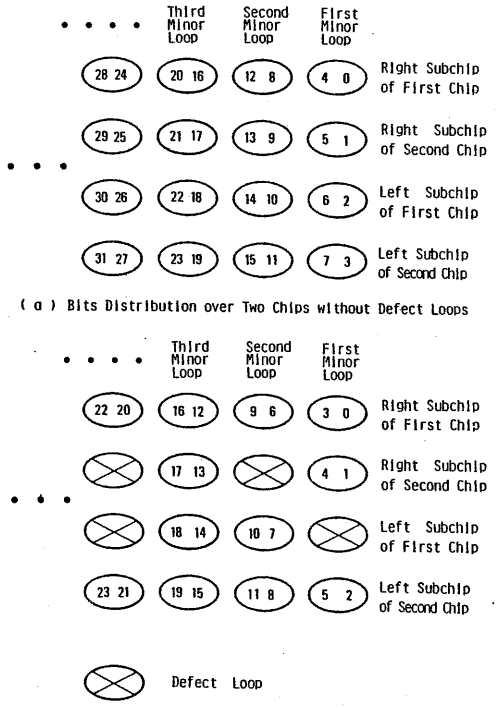

Figure 6 An Example of Bits Distribution over Two Chips with Defect Loops

to be inserted when defect loops are encountered. Thus the bits of a record are distributed over the bubble chips adequately. For the read operation, two bits corresponding to defect loop are discarded and the bit stream is compressed to form the original record. The rate of the data stream varies due to defect loop processing.

The defect loop information is used for the above bits distribution over loops. The bubble chip fabricated by NEC has the house keep loops for the defect control on chip, which are one bit long loops and contain 1 for a correct loop and Ø for a defect loop. Though this organization was intended to simplify the control circuits, they don't work well in our environment. The length of a record is programmable in our design. If the defect loop information for all the loops is stored in house keep loops, overwriting of the bubbles occurs on house keep major line. Hence we use ROM instead of it.

Fig.7 shows the diagram of the defect loop control circuit for read operation. The raw data from the bubble chips are latched into the shift register in parallel, the output of which is serially fed to one byte shift register, depending on the defect loop information in ROM. Only the correct bits are controlled to be supplied to it. Thereafter they are transferred to the output buffer byte-serially. Thus the original record is

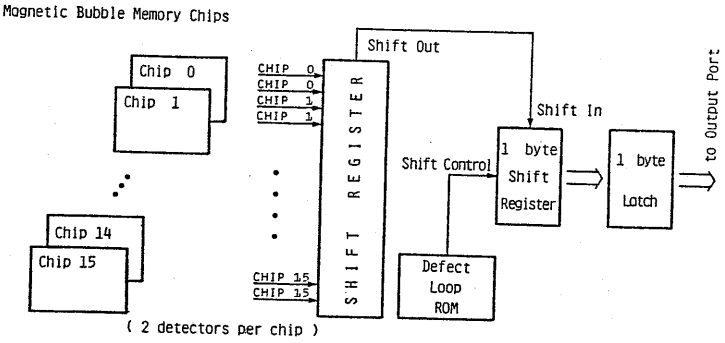

Figure 7 Defect Loop Control Logic for Read Operation
(16 chips are activated in parallel)

reformed with such simple logic. For write operation, we can construct similar control circuit with slight modification.

6. The Organization of The Pilot Module

6.1 Global view of the pilot module

Fig.8 shows the block diagram of the pilot module. Several control mechanisms which are described in the previous sections are incorporated. To attain higher data transfer rate, sixteen 1 Mbit bubble chips are activated in parallel. Each chip is further divided into two subchips. Magnetic field frequency of 125 KHz attains the data transfer rate of 250 Kbit/sec per chip. The buffer loop is 128 bits long and the main loop 4Ø92. The pilot system is implemented using TTL SSI's and is connected to personal computer which can monitor the status of the several registers and RAM. The entry of RDM is 16 bit each.

This unit consists of five kinds of modules.

<> Central Control Unit--- The commands and status information is transmitted between the host and this unit. Receiving the commands, it controls all other units appropriately. It initializes the several registers such as record length counter and

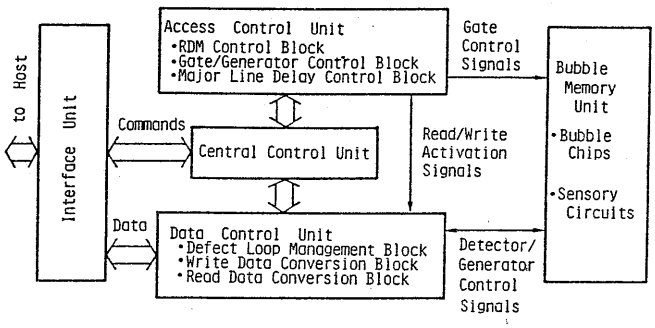

Fig.8 Overview of The Pilot System

address registers, and also supplies the comparand register in Access Control Unit with the requested record descriptor. The common clock signals for other control units and the rotating magnetic fields are also generated here.

<> Access Control Unit--- This unit controls the swap gate and transfer/block replication gate adequately using the record descriptor memory. For write operation, it checks several shadows and sends the generator initiation signal to Data Control Unit. For read operation, it sends the start/stop detection signals to the Data Control Unit accounting for the propagation delay on the major line.

<> Data Control Unit--- This unit is activated by the cue from the Access Control Unit. For write operation, it rearranges the data using the defect loop information and distributes the bits of record to multiple chips. For read operation, it reconforms the original record.

<> Interface Unit--- This performs the transmission control between the host and the pilot module and provides some buffer for the records.

<> Bubble Memory Unit--- This unit contains the bubble chips and their peripheral analog circuits such as the sensory circuits and rotating magnetic field driver circuits.

The two units, Access Control Unit and Data Control Unit, which play key roles in this pilot module, will be examined in detail in the following sections.

6.2 Access Control Unit

As shown in Fig.9, this unit is composed of three blocks, RDM Control Block, Gate Control Block, and Major Line Delay Control Block.

(1) RDM Control Block

This block, shown in Fig.9, contains two RAMs and several address registers to control gates and generators. As is described in section 4, several points in RDM need to be inspected for our dynamic addressing mechanism. Fig.10 summerizes these points:shadow on buffer(SB), on major line(SJ) and on main loop(SM0, SM1, SM2), and transfer point on buffer loop(TB), swap point on buffer loop(WB) and swap point on main loop(WM). RDM is constructed by 16bit-2048 words RAM for main loop and 16bit-64 words for buffer loop. Actually five address registers are associated with these RAMs and used to point the positions above except SJ. Swap point is taken as an origin and some of registers are shared. During every period of one field rotation, several addresses in RDM are read, and sent to Gate/Generator Control Block. New descriptor is written in RDM for write operation and the descriptors are exchanged for swap operation. Though many memory accesses are required for one bit shift time, relatively low frequency of magnetic fields makes the design of this block easier. Since a record is represented using two adjacent physical rows as described in section 3, the number of RDM entry is half of the main loop length and the duration of two magnetic field rotations are left for the gate control. Two clock cycles of 125 KHz are relatively long and serial memory access incurs no severe timing constraints. The initial values of address registers for shadows which depend on the

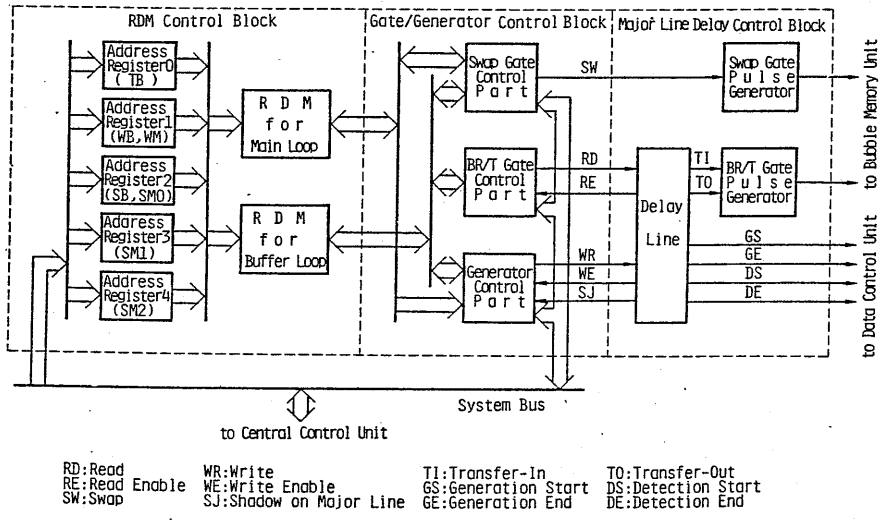

Fig.9 Organization of The Access Control Unit

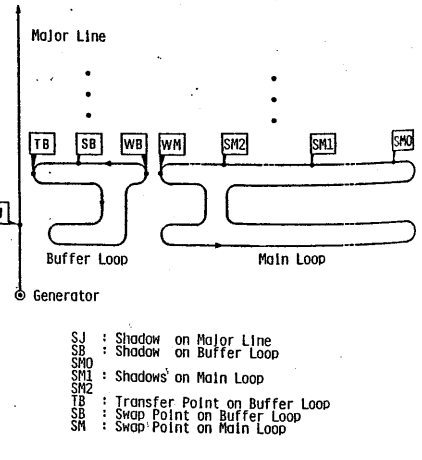

Fig.10 Hardware Pointers for Magnetic Bubble Memory Contro

record length are calculated and set by Central control Unit before the operation.

(2) Gate/Generator Control Block

As shown in Fig.9 this block consists of three part, swap gate control part, and block replication/transfer(BR/T) gate control part, and generator control part. The role of each part is literally apparent. The activation condition for each gate and generator is already presented in section 4. These parts contain comparators and search operand registers. The first one compares the record descriptor with the search operands given by Central Control Unit and determines the priorities of the two records at the swap gate. The swap gate is activated based on these priorities. Under current implementation sixteen search operands can be held. Due to relatively long period of two field rotation time, comparisons are performed serially with simple logic. The number of operands is easily increased. When the swap gate is activated, the record descriptors are also exchanged in this part. The second one, which controls the transfer gate, has the similar structure to the first one. The third one activates the generator. The expression of {6} in section 4 is evaluated here. Whether the shadows on loops are empty or occupied is examined using the record descriptor information given from RDM control block. Each activation signal is sent to Major Line Delay Control Block.

(3) Major Line Delay Control Block

This block controls the propagation delay on major line of a transfer gate to a detector and of a generator to a transfer gate. For read operation, transfer is prohibited for the time duration of record length, which is controlled by record length counter. In Fig.9 read enable signal to BR/T gate control part is used for this control. During detection of a record, the subsequent records are transferred out and multiple records proceed on major line. The heads of records on major line are marked in delay line constructed by RAM and counter. The detect start/end signal is made up using this delay and sent to Data Control Unit appropriately. For write operation, similarly to read operation, write enable signal is sent to Gate Control Unit. For records of some length, the shadow is produced on major line. SJ signal is generated by inspecting this delay line. The transfer gate is activated GL + RL unit time after the bubble generation. This control is also performed in the same way as for the read operation using the delay line. Registers for record length and the shadow addresses are initialized by Central Control Unit.

This block also contains gate pulse generators which arrange the phase and the width of gate control pulse appropriately. These parameters are tunable.

6.3 Data Control Unit

This unit distributes the bits of entering record and reconform the bits from chips into an original record using the defect loop information. It consists of three major block, defect loop management block, write data conversion block, and read data conversion block. As shown in Fig.2, the bits of a record are supplied in a way like right subchip of 1st chip, right subchip of 2nd chip,..., right subchip of 16th chip, left subchip of 1st chip, left subchip of 2nd chip,..., left subchip of 16th chip. Let the length of a physical record for a chip accounting the defective loops be RL. RL loops are used for each chip. It takes 16xRL loops as a whole to represent a record. The bits are read out from and written into sixteen chips in parallel. The defect information is stored in ROM adequate to this usage. The state of N-th loop is at addresses of from 8N to 8N-3 and from 8N-4 to 8N-7 of 8bit/word ROM. Once RL is given, related defective information is obtained by decrementing the address from 8RL. This is sent to read/write data conversion block bit serially.

As described in section 5, read/write data conversion block contains the data control mechanism shown in Fig.7, where the shift register is 32 bit long since 16 chips with two generator/detector per chip are activated in parallel.

7. Summary

This paper investigates the efficient control mechanism for the magnetic bubble memory with on-chip hierarchy. The design and the implementation of the RDM based controller are presented. The contents of RDM may include the logical record address, access key information and/or memory management flags. There is no absolute address

physically and all the access for the record is performed through the RDM synchronized with the movement of bubbles. The physical location of the record is dynamically allocated. The mapping between the logical address and the physical location is maintained by the controller using RDM. Frequently referenced records gather in the buffer loop and have short access time. The potentials of the hierarchical structure can be exploited by this flexible control mechanism.

For continuous read/write operation, this removes the inter-record gap time and attains high effective transfer rate, which is useful for the set oriented access of data base processing. The related records are gathered into the buffer loop through swap gate of high bandwidth, while the current record is being output. The buffer loop is almost full of output records, which results in continuous read operation almost without inter-record gap time. Besides, even for the short records they can be read out efficiently by transferring multiple records onto the major line continuously. For write operation, buffer loops also work well. Control scheme using shadows, which allows the dynamic assignment of vacant physical location for incoming records, realizes the efficient continuous write operation. These mechanisms plays a key roll in a data stream generator of relational algebra machine GRACE.

Thus by the flexible control mechanism, the memory system with the two loop bubble chips is expected to exhibit much higher performance than the conventional major/minor bubble memory system.

The pilot system with sixteen 1 Mbit bubble chips is implemented. The controller is made up with TTL SSIs, which results in relatively large board size. The current semiconductor technology, however, would integrate RDM and control circuits on a single chip.

The current system is rather restricted due to the organization of the bubble chip. The lengthes of two loops are already given. Incorporation of several lenghs of loops realizes the multiple level on-chip hierarchy. Higher performance can be expected. If the chip were to have another I/O port instead of house keeping area, the read and write operations could be performed concurrently. The further study will be presented in the future paper.

Acknowledgement

The author would like to thank Mr. K.Takahashi and Mr. H.Nagai of NEC for their offer of magnetic bubble memory devices and helpful advices on the implementation.

References

1. Y.Kita, N.Yamaguchi, M.Sugie, S.Yoshizawa, "The Development of a Bubble Memory Controller for Low-Cost File Use", IEEE Trans. on Comput., Vol.29, No.2, pp.89-96, Feb. 1980.

2. C.Tung, T.C.Chen, and H.Chang, "Bubble Ladder for Information Processing", IEEE Trans. on Magnetics, Vol.11, No.5, pp.1163-1165, Sep. 1975.

3. T.C.Chen, V.Y.Lum, and C.Tung, "The Rebound Sorter:An Efficient Sort Engine For Large Files", Proc. of 4th Int. Conf. on VLDB, pp.312-318, 1978.

4. W.Anacker, "Possible Uses of Charge-Transfer Devices and Magnetic Domain Devices in Memory Hierarchies", IEEE Trans. on Magnetics, Vol.7, No.3, pp.410-415, Sep. 1971.

5. P.I.Bonyhard, and T.J.Nelson, "Dynamic Data Relocation in Bubble Memories", Bell System Technical Journal, Vol.52, No.3, pp.307-316, Mar. 1973.

6. K.Takahashi, United States Patent. 3916397. Oct.28 1975.

7. A.V.Pohm, M.L.Convault, and S.R.Doctor, "Bubble Memories for Microcomputers and Minicomputers", IEEE Trans. on Magnetics, Vol.12, No.6, pp.636-638, Nov. 1976.

8. H.Chang, "On Bubble Memories and Relational Data Base", Proc. 4-th International Conference on Very Large Data Base, pp.207-229, 1978.

9. H.Kohara, K.Takahashi, S.Suga, and S.Fujiwara, "True Swap Gate Design for On Chip Cache Organization Megabit Bubble Memory", IEEE Trans. on Magnetics, Vol.17, No.6, pp.3038-3040, Nov. 1981.

10.H.Chang, "Data Structures and VLSI Algorithm for Bubble Chips", IEEE Trans. on Magnetics, Vol.16, No.5, pp.764-769, Sep. 1980.

11.M.Kitsuregwa, H.Tanaka, and T.Moto-oka, "Relational Algebra Machine GRACE", RIMS Symposia on Software Science and Engineering, Lecture Notes in Computer Science,Vol.147, pp.191-212, 1982.

12.M.Kitsuregwa, H.Tanaka, and T.Moto-oka, "Application of Hash to Data Base Machine and Its Architecture", New Generation Computing, An International Journal on Fifth Generation Computers, Vol.1, No.1 pp.63-74,1983.