POUGHKEEPSIE
Department 539
South Road Laboratory

October 6, 1958

FILE MEMORANDUM

SUBJECT: Binary Serial Disk File and Error Correction

BY: W. Buchholz

1. Lay-Out

For computers using binary addressing, such as the 709TX and 7000 Sigma, it will be very desirable to provide a version of the Advanced Disk File which is modified for binary addressing. This version would require a binary track selection mechanism, such as the one already being provided for the High-Speed Disk File. It also requires a slightly different information lay-out, for example:

	Decimal	Binary	
Usable disks	21/module 63 max.	22/module 66 total	
Information disk sides	120 max.	128	
Clock and spare disk sides	6 max	4	
Tracks per disk side	250	256	
Bits per byte (character)	7+ space bit	9+ space bit	
Bytes (chars.) per track	2500	2048	
Bits per track	20,000	20,480	
Max. capacity	75 million chars.	64 million (2^{26}) bytes 8 million (2^{23}) words	To be halved for error correction (see below).

Note: A. The added disk per module is already being considered for the High-Speed Disk File.

B. Three modules would always be provided for the binary file. Addresses proceed from disk to disk before advancing to the next track position.

C. Each byte contains 8 information bits which may be used for 2 decimal digits. The decimal digit capacity is, therefore, 128 million digits before error correction.

Dr. H. G. Kolsky

It appears reasonable to assume a single block (record) per track, so that the track would be the smallest addressable "bucket". The address assignment is then:

$$\underbrace{XXX}_{\text{Access}}\quad \underbrace{XXX\quad XXX\quad XX}_{\text{Track}}\underbrace{X\quad XXX\quad XXX}_{\text{Disk Side}}$$

The three high-order bits could specify additional access mechanisms either on the same stack of disks for multiple access or on additional stacks for greater capacity with a single channel.

2. Error Correction

Error detection and correction are important considerations in storage devices intended for long-term storage of vital data. When the probability of an error or malfunctioning causing irrecoverable loss of data exceeds a certain amount, the amount depending on the application and on human judgment, it will be necessary to provide back-up storage, such as repeated dumps on tape, to allow the file to be reconstructed. The cost of back-up storage and the complexity of restart programs will justify a certain amount of extra equipment in the main file to reduce the probability of failure below an acceptable threshold so that the back-up procedures and equipment become unnecessary. While the threshold and the probability of different types of error are hard to determine, it seems that reasonable error correction techniques can be devised which will be considered by most people as adequate to tip the scales in the direction of no back-up storage for most applications.

Error correction in a purely serial device, such as the disk file considered here, presents some problems because bits in a word or record are not independent. The Hamming type of error correction is biased towards a parallel bit representation where single bit failure is independent of other bits in the set. To record a Hamming type code serially on a single track appears to be of little value. It does not protect against outright failure of that track, whereas random single errors due to noise or intermittent failure can probably all be corrected by re-reading. If the probability of multiple errors in consecutive bits of a track is at all significant, as is certainly the case with high-density magnetic tape, the Hamming type of error correcting code recorded serially leads to the distinct possibility of false error correction being undetected.

The solution proposed here is to use duplicate recording. The specific proposal is to split the heads into two sets. The same information is written simultaneously on two tracks, using one head from each set. (Echo checks may determine that the writing circuit is functioning, but no attempt is made to read back after writing.) Only one of the two sets of heads is used for reading. As each byte (character) is read, it is parity checked. At the same time a count is kept of the number of bytes read in the track. If an error is found, reading is interrupted. The read circuits are switched to the alternate head and track. A whole revolution of the disks is taken until the byte in question is reached on the second track, as determined by counting bytes. Reading then continues. If another error is found in a subsequent byte, reading is again transferred to the first track in the same manner.

To ensure that both sets of heads are kept in use and in good repair during periods when no errors are encountered, it would seem advisable to alternate heads every time a new Read operation is started.

Two variations of the scheme might be considered to avoid the extra revolutions which might become a problem when operation is to be continued during a time of intermittent troubles until the next scheduled maintenance period:

1. Shift one set of heads along its tracks slightly. Read and count bytes in both sets. Normally transfer information from the early set. In case of error, switch to the late set. The two sets are shifted enough so that operation can continue with the maximum expected skew.

2. Read, check, and compare bytes from both sets. Because of skew, this requires a certain amount of byte storage and skew elimination control in each set of tracks.

It would appear that the main cost of the error correction scheme is in halving the total capacity of a disk file. Actually, the storage reduction by 50% is not too much out of line with the Hamming scheme. If the 8-bit bytes were to be handled in parallel, 5 bits must be added in the Hamming scheme. The reduction is then 5/13 or 38%.

3. Comments

The decimal Advanced Disk File organization, as now proposed (*), provides for verifying the writing operation by reading back from the file on a second revolution and repeating the operation in the computer so that the data may be compared. The assumption is that errors introduced after the information has been written correctly are only those caused by random failure or noise during reading, which can be corrected by re-reading, or those due to outright failures which after repair allow the information to be read properly. This appears to be a reasonable assumption but there are several reasons why this mode would be unsatisfactory for the 7000 Sigma system:

(*) Advanced Disk File Engineering Machine Objectives, San Jose PDL, July 24, 1958.

1. Every Write operation is lengthened by the 33 milliseconds needed to read back on a second revolution. This compares with taking an extra revolution during reading only when an error is detected or with the possibility of adding hardware to avoid even this small loss. The loss in performance during writing is significant in a high-performance system, especially when the disks are used in a sequential mode.

2. The read-back scheme, as proposed, is restricted to writing one track, or a portion thereof, at a time. The 7000 Sigma system provides for scatter reading and writing and for handling multiple records with a single instruction, which would make a track-at-a-time operation very awkward and probably unusable.

3. Because of point 2, the Exchange to which the disks will be attached in the 7000 Sigma system has not been provided with the ability to repeat an operation automatically. Read-back after writing would, therefore, have to be programmed, which is equivalent to not providing anything to assist the user.

4. A major advantage of error correction schemes is to allow operation to continue after outright failure and to avoid unscheduled maintenance whenever possible. The read-back scheme lacks this important facility and, therefore, still appears to require back-up storage for most real-time applications.

In a sense, the difference between the two schemes is a factor of 2 in storage capacity versus a factor of 2 in performance during sequential writing operations. Because of systems considerations, the error correcting scheme is the one we would like to consider for the binary version of the serial disk file.

WB/pkb

cc: J. E. Applequist - San Jose PDL
J. D. Calvert
S. W. Dunwell
W. N. Hagood - San Jose PDL
H. G. Jones
J. R. Lyon
T. Noyes - San Jose PDL
L. D. Stevens - San Jose PDL
J. M. Taylor - San Jose PDL
R. L. West - SEPD
H. K. Wild
7000 Product Planning
7000 Planning