An Analysis of Tape Drive Technology

JUAN A. RODRIGUEZ

Abstract—A history of the half-inch digital magnetic tape drives as viewed through the development improvements in the various component subsystems within the tape drive is given.

I. Introduction

A. History

THE HISTORY of magnetic tape data storage has been the history of half-inch tape. The first successful commercial product appeared in 1953 where data were recorded at a density of 100 bytes per linear inch (byte/in) of half-inch-wide magnetic tape. The initial success further financed research and development of newer, faster, and better machines to the point that approximately 100 000 half-inch tape transports are in existence today.

Today there are additional forms of magnetic tape storage in use which have been introduced to fill the need in areas of data gathering and processing not covered by half-inch tape. One of the most successful has been the tape cassette, a field used to collect and process data in those applications where low performance in terms of density of recording and processing speeds is compatible with low cost, such as point-of-sale terminals, very small data processors, etc.

This paper will concentrate on the history of half-inch tape.

B. Compatibility-Performance

The most amazing thing about half-inch tape is the upward/downward compatibility that has been imposed upon this medium. Today's machines can read tapes that were generated 20 years ago at 200 byte/in and have the capability of regenerating data at 6250 byte/in on that same reel of tape. In the same time frame, throughput rates have increased from 15 000 to 1 250 000 byte/s.

II. The Drive Architecture

A. General Drive Architecture

Fig. 1 shows the tape path of a 200-in/s half-inch magnetic tape drive. The tape reel on the right contains the source data. It is mounted on the drive by a computer operator. When the loading operation is initiated, the hub expands automatically to grip the reel. The reel then rotates in the clockwise direction and the tape moves toward the left reel. The tape is supported by air jets located in the right threading channel. The tape drops into a channel as it passes the READ/WRITE head and then is lifted by air jets on the left threading channel and carried through to the machine reel. The hub of the left reel has vacuum applied which attracts the tape. The clockwise motion of the reel wraps the tape several revolutions until enough tape is taken up so that vacuum on the reel can be discontinued. At this point, the left reel stops clockwise rotation and starts a counterclockwise rotation. The right reel keeps on turning clockwise, which provides a slack condition of tape between the two reels.

Vacuum is then applied to the columns. This action pulls the tape into the columns. Once the tape reaches the middle of each of the two columns, the capstan is activated to move the beginning-of-tape reflective marker to a position above the beginning-of-tape/end-of-tape sensor. The machine is then ready for system use. The tape drive shown is the latest in the evolution of tape drives. About 23 years ago the vacuum column was invented for tape control. It was designed to act as an air-cushion buffer between the acceleration requirements of the capstan and the acceleration limits of the tape reel and reel motor. In the machine shown in Fig. 1, the capstan will move 10 in of tape before the reel motors start to move.

The tape is shown with the oxide-coated surface touching only two points: the READ/WRITE head and the tape cleaner. This development did not take place until the late 1960's. Before then, the oxide coating was in contact with the vacuum column walls, capstan, READ/WRITE head, and tape cleaner. The change was made to minimize the debris generated from the oxide coating rubbing against the column walls.

The automatic hub engagement and disengagement occurred in 1970. The needs for such automatic engagement came about from the force requirements needed to hold the reel on the hub without slippage. The typical hub needs approximately 10 ft · lb of torque to prevent the reel from slipping. Forces necessary to generate this torque are in the order of 30 lb. This was a serious human factor problem since people were required to provide this force. The problem is less significant in lower speed machines since they require less force to close and open the hub (the torque required is proportional to the square of the speed). Nevertheless, the automatic hub was introduced as a matter of convenience in the lower speed machines. The column walls shown in Fig. 1 contain a sensor as part of the reel control system. The ultimate purpose of the reel control system is to prevent the tape loop from going beyond the top of the column, or going to the bottom of the column during normal operation. If the tape goes beyond the top, it will uncover the inner air bearings, which in turn will drastically reduce the tape tension across the head; or if the tape reaches the bottom of the column, the vacuum port will seal off at the bottom of the column, which again reduces the tape tension to zero. Either condition is a failure, which will cause a read or write error; in addition, because of the lack of tension, mispositioning of the tape with respect to the head may occur. The likelihood of the latter is such a serious potential problem in data processing (the mispositioning tape can cause the skipping of large blocks of data) that the drive will indicate the failure, and the system will most likely abort the data processing on that pass. The continuous sensor provides position feedback to the classic feedback control system in order to provide better loop control and minimize the velocity overshoot in the system. On minimizing the velocity overshoot, we can minimize the maximum torque requirements of the system and thus minimize the acceleration requirements

Manuscript received October 22, 1974; revised January 28, 1975.
The author is with the Storage Technology Corporation, Louisville, Colo. 80027.

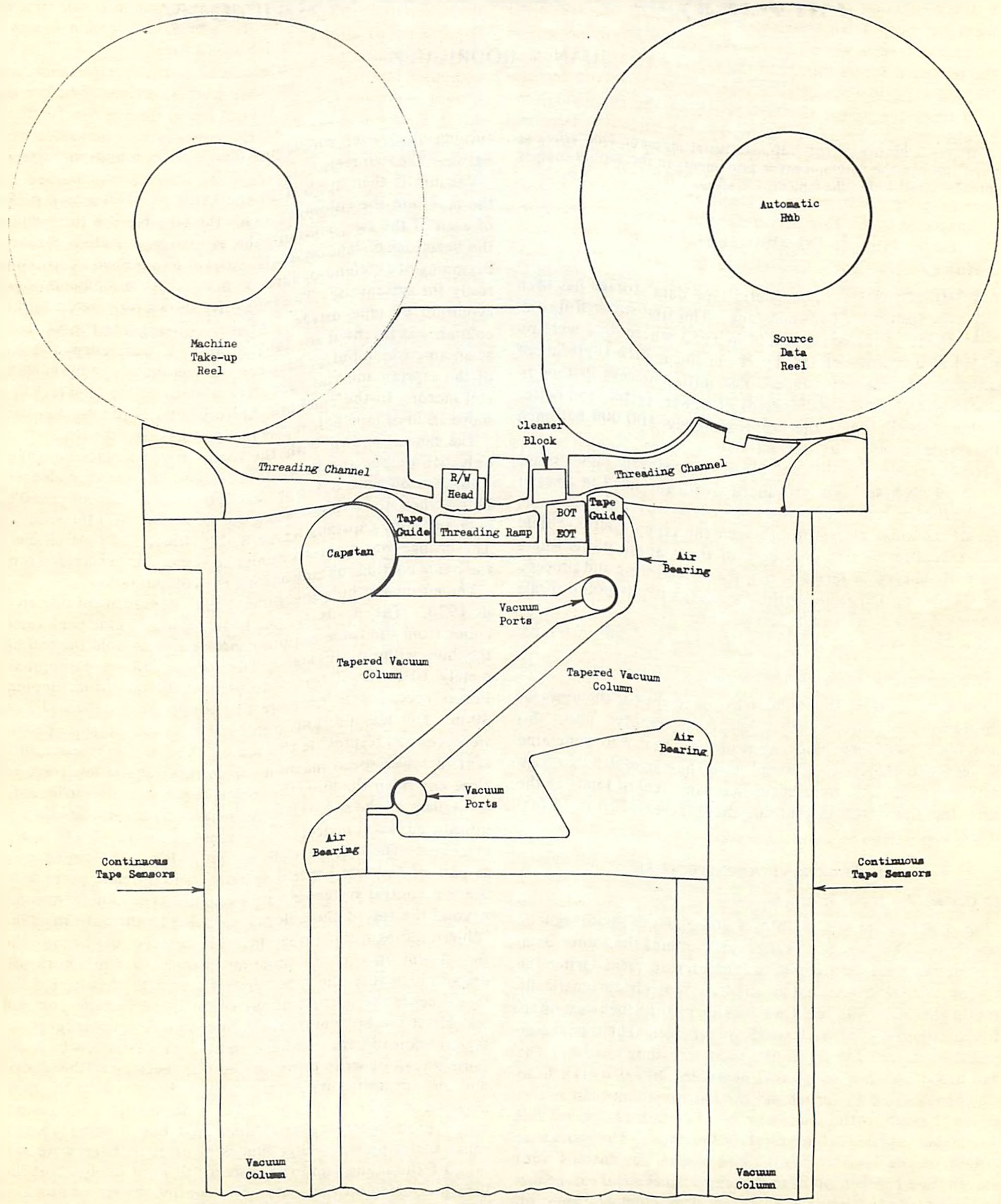


Fig. 1. Typical tape path area of magnetic tape unit.

on the reel. There have been several studies made on how much acceleration the reel can withstand before the tape stacked on the reel will shift. The reader can visualize that the layers of tape are kept together in the reel only by the frictional forces between the layers in the reel. Also, the reader can conceive that if this reel were to be accelerated with higher and higher torque, there would be a point where the acceleration forces would be large enough to overcome the frictional forces holding the layers of tape together in the reel. Once the frictional forces are overcome, layers of tape slip with respect to each other, which damages the tape. These studies have been quantitatively inconclusive because they are a direct function of the tape environment and that environment has been a large variable. For example, the tension at which the tape is wound on the reel contributes immensely to the frictional forces that hold the layers together. In storage, because tape is a viscoelastic material, these forces decrease with time; the longer the tape is stored, the more likely the acceleration forces are to overcome the interlayer frictional forces; consequently, the longer the tape is stored, the more likely it is to have interlayer slippage when it is read. This is why periodic maintenance programs are suggested to the users of magnetic tape. This periodic maintenance both minimizes the tension variations on the tape and provides for periodic cleaning of the tape. Empirical data suggest that the maximum angular acceleration that is applied to the reels should be kept to less than 10 000 rad/s^2 if the unit is to perform reliably in this most important area of tape handling.

The tape cleaner shown in Fig. 1 is another part of the machine that has progressed through 20 years of evolution. Changes in this area have resulted from botn the need to reduce the dirt particle size with increasing densities and the change in tape formulations. Under nominal conditions, tape is in contact with the head during a READ/WRITE operation. A particle that is imbedded in the tape will force separation from the head. If we assume, as a general rule, that a 26-dB loss in signal amplitude will cause failure in detection of the read back signal, we can formulate that tape cleaners are needed to remove particles larger than 107 μin at 6250 byte/in. In addition, viscous materials secreted by tapes must be removed from the surface of the tape before the tape reaches the head. These highly viscous materials, on contaminating the head, provide a base for loose particle retention. Only through the design of better cleaners and self-cleaning head contours can this viscous material problem be resolved. The solution to this problem is not a general one. Particular solutions implemented are highly proprietary in nature, and each solution is good only until that next time when either the tape formulation or the user application changes.

III. Cost/Performance of Half-Inch Tape

Over the past 20 years there has been a significant improvement in the performance of the tape. These improvements have come in the area of physical, chemical, and mechanical properties of the tape. There has been little or no change in the area of magnetic properties due to the upward/downward capability imposed on this medium. This is the reason that today's 6250-byte/in machine can use a tape manufactured 20 years ago. The overall READ/WRITE performance will be poorer with a tape manufactured 20 years ago, but the general signal quality will be very close, if not identical, to that generated in the tape manufactured today. In the same time frame, the cost for a reel of tape has come down by a significant factor. The combination of lower tape prices and higher densities has improved the two figures of merit that are so important to mass storage. Twenty years ago the cost per bit stored was 10^{-5} U.S. \$/bit and the volume needed to store a bit was 0.3×10^{-5} in^3/bit. Today they are 10^{-10} U.S. \$/bit and 10^{-7} in^3/bit.

IV. Actuators

The performance requirement for tape actuators has been dictated by the need of the system to quickly access the data that are recorded on tape, or to quickly access that point in tape that is to be recorded with data from the computer. Quick access is dictated by the need to minimize idle time by the computer. Reading in the backward direction is also a requirement dictated by quick access time.

A compromise between access time, cost/performance, and computer idle time is one that must be arrived at before the system is designed. The typical access time in the 1960's at 112.5 in/s was 6 ms. At the time, typical actuators were constant-speed dual capstans driven by synchronous motors. In one case, to move the tape, an idler capstan pushed the tape against the moving capstan. This system was commonly referred to as a prolay. The next most common system used air pressure and air vacuum on the capstan: moving the tape away from the capstan with pressure and thereby stopping tape, and moving the tape to the capstan with vacuum and thereby moving tape. Both systems behaved very much in the same fashion; because of electromechanical delay times or pneumatic delay times, the tape did not actually begin to move until 2 or 3 ms after initiation of the command to move the tape. Once contact was made, the tape accelerated very fast, typically in less than 1 ms, but due to the inherent spring mass properties of the tape and the capstans, the tape did not achieve final velocity (within 10 percent of nominal) until 2 or 3 ms later.

In the late 1960's, a new motor appeared with a very high torque-to-inertia ratio. The basic innovation in this dc motor was to replace the classical armature consisting of wires imbedded in a rotating solid-iron cylinder with an armature consisting of a rotating thin cylindrical shell holding the armature conductors (either a woven-wire armature or a printed-circuit armature) and a concentric nonrotating solid-iron cylinder (to serve as a return path for the flux in the motor). This change significantly reduced the inertia of the armature (typically to 0.005 in · oz/s^2) while maintaining the same torque of a classical motor of similar geometry. This motor was created to fill the needs for faster access time and was a significant step forward in tape velocity control for two reasons: there was no elapsed time before the capstan started to move, and, because of this, the tape acceleration was less than with the previous actuators described. The combination of a softer acceleration curve and closed-loop control of the capstan resulted in achieving stability much sooner than with the previous actuators. Because of design and cost/performance factors, only a single actuator could be placed in the machine.

This led to the development of single-capstan half-inch tape drives, which in turn resulted in a new set of problems; now the capstan had to both pull the tape in the forward direction and push the tape in a backward direction, whereas before with dual capstans the tape was pulled in both directions. This new set of problems came about because in accelerating the tape in the backward direction, acceleration forces decreased the tension of the tape across the head (the head-to-tape separation is directly proportional to the tape tension). In order to decrease the length (mass) of the tape being accelerated, the right-hand pocket was established (see Fig. 1). With this pocket, the length of tape being accelerated became that length of tape between the capstan and the middle of the right pocket. With a 2-ms access time in a backward direction, the acceleration forces on the tape created a decrease in tape tension across the head. This effect precipitated a new concept in the design of head contours to minimize the tape-to-head separation.

In 1974, tape drives of the 6250-byte/in format were introduced. The format at 6250 byte/in required a 0.3-in inter-record gap and 1-ms access time. The acceleration forces on the tape in the backward direction reduced the tension on the tape to nearly zero during the backward acceleration time. This again created the need for further improvement in the design of head contours.

The decrease in tape tension to zero during the backward start dictated the need to apply vacuum to the capstan in order to increase the frictional driving forces between the capstan and the tape. To understand the effect of vacuum on the capstan, let us write the equation for the tension forces across the capstan without vacuum:

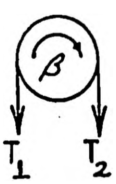
$$\Delta T \leqslant \epsilon^{\mu\beta}$$

where

$$\Delta T = \begin{cases} T_1/T_2, & T_1 > T_2 \\ T_2/T_1, & T_2 > T_1 \end{cases}$$

in which T_1 and T_2 are the tension forces on both ends of the tape across the capstan, μ is the coefficient of friction between the tape and the capstan, and β is the wrap angle across the capstan. The inequality indicates that if the value of ΔT is greater than that of the right side, the equation is no longer valid and slip will occur between the tape and the capstan.

Let us compute some typical values of $\epsilon^{\mu\beta}$: β is typically π rad; μ can vary from 0.3 to 1. This gives us a typical range of $\epsilon^{\mu\beta}$ from 2.5 to 23. If we take T_2 to be the tension of the tape across the head, therefore the tension on one side of the capstan, and if T_2 approaches zero, T_1/T_2 will become very large and certainly exceed the maximum value of $\epsilon^{\mu\beta}$.



On applying vacuum to the capstan, we modify this expression to be

$$\frac{T_1 + T_v}{T_2 + T_v} \leqslant \epsilon^{\mu\beta}$$

where T_v is chosen to be equal to twice T_1 under steady-state conditions, and the other terms are as in the previous equation. If now T_2 approaches zero and T_1 approaches twice its steady-state value, the ratio will now be 2, which is less than the minimum value of $\epsilon^{\mu\beta}$.

V. Coding Schemes

A. NRZI (see Fig. 2)

In NRZI, a logical ONE is recorded on tape as a transition (change in magnetic polarity) on tape. A logical ZERO is no transition on tape. This scheme requires an external clock to identify bit cells, that is, a clock that will define the position of logical ZEROs (where no transitions are actually present on tape). The clocking in the half-inch NRZI schemes will be described in Section VI-A.

Failure to detect NRZI properly can occur through a loss in signal-to-noise ratio or a shift in the position of the magnetic transition due to pulse crowding.

The concept of loss in the signal-to-noise ratio must be examined in the light of the fact that a logical ZERO is no transition at all. This problem has been solved by guaranteeing that

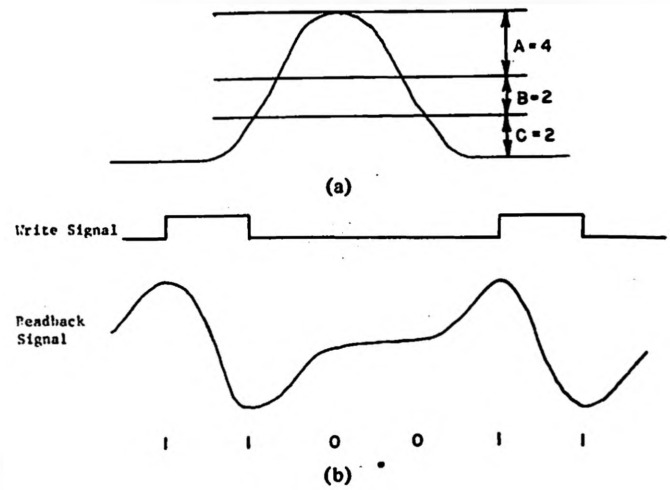


Fig. 2. (a) Isolated pulse showing write detection band. (b) NRZI recording.

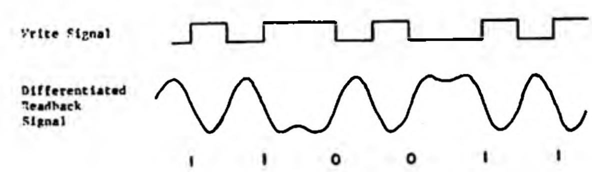


Fig. 3. Phase-encoded recording.

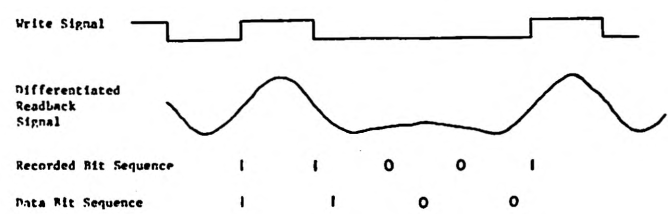


Fig. 4. Group-coded recording.

during the write process, there is an amplitude band (B in Fig. 2(a)) above which a ONE is detected (A) and below which a ZERO is detected (C).

The amount a ONE's transition can move before failure occurs is halfway between its nominal position and the ZERO's nominal position (Fig. 2(b)). At this point, the position of that transition is indeterminate.

B. Phase Encoding (see Fig. 3)

In phase encoding, a logical ONE is defined as a transition from one magnetic polarity to another at the center of the bit cell, and a ZERO is defined as a transition in the opposite direction at the center of the bit cell. The transitions at the center of the bit cell are called data transitions. Between data transitions, there may or may not be a transition depending on the sequence of ONEs and ZEROs. This in-between transition is called a phase transition. In this encoding scheme, the limits of detection allow the data transition to move to the edge of the bit cell before failure occurs. The advantage of this encoding scheme over NRZI is that there is at least one transition per bit cell. This transition per bit cell gives phase encoding a self-clocking capability, that is, there is information in every bit cell and, therefore, no need for an external clock. The disadvantage of this scheme over NRZI is that, for a certain bit density, the system must have the capability to record at twice the transition density of NRZI.

C. Group-Coded Recording (see Fig. 4)

This encoding scheme offers a compromise between the minimum flux transition requirements of NRZI and the self-clocking capabilities of phase encoding. This scheme takes 4 data bits and converts them into 5 recorded bits. A recorded bit is defined as in NRZI.

The limitation on the 5 recorded bits is such that there shall be no more than 2 recorded NRZI ZEROs in a row. The 4 data bits have 16 possible combinations, while the 5 recorded bits can have 32 possible combinations. Out of the 32 possible combinations, there are 24 combinations that obey the basic rule of no more than 2 consecutive recorded NRZI ZEROs. By placing a further restriction that there shall be no more than 1 ZERO at a boundary of a 5-recorded-bit group, the number of 5 recorded-bit combinations that are allowable is reduced to 17. Of these 17 bit combinations, 16 can be chosen to be equivalent to the 16 data-bit combinations that can be recorded. This restriction is made to facilitate the logical implementations of sequences of 4 data-bit groups, that is, by making this restriction at the boundary, we can write any sequence of 5 recorded bits followed by any other sequence and still maintain the original requirement of no more than 2 consecutive recorded ZEROs. This recording scheme adds to NRZI a self-clocking capability by guaranteeing a minimum of 2 out of every 5 possible recording transitions while reducing the transition density requirement of phase encoding by 37.5 percent. The limits for detection are the same as for NRZI.

VI. Format Schemes

A. Seven-Track 200 Byte/In

1) The Format: In this format there are seven tracks recorded simultaneously on tape, six of which are data, and the seventh a parity track. All seven tracks are recorded in the NRZI coding scheme. In this format, a byte is defined as 6 parallel data bits and a parity bit.

The number of bytes recorded is equal to the number of data bytes plus an additional byte called the "longitudinal redundancy check" (LRC) byte. This LRC byte serves two purposes: to return the track to the polarity of the interblock gap, and to provide a means of error detection. By returning the polarity of each track to that of the interblock gap, the number of transitions in each individual track is guaranteed to be even.

This format has very little error correction capability, that is, if a bit is dropped out of any track, it is theoretically impossible to determine in which track the error occurred. The only hope for error recovery is to reread the record until no error occurs.

The actual byte detection in NRZI will now be described. The parity track is defined to contain odd parity, that is, the parity track is written in such a way that the number of transitions in a byte is odd. From this odd number of transitions, we will generate a byte clock, that is, the first bit in a byte will open a window whose typical duration is half a bit cell. Any transitions that occur in the other six tracks while that window is active will be determined to be ONEs. When the window ends, the byte is assembled, odd parity is checked, and a Modulus 2 bit count is kept on every track until the LRC byte is detected. If even byte parity is never detected, and if every one of the individual Modulus 2 bit counters is ZERO, the record read is considered to be error-free.

2) Mechanical Problems in Aligning Bits Within a Byte: The basic problem in NRZI detection is one of aligning all the bits in a byte, that is, guaranteeing that after writing the tape, the machine that reads the tape will align the tape in such a way that all the bits within a byte will fall within half a bit cell. To see how this is obtained, we shall examine the geometry of tape moving past the head. At 200 byte/in, the nominal distance between bits is 0.00500 in. The byte cell is half a bit cell, which means that all bits must come within 0.00250 in. When writing the tape, additional tolerances must be subtracted from this number because of coded skew (due to pulse crowding), that is, when a bit is written on tape, its position relative to its nominal position will change depending on the bits recorded in the vicinity of the bit in question. In general, the bit in question will tend to move toward that position occupied by a ZERO. This coded skew can typically be 15 percent of the bit cell, that is, 0.00075 in, impling that we must write the bits within a byte at least within 0.00175 in from each other.

Now we shall look at the different mechanical parameters that determine the bit placement. The tape width is specified to be 0.498 ± 0.002 in. The tape guides are designed to be at least 0.5000 in wide to pass the widest tape. When the narrowest tape is past, there is 0.004-in tolerance in guidance. (Given a narrow tape and fixed flanges, the position of the tape with respect to the guides is indeterminate within the guides.) By making a simple geometrical calculation, and assuming that the head write gap is perfectly parallel to the center line of the two guides, and that the two guides are perfectly parallel to each other, we can calculate a worst case movement between two consecutive passes of 0.0033 in. Obviously, this is almost twice as much as our maximum detection limit. This problem is solved by inserting compliant guides, that is, the back flange of each of the two guides is a movable flange which pushes the tape against the front flange, thereby minimizing the relative movement of the tape across the head between passes. The front flange of each guide becomes the reference surface of the tape toward which the tape guides.

Let us now assume that this guiding is perfect and that there is no relative movement between successive passes. (The actual amount of relative movement measured between passes is less than 0.001 in.)

Now we come to a real problem in interchangeability between machines of different manufacture. The problem arises if the machines are not designed with the same spacing between the two guides and the same position of the head with respect to the two guides. We consider as an example the case where the tape curvature has been simplified to be triangular (see Fig. 5). The placement of the guides in the two independently designed machines is the same but the head position has been shifted in such a way that in one machine the head is to the left of the center line and in the other machine the head is to the right of the center line between the two guides. Again, going through a geometric exercise, using numbers for curvature that are equivalent to those specified for half-inch tape (1/8 in in 36 in), we can see that when interchanging tapes between these two machines of similar design, there will be a relative movement of 0.0066 in between the two machines. This is beyond the detection capabilities of the system. The real solution to this problem is for every tape transport to use the same spacing between guides and the same relative position of the head to the guides. In actual practice, differences be-

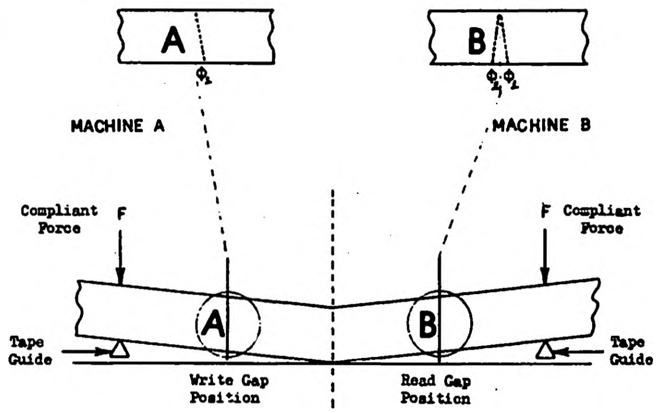


Fig. 5. Geometric presentation of the problem of skew.

tween machines are not as great as I have shown here and tapes do not, in general, show that much curvature. But the example indicates the type of problem encountered if machines are not designed with exactly the same spacing. This problem has not been solved by the industry because it was not recognized to be a problem until many thousands of machines had been built by the different equipment manufacturers. It follows that if you are required to design equipment that has to be "Brand X" compatible, you take a very close look at their tape path geometry.

B. Nine-Track 800 Byte/In

This format is an extension of the seven-track 200-byte/in format, with obvious improvements over that format and some added complications. The data format consists of nine tracks recorded simultaneously. Each track is recorded in the NRZI format. A byte consists of 8 bits and a parity bit, odd parity. The next to the last byte written in the record is called a cyclic redundancy check (CRC) byte, which is a byte generated using the following formula.

1) C is computed from data polynomials M_1 through M_n using the generator polynomial

$$G_1 = X^0 + X^3 + X^4 + X^5 + X^6 + X^9$$

according to the following relationship:

$$C = (X^n M_1 + X^{n-1} M_2 + \cdots X^2 M_{n-1} + X^1 M_n) \text{ Modulo } G_1.$$

All arithmetic operations are Modulo 2.

2) The calculated CRC character is modified by the polynomial

$$1 + X + X^2 + X^4 + X^6 + X^7 + X^8$$

EXCLUSIVE ORed with C in the corresponding bit positions; the resultant is the CRC character.

The last byte in an LRC byte is similar to that in the 200-byte/in format. The CRC byte has the characteristic that it can detect the track that is in error as long as the error is not cyclic every 9 bits. On identifying the track in error, the system attempts a reread to correct the track previously identified in error. Whenever a parity error indication is detected, the CRC character is regenerated during the error correction operation and, if it matches that recorded on tape and the LRC byte is matched, the record will be considered to have been corrected successfully, and the data will be shipped to the user. In all other respects, this format contains all the advantages and disadvantages of the 200-byte/in format, with the added complication that all the tolerances for successful detection of a byte described in the 200-byte/in section are reduced by a factor of 4. That is, in order to detect a byte successfully, all the bits within the byte must be aligned within 0.0004 in. In spite of all these problems, and because these problems were solved, this system was very successful and was the base of half-inch tape technology for over eight years.

C. 1600 Byte/In

This system contains 9 tracks recorded simultaneously. Each track is recorded in the phase encoded scheme. A byte is defined as 8 bits plus a parity bit, odd parity. Each track starts with a 40-ZERO preamble followed by a ONE. The 40 ZEROs are used for synchronizing clocks, the ONE to detect the beginning of data. The last data byte is followed by a postamble consisting of a ONE followed by 40 ZEROs.

This format has a great advantage over 800 byte/in in that through electronics the skew problems are solved. Since each track is self-clocking and each track is detected independently of every other track, its relative position to any other track is rather immaterial, at least in concept. Aligning of the bits in each byte is done electronically through the use of skew buffers. The first ONE in the format identifies the beginning of data. Thereafter, each bit is assigned sequentially to a byte. When all the bits in that byte are received, the byte is transferred out.

The 1600-byte/in format gave significant improvement in performance over the 800-byte/in format. It also was the base for half-inch tape technology for over seven years.

D. 6250 Byte/In

This system has nine tracks. A byte consists of 8 data bits and a parity bit, odd parity. Each track is recorded in the group-coded recording code. The great deviation from 1600 byte/in is that every eighth byte is now an error-correcting code (ECC) byte that is generated using the following formula.

1) The 8 bits of each data byte D_1 through D_7 are coefficients of polynomials D_1 through D_7, respectively, and have the following assignments:

polynomial position	X^1	X^4	X^7	-	X^3	X^6	X^0	X^2	X^5
track number	1	2	3	4	5	6	7	8	9.

Track number 4 contains odd parity bit P and is not a part of the ECC character.

2) The 8 bits of the ECC byte E are also coefficients of polynomial E with the same track assignments. Track number 4 contains odd parity on the 8-bit byte E.

3) E is computed from data polynomials D_1 through D_7 using the generator polynomial

$$G = X^0 + X^3 + X^4 + X^5 + X^8$$

according to the following relationship:

$$E = (X^7 D_1 + X^6 D_2 + X^5 D_3 + X^4 D_4 + X^3 D_5 + X^2 D_6 + X^1 D_7) \text{ Modulo } G.$$

All arithmetic operations are Modulo 2.

By combining the parity track and the ECC byte, this system can detect and correct all single-track errors and, with *a priori* pointers, correct all double-track errors. This format has an additional advantage in that there is a resync burst every 1106 data bytes. The purpose of the resync burst is to resynchronize the skew buffers in those tracks that have been declared bad. When a track has been declared bad (for example, through the ECC error detection mechanism), the position of a bit in that track to its byte is assumed to be lost. The resync burst is uniquely detected in each track which is used to synchronize all tracks to each other. The resync burst, in essence, allows every track in a record to go bad at one time or another and the record will be read successfully. In order to ensure that all this error correction has been done properly, two independent CRC bytes are recorded at the end of the data portion of the record. During the read operation, and after error correction, similar generators are used to generate identical bytes and, if these CRC bytes match at the end of the read back operation, successful read operation is guaranteed. The format, as in 1600 byte/in, contains a preamble, in this case consisting of 80 ONEs and a Mark 1 character. This Mark 1 character is not one of the 16 data combinations; it is one of the 24 combinations previously described in Section V-C. The Mark 1 character is a 00111 combination of recorded bits. It can be written because we know *a priori* that it is preceded by all ONEs. The basic restriction of a maximum of 2 consecutive ZEROs is guaranteed. The record is followed by 80 ONEs preceded by a Mark 2 character which consists of a 11100 combination of recorded bits. Again we guarantee the maximum of 2 consecutive ZEROs, knowing *a priori* that they will be followed by the 80 ONEs in the postamble. The resync burst consists of a Mark 2 character followed by 2 all-ONE's 5-bit characters followed by a Mark 1 character. This 20-bit sequence is found nowhere in data and, therefore, can be recognized uniquely as a resync burst. At this time, the 6250 byte/in is outperforming the 1600 byte/in. Being a student of history, the author can only say that formats will be developed at higher densities than 6250 and that these higher densities will outperform the present density.

VII. The Future of Magnetic Tape Storage

Over the past five years, several forms of magnetic tape storage have been created to fill the spectrum of needs in the ever-growing data processing industry: the magnetic strip credit card, the tape cassette, the floppy disk, and the IBM 3851 tape cartridge. The 6250-byte/in half-inch format was also developed and delivered in this time frame.

Each of the above methods of magnetic storage has been created to fill a different need at a different optimum point in the cost/performance curve. To say that the magnetic strip credit card is a "low-performance system" is to minimize the environmental hazards in which such a card has to perform. None of the other forms of magnetic storage would survive in the environment of the credit card. The example is chosen to demonstrate that no one form of storage is the answer to every one of the needs.

The future development of magnetic tape storage is one which will occur as the need arises. Some of the present-day commercial limits in magnetic recording include the following:

1) 9000 linear bit/in in the 6250-byte/in half-inch tape format;
2) 400 tracks/in in high-performance disk systems;
3) face powder, wallets, heat, cold, and sweaty hands for the credit card environment.

Within the confines of the commercial limits, the technology can develop many variations to fit specific needs in the data processing world. Each new variation is, by definition, attempting something never tried before, that is, each is working at the state of the art. Each variation faces its own unique set of problems, each of which must be solved in an inventive way.

To summarize, in the future of magnetic tape storage we will find all of today's products, or an improvement of them, plus many other products developed to meet a need.