An experimental digital consumer recorder for MPEG-coded video signals

Citation for published version (APA):

DOI:
10.1109/30.468087

Document status and date:
Published: 01/01/1995

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
Abstract — The concept and real-time implementation of an experimental home-use digital recorder is presented, capable of recording MPEG-compressed video signals. The system has small recording mechanics based on the DVC standard and it uses MPEG compression for trick-mode signals as well.

1. Introduction

Recently, advanced video compression has become an emerging and enabling technology for new systems in digital transmission and storage. The MPEG standard [1] [2] is considered as a key technology for processing high-quality video pictures at minimum bandwidth. Digital video broadcasting based on MPEG-2 will start in near future and the necessity for exploring home-use MPEG recording becomes apparent.

Simultaneously, several experimental recording systems have been worked out for standard-definition television (SDTV) signals [3] at ca. 20 Mbit/s video bit rate or even high-definition signals (HDTV). Key technologies in these systems are bit-rate reduction [4] and a trend towards small recording mechanics in order to ensure portable applications. The system proposals have culminated into the so-called DVC system, which was recently announced [5] and which is being developed by a substantial group of companies. The DVC standard includes digital channel coding (e.g. ECC) for high robustness and error recovery, channel modulation with digitally embedded tracking tones [7], and advanced frame-based video compression techniques for obtaining sufficient playing time in combination with a small cassette.

Using the DVC recording system with its channel coding for error protection as a reliable recording channel, we attempt to record MPEG-compressed signals on this channel directly, without applying the DVC video compression, so that expensive video transcoding in the recorder is avoided. Currently, MPEG recording modes for ATV (HDTV in USA [8]) and DVB (Europe) with bit rates of 25, 12.5 and 6.25 Mbit/s are being standardized. The system described in this paper is based on 12.5 Mbit/s which is expected to be the most promising value for DVB recording.

Let us now briefly identify the problems that occur when MPEG signals have to be recorded. Firstly, the signals have a variable bit

<table>
<thead>
<tr>
<th>parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>scanner</td>
<td>2 x 2 heads</td>
</tr>
<tr>
<td>scanner diameter</td>
<td>21.7 mm</td>
</tr>
<tr>
<td>scanner speed</td>
<td>9000 rpm</td>
</tr>
<tr>
<td>tape width</td>
<td>6.35 mm</td>
</tr>
<tr>
<td>tape thickness</td>
<td>7 μm</td>
</tr>
<tr>
<td>track pitch</td>
<td>10 μm</td>
</tr>
<tr>
<td>recording bit rate</td>
<td>41.85 Mbit/s</td>
</tr>
<tr>
<td>tracking modulation technique</td>
<td>embed. track. tones</td>
</tr>
<tr>
<td>ECC channel coding</td>
<td>24-25 code</td>
</tr>
<tr>
<td>available bit rate</td>
<td>R-S product code</td>
</tr>
<tr>
<td>input signal</td>
<td>12.5 (25/50) Mbit/s</td>
</tr>
<tr>
<td>trick-mode coding</td>
<td>separate MPEG-2</td>
</tr>
</tbody>
</table>

Table 1. Key parameters of MPEG recorder.

---

1The work reported in this paper was carried out in the framework of the European RACE 2026 DART project and was sponsored by the European Community. The authors are responsible for the contents of the paper.
cost per compression unit, asking for more robust synchronization and reliable data recovery. Secondly, the compressed data are not randomly accessible, but need decoding in a predetermined order. This is particularly problematic for implementing trick modes, such as fast search. Thirdly, MPEG signals are subject to accurate timing constraints, which are difficult to maintain, due to the distribution of data over the track format and the non-active periods during recording. The aim is to find a robust mapping of MPEG signals on tape, while providing facilities for trick-mode playback. A second objective for the system is to design a recorder that enables a spectrum of recording rates, in order to match with the different types of MPEG-coded signals (single program SDTV, multiprogram SDTV, HDTV).

For our experimental recorder we have chosen a basic recording mode of 12.5 Mbit/s, in which MPEG signals of up to 9 Mbit/s can be recorded. This capacity is expected to be sufficient for single-program recording, at least for the initial stage in typical broadcast situations like DVB. This is because SDTV streams for consumer applications will certainly not exceed 9 Mbit/s, even though the MPEG-2 main profile at main level can in principle go up to 15 Mbit/s.

From the aforementioned list of constraints it is clear that the limited access to MPEG data on a local basis, such as subimages, hampers a system solution in which the same recorded signals are used for both normal playback and trick-play. A high-quality implementation of trick-play needs periodic access to individual video frames, which would require full decoding of the MPEG stream. For this reason, we have adopted a solution in which separate (MPEG-coded) signals are recorded for trick-mode use, whereas the received broadcast signal is used for full-quality normal playback. The separate trick-mode signals are extracted from the MPEG transport stream on image level and converted into trick-mode signals. The special trick-mode signals are recorded, with some duplication, at specific positions on tape, so that these data can be accessed at particular search speeds.

2. System architecture

Figure 1 portrays the block diagram of the recording system. The MPEG data is received in transport stream format, in the form of fixed-length transmission packets and may originate from one or more programs. For the basic recording mode of 12.5 Mbit/s, only those packets are selected that contain data of one particular program. Simultaneously, the time at which packets are received is preserved accurately, because all selected packets will be subject to various manipulations for recording, while the MPEG decoder demands accurate timing and ordering of packets after data recovery from tape. Special time stamps are generated at the input and stored along with the data on tape.

MPEG coding uses intraframe (I-frame) and interframe-coded pictures (P and B-frames). The latter ones rely on previously coded frames,
whereas the former types are compressed independently. Consequently, only I-pictures are suited for trick-mode data, so that these I-data are extracted and transcoded to construct separate trick-mode streams. The total trick-mode data amounts to about 2.5 Mbit/s. The normal playback stream of up to 9 Mbit/s and the trick-mode streams are then multiplexed into a joint tape format of 12.5 Mbit/s for recording. For robustness, a fixed mapping of transport packets onto channel-coded data packets is applied. At playback, the recovered data from tape are de-multiplexed, and only data from the stream that corresponds with the actual playback mode is selected. From this data, the appropriate MPEG output stream can be reconstructed.

For error-correction coding (ECC), the DVC recording format employs Reed-Solomon (RS) product codes defined within a track. Although the performance of these codes is good for random errors, they are susceptible to burst errors in the direction of a track (e.g. head clogging). For this reason, extra ECC has been defined of about 1 Mbit/s, in which an RS codeword is formed by interleaving in a direction across the tracks. As a second step in channel coding, we have adopted the high-efficiency DC-free 24-25 channel code, described in [4][7], to constitute pilot tones which are embedded in the data.

The channel-coded signals are recorded with an experimental compact DVC recorder mechanism. In order to implement a set of possible bit rates using the same recorder (from 50 Mbit/s down to 6.25 Mbit/s), a scanner concept with $2 \times 2$ heads at 9000 rpm was adopted. At 50 Mbit/s, both head pairs are used continuously. The effective bit rate is halved by using only one head pair, and again halved by using this pair only once every two revolutions, and so on. The tape speed is lowered accordingly for an increased playing time.

3. Mapping of MPEG signals

3.1 Selection and smoothing

The input to the recorder is an MPEG-2 transport stream, consisting of fixed-length transport packets, as defined in [2]. A transport stream is a multiplex of multiple elementary streams, such as compressed audio, compressed video and data streams. It may carry several programs, each of which corresponds to a subset of the elementary streams (e.g. one video stream and one or more audio and data streams). The packets in the stream have an identification to indicate the elementary stream (and hence the program) that they belong to. This is illustrated in Figure 2a, where an input stream with packets from two programs A and B is shown. In the figure, the packets are drawn as rectangles on a time axis, in order to indicate a possible byte arrival pattern at the recorder input.

At the input of the recorder, only packets from a single program are selected for recording. The result is a partial transport stream with packets occurring at irregular intervals, as illustrated in Figure 2b for the case where program A is selected from the complete transport stream of Figure 2a. The average rate of packets in the partial transport stream equals the program rate and is less than 9 Mbit/s. Packets arrive, however, in bursts at the transport rate, e.g. at around 40 to 60 Mbit/s in case of streams originating from satellite or cable transmission.

In order to fit the bursty partial transport stream onto the fixed-rate recording capacity, both ‘smoothing’ and ‘stuffing’ must be performed. Conceptually, smoothing means reducing the burst rate of packets to the maximum available data rate (or a lower rate), and stuffing means adding dummy data to fill up the remaining gaps. The concepts are illustrated in Figure 2c. This figure shows the result of smoothing and stuffing the partial transport stream from Figure 2b. The elongation of the packets on the time axis illustrates the lower rate, and the shaded areas mark the inserted dummy data.
The result of smoothing and stuffing is that the relative timing of the packets in the partial transport stream is disturbed. Once the original timing is lost, a remultiplexing operation would be needed to generate a valid output that satisfies all complex MPEG timing constraints. For this reason, a time stamp mechanism has been introduced that enables the recorder to reconstruct the original relative timing at playback.

In the recorder time stamps are generated immediately after program selection. One time stamp is attached to every packet. Its value reflects the moment of arrival of the first byte of the packet at the input. Instead of taking the moment of arrival itself, however, a constant delay $\tau$ is added. This delay is such that de-smoothing (the inverse of smoothing) can be performed by simply taking the time stamp value as the new moment of delivery for each packet. This de-smoothing operation is illustrated in Figure 2d. It shows the stream that results after de-smoothing the fixed-rate stream from Figure 2c. It has the relative timing of Figure 2b, with an over-all smoothing/de-smoothing delay of $\tau$.

3.2 Time base generation

In the recorder, smoothing is performed at recording, whereas de-smoothing is at playback. Although these are unrelated in time, we nevertheless use the time stamps as described above, but now to indicate the moment of delivery from the recorder output at playback. This is possible by the definition of an appropriate reference time base that relates to the tape contents. This definition is used to specify a reference timing for a smoothed stream on tape. Based on this specification, we have implemented the time base generation in the recorder. The implementation is such that, with respect to the actual time bases, the timing of the smoothed stream will be the same at recording and playback. Likewise, the timing of the de-smoothed stream at playback will be the same as it would have been if de-smoothing had taken place at the recording side. Hence, the de-smoothed output at playback will indeed have the same relative timing as the (un-smoothed) input at recording. In terms of Figure 2, this implies that, if we take the time axis to represent the reference time base, then Figure 2c applies to both recording and playback, and the stream in Figure 2d can be interpreted as the output stream at playback.

The reference time base is a counter that is incremented by a clock at a nominal frequency of 27 MHz, and reset periodically, after a fixed number of clock cycles. This number has been chosen such that the reset period exactly corresponds to a small integral number of scanner revolutions at the nominal scanner speed. Hence it corresponds to a fixed number of tracks on tape. Based on this correspondence, the reset is defined to occur at specific track boundaries that are identified by the track numbering.

The actual generation of the time base at recording and playback is indicated in Figure 3 and 4 respectively. In these figures the relevant parts of the block diagram from Figure 1 are repeated, with more detail added. At both recording and playback, the reset signal for the reference time base counter is derived in relation to the servo reference and the track numbering in the format. As a result, the reset period will always be in accordance with the actual scanner speed, and the reset moments will match the correct track boundaries.

Let us now briefly consider the desired clock accuracy for correct MPEG program storage and playback. At recording, the 27 MHz clock is locked by a PLL to the Program Clock References (PCR's) that are present in the input stream. In this way, the recorded stream will have time stamps that are based on the same frequency as the PCR's in the stream. At playback, the delivery of this stream at the output is based on the time stamp values. Therefore, the frequency of the output Program Clock, which is reflected by the PCR's in the output stream, will equal the frequency of the playback time base. This is a 27 MHz clock taken from a free-running crystal. As a result, only the inaccuracy of the playback crystal determines the frequency accuracy of the output Program Clock. Any inaccuracy of the input Program Clock does not contribute, which is important because MPEG-2 allows at most 30 ppm deviation from the nominal 27 MHz frequency. Also, it implies that for copying the final accuracy is not affected by the intermediate playback and recording stages.

3.3 Mapping of normal-play signals

So far, we discussed smoothing and de-smoothing as conceptual operations. In the recorder, they
are implemented by appropriate buffering of packets. The respective buffers have been drawn in Figure 3 and 4. In the sequel, we will show that these buffers can be related to a buffer description in the MPEG system layer and we will address the value of the delay $\tau$ in our system.

The fullness behavior of the buffers is closely related to the implementation of the smoothing operation and the value of the over-all delay $\tau$. This can be seen from Figure 2, if we take the timed streams from Figure 2a-d as examples for the streams occurring on the points marked (a)-(d) respectively in Figure 3 and 4. In order to avoid overflow of the buffer at the playback side, $\tau$ should be as small as possible. On the other hand, in order to guarantee that the same buffer will never underflow, $\tau$ must at least be equal to the maximum relative delay that packet data may have incurred due to the smoothing operation at recording. In general, this relative delay follows from the buffer fullness of the recording-side buffer at the moment of the reception of the data at the input. Therefore, it depends on the burstiness of the selected program. In MPEG-2, limits on this burstiness are implied by a so-called smoothing buffer description in the MPEG system layer. This description refers to a hypothet-

ical buffer that accepts packets from a program, with data leaking out at a fixed rate. It states that the timing of program packets is such that a buffer of given size and leak rate will not overflow. If the buffer in Figure 3 would leak data at the maximum normal-play data rate to the format multiplexer at point (c), then it would be a smoothing buffer in the exact MPEG sense. For a given physical size of this buffer, a no-overflow guarantee can then be derived if selected programs have a suitable smoothing buffer description in the input data stream. Inversely, the safety of the de-smoothing buffer in Figure 4 may be guaranteed also, provided that it has at least the same size and that the over-all delay $\tau$ is set to match this size.

In the actual design of the recorder, a different rate is used at the interfaces with the format multiplexer and demultiplexer, at point (c) in Figure 3 and 4 respectively. Also, the data does not leak from the recording-side buffer, or enter into the playback-side buffer, uninterruptedly. This is due to the interleaving of trick-play data in the format multiplex. Similar buffer guarantees, however, can be derived in an analogous manner to those described above.

For the tape formatting, transport packet data is mapped into sync blocks. Sync blocks are the data packets that are used for the channel modulation and error correction coding (see Section 4.1). For robustness, we use a fixed mapping of 2 transport packets into 5 sync blocks. The complete format multiplex contains normal-play data, trick-play data and stuffing data, which are interleaved in fixed groups of 5 sync blocks.

3.4 Mapping of trick-play signals

Alongside the normal-play signals, which are mapped as described above, special trick-play signals are recorded on tape. These signals are intended for fast-playback modes of the recorder. The allocation of the signal data in the tracks is such that, under suitable playback conditions, all data for a given speed can be recovered, even though only a small portion of the tracks is scanned.

Prior to introducing the particular allocation scheme and the associated playback conditions, we first summarize the general aspects of data recovery in the helical-scan system during trick play. At fast playback, the scanner rotates at
the same speed as at recording and normal playback. The tape, however, moves at a higher speed than at recording and normal playback. As a result, the tape will travel a larger distance between two subsequent scans, and the heads on the scanner will move over the tape at a different relative angle from the track angle. The resulting movement of the heads relative to the tape is illustrated in Figure 5a for a fast-forward mode at a given speed.

Along the scan path, the amplitude and SNR of the trick-mode replay signals varies considerably. This is illustrated by Figure 5b and 5c. In Figure 5b we have schematically redrawn the scan path of Figure 5a, but now with respect to a more convenient representation of the tracks. In this figure, + and − are used to indicate the opposite azimuths of the tracks. Figure 5c shows the signal level for a single head with + azimuth. It represents the signal level as a function of the horizontal displacement, when the center of the head moves along the trajectory depicted in Figure 5b. The level is at a maximum wherever the center of the head is sufficiently close to a track of the correct azimuth. So wherever the head crosses such a track, a limited burst of sync blocks will be recovered. This is illustrated in Figure 5d, for the example of the threshold indicated in Figure 5c.

In our experimental recorder, we use two particular trick-play speeds, viz. 3.5× and 8.5× relative to the normal-play speed. At these speeds, the bursts of two subsequent scans will contain sync blocks at complimentary positions in the tracks. In general, this is true for any speed $m \times \frac{2}{7}$ with $m$ odd. For an explanation, we will concentrate on the + head again, as in Figure 5a–d. Incidentally, these figures exactly reflect the situation for the example of 3.5× ($m = 7$). At $\frac{2}{7} m \times$, the distance between subsequent + scans is $\frac{m}{2}$ times the distance between the + scans at recording and normal-play, i.e. $m$ times the track pitch. So, if we consider any given sync block position in the tracks, say on the dashed horizontal line in Figure 5d, then two subsequent scans will be equally far away from the center of the nearest track. Since $m$ is odd, the respective nearest tracks will have opposite azimuths. So at least one scan out of the two will be close to a + track. At worst case, both scans are half a track pitch away from the center of a + track, which is still sufficient for data recovery. So, for all positions, it holds that at least one of the two scans will yield a recovered sync block.

We use the aforementioned speeds in combination with a special allocation scheme for trick-play data. This scheme is such that the contents of each trick-play sync block is repeated at the same position in a number of subsequent tracks of equal azimuth. Thanks to this particular duplication of the data in a given sync block, we can guarantee that it will be recovered at the appropriate tape speed, without any additional tracking requirements. In general, a tape speed of $m \times \frac{2}{7}$ can be used if the number of copies is at least $m$. This follows from the fact that the group of tracks containing $m$ copies is scanned twice at $\frac{m}{2} \times$. From the above-mentioned complementarity of subsequent scans, it then follows that at least one copy will be recovered.

The above trick-play regime is robust for track non-linearities and scan-path non-linearities, and
it will work for different scanner configurations. This is due to the complementarity property, which holds irrespective of the scan path angles. In general, data from a trick-play stream is allocated at several consecutive sync block positions per track. Copies of sync blocks at different positions in the tracks may be recovered in different scans. Therefore the results of several scans have to be collected and sorted in the recorder, in order to regenerate the complete stream with all data in the correct order.

Figure 6 schematically represents the complete trick-play data allocation. We use two dedicated trick-play streams for +3.5× and +8.5×. The trick-play data from these streams are mapped into reserved groups of sync blocks in the tracks of one azimuth. The sync block contents of these groups are copied 7 times for 3.5× data and 17 times for 8.5× data. This is illustrated in Figure 6, where a set of copies has been highlighted in black for each speed.

Each trick-play stream is a valid MPEG transport stream. The mapping of transport packets is the same as for the normal-play stream, including the use of special time stamps. At trick play, a similar reference time-base is generated, again in relation to the servo reference in order to be in accordance with the actual tape speed. As a result, the trick-play processing at playback is almost identical to that of normal-play. This is in contrast to [6], where a similar allocation scheme is used. The track format in [6] has a single trick-play stream, which is intended for trick-play at several speeds. In order to deliver a valid MPEG stream at each speed in this case, considerable processing is required at playback. Moreover, it implies that the same picture data is used for the output at both low and high speeds, so that the output quality at low speeds suffers from a low picture refresh rate. In our solution, we can optimize low and high-speed quality independently, because separate picture data is available.

4. Channel coding

4.1 Channel modulation and ECC

We have adopted the high-efficiency 24-25 channel modulation [4] [7] of the DVC system to construct recording data with a DC-free spectrum, while generating digitally embedded pilot tones to allow automatic or dynamic track following. In [7], it was indicated that besides the aforementioned properties, the runlength of bits can be limited and nearly unique sync words can be obtained for synchronizing on data packet level. The data packets on the recording channel are of fixed length (90 Bytes) and referred to as sync blocks, to avoid confusion with the MPEG transport packets of Section 3.

The DVC track format is divided into the ITI, audio, video and subcode sectors, as shown in Fig. 7. Note that the terms ‘video’ and ‘audio’ stand for the data streams as referred to in the DVC format description [5] and have no relation to the MPEG stream. Each sector in the format is based on a sequence of sync blocks, and the size of the individual sectors is different.

Error-correction coding (ECC) of the video and audio sectors is based on intra-track Reed-Solomon (RS) product codes. This means that the sectors are treated in the ECC as two-dimensional blocks of data. The rows of such blocks are both in size and format equal to the sync blocks mentioned previously. The ECC from the DVC format processes rows (code C1) and columns (code C2) from a sector. The C1 code protects individual sync blocks, primarily against random errors. It is the same for both the audio and video sector, and denoted as RS(85,77). C2, the second RS code, protects a complete sector, primarily against burst errors spanning a few sync blocks. The C2 code for the video sector is RS(149,138), while that for the audio sector is RS(14,9).

In our experimental MPEG recorder, both the normal- and trick-play data will be placed inside the DVC video sector. In principle, the DVC video ECC can improve a raw symbol error rate of random errors of $10^{-3}$ to a corrected symbol error rate of less than $10^{-50}$. This applies when the C1 decoder is set to correct a maximum of 3 errors and flags all uncorrectable C1 codewords for erasure correction by the C2 decoder, set for its
maximum correction capability \((t \text{ errors} + e \text{ erasures} \leq 11)\). Against burst errors, it can withstand vertical scratches of 1.8 mm width, horizontal scratches of 0.29 mm width, circular defects of 1.8 mm diameter and helical errors (in the direction of a track) up to 6.7\% of the length of the video sector within a track.

4.2 Enhanced ECC for MPEG data

Although the DVC video ECC provides a good protection against random errors, it is rather susceptible to long helical errors which can occur in case of head clogging. In such a case, sync block-based concealment can be employed for DVC playback, but this approach cannot be pursued for MPEG data, due to the higher compression ratio and lack of alignment between the elementary MPEG data and the sync blocks of the DVC tape format.

For this reason, an extra error-correction code C3 has been defined, in which a Reed-Solomon codeword is formed by interleaving in a direction across the tracks, thereby distributing errors within a track uniformly into different codewords. Each codeword thus carries a reduced number of errors with an increased chance of error correction. In our experimental MPEG recorder, the C3 code applies to a data block of \(N\) tracks, with track interleaving defined on a modulo-\(N\)-track basis. The MPEG data and the C3 parities are recorded in the video data area of the track format, as is indicated in Figure 8.

To cater for different channel behaviour and requirements, we have designed a flexible hardware architecture in which the number of C3 parities, \(N\), and the interleaving algorithm are configurable. Currently, the C3 code is RS(135,125), applied to \(N = 12\) tracks, as shown in Fig. 8. It can recover all data as long as less than 88\% of the MPEG sector in one of the 12 tracks is corrupted by e.g. head clogging or editing. In that case the C2 decoder will flag all uncorrectable symbols for erasure correction by the C3 decoder.

5. Recorder mechanics

5.1 Mechanism

The channel-coded signals are stored on a 1/4 inch ME tape by means of an experimental compact recorder mechanism based on the DVC standard (see Figure 9). This mechanism has been constructed in our laboratory based on a scanner concept with 2 \(\times\) 2 heads at 9000 rpm, allowing for a maximum data rate of 50 Mbit/s. This results in a total channel bit rate of \(2 \times 41.85\) Mbit/s, leading to a bit length of 0.25 \(\mu\)m with the drum diameter of 21.7 mm. Two heads of opposite azimuth with a relative height equal to the track pitch of 10 \(\mu\)m form one head pair. The relative position of the two headpairs, A and B, has been indicated in Figure 1.

The dual channel, dual head pair approach has been adopted for two reasons. Firstly, the highest data rate of 50 Mbit/s required for DVC HDTV recording, can be achieved with a small drum diameter. Secondly, the dual channel approach enables all integral dividers of the highest bit rate without any restrictions due to the azimuth recording. Attractive divider data rates are 25 Mbit/s for ATV in the USA, as well as 12.5
5.2 Interval recording

At the highest data rate of 50 Mbit/s, both head pairs are used continuously. Lower bit rates are realized with interval recording based on head pairs to ensure the correct azimuth of the recorded tracks. The effective bit rate is halved by using only one head pair and again halved by using this pair only once in two revolutions etc. The tape speed is lowered with the same ratio, leading to a proportional increase in playing time.

In Figure 10a–d the record current as a function of time, in one channel, is depicted for four different effective data rates with dividers 1, 2, 4 and 8, respectively. Resulting track pair numbers are shown below the recording packets. In all cases, two recording channels with a fixed recording channel bit rate of 41.85 Mbit/s are used. The timing of the second recording channel is essentially the same, apart from a slight offset in time to compensate for the distance between the two heads forming one pair. It can be clearly seen that the effective rate is lowered by using fewer active time intervals. Each of these intervals is equal to half a revolution of the scanner (1/300 second), synchronized with the heads being in contact with the tape. For a regular track pattern on the tape it is necessary that the active intervals are equally distributed on the time axis. As can be seen from Figure 10, head pair B can be discarded if 50 Mbit/s is not needed. However, as will be explained in the subsequent section, this head pair can be used to increase robustness during replay. For the sake of completeness, it is mentioned here that also the odd dividers 3, 5, 7 etc. can easily be accomplished. The system in this paper uses one in every four time intervals, as in Figure 10c.

5.3 Interval replay

During replay, the same interval method can be used. However, the remaining time intervals are also available for reading data from tape. In normal play only one interval leads to the correct tracking position of the heads with respect to the tracks, resulting in the maximum signal amplitude and SNR. This is depicted in Figure 11. In the system described here, the other three time intervals result in tracking errors of +1/2, +1 and -1/2 track, respectively. As a result, the use of these intervals seems unattractive. Still, from
Figure 11c it can be seen that one track is read several times with different signal levels. The figure is only valid for nominal tracking conditions. In case of a tracking error, the signal level of the main scan is reduced and of the preceding or following scan is increased, depending on the direction of the error. The turnover point at which both levels are equal is reached for a tracking error of 1/4 track. If the width of the head is equal to the track pitch, this level is 75% of the nominal value. In the usual case, with the head width wider than the track, this can be as high as 95%. The minimum signal level of all data, irrespective of the tracking accuracy, is equal to the level at the turnover point. Consequently, a combination of the results of the different intervals after the first level of ECC (C1) allows for larger tracking errors. Effectively, there are no longer restrictions on the local or global tracking accuracy, leading to a dramatic increase in robustness which is especially important for MPEG signals. In the experimental setup, the use of 1/4, 1/2 or all the time intervals can be selected during replay. In trick play, the use of this technique is also advantageous.

6. Conclusions

We have presented an experimental recording system based on the DVC recording standard which is capable of recording MPEG signals. The system approach adopted for trick play is to record separate data streams for individual speeds with a special data allocation. This allows for dedicated MPEG streams per speed and it imposes minimal tracking conditions on the recording mechanism. The robustness of the system has been improved by an additional error correction code which is interleaved in a direction across the tracks, so that long burst errors due to head clogging can still be handled. For accurate timing of MPEG signals at playback, we have introduced a time stamp mechanism, featuring a reference time base which is related to the tape contents. The time stamps serve to reconstruct the original input timing. The data rate of the experimental recording system is 12.5 Mbit/s. However, the recording mechanism supports several alternative data rates. These rates are dividers of the maximum data rate.

Acknowledgements

We gratefully acknowledge several colleagues of Philips Research Laboratories, in particular Ing. O. Eerenberg, Ing. F. Jorritsma, Ing. M. Looykens, and Ing. A. Palm.

References

Biographies

Ronald W.J.J. Saeijs was born in Maastricht, The Netherlands, in 1962. In 1986 he received the M.S.E.E. degree from the University of Technology in Eindhoven. In 1986 he joined Philips Research Laboratories Eindhoven, where he did research on asynchronous VLSI design methods and silicon compilation for a number of years. In 1992 he became a member of the Magnetic Recording Systems Department, where he is involved in system research on helical-scan recording for MPEG applications.

Peter H.N. de With was born in Lexmond, The Netherlands, in 1958. He graduated in electrical engineering from the University of Technology in Eindhoven. He joined Philips Research Laboratories Eindhoven in 1984, where he became a member of the Magnetic Recording Systems Department. In 1992 he received the Ph.D. degree from the University of Technology Delft, The Netherlands, for his work on video bit-rate reduction for recording applications. From 1985 to 1988 he participated in a European research project on digital video recording. Since 1988 he has been involved in the European RACE projects 1001 and 2026 in which digital HDTV and data recording is studied.

Albert M.A. Rijckaert was born in Eindhoven, The Netherlands, in 1945. He received the Ing. degree in electrical engineering from the Polytechnic College in Eindhoven in 1966. He joined Philips Research Laboratories Eindhoven in 1968 and has been a member of the Magnetic Recording Systems Department since then. He has worked on various subjects related to magnetic recording, such as the physical recording process, and tracking and skew correction systems for linear recording. Since 1982, he is involved in system research on helical-scan recording. Mr. Rijckaert has participated in several European projects on digital video recording.

Calto Wong was born in Hong Kong, in 1964. He received the B.S.E.E. degree from the University of Hong Kong in 1986 and the M.S.E.E. degree from the Eindhoven International Institute, The Netherlands, in 1988. In 1988, he joined Philips Research Laboratories in Eindhoven, The Netherlands, as a member of the Magnetic Recording Systems Department. Since 1991 he has been involved in the design and implementation of the error-correction and digital interface systems for digital video recording applications. He has taken part in the European RACE 1001 and 2026 projects on digital video and data recording.