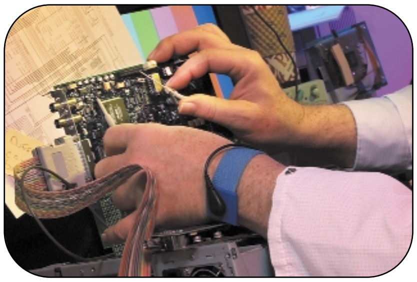
MPEG Tutorial

A Guide to MPEG Fundamentals and Protocol Analysis

(Including DVB and ATSC)



COMPUTING

COMMUNICATIONS

VIDEO

Tektronix

Enabling Innovation

Section 1 – Introduction to MPEG

MPEG is one of the most popular audio/video compression techniques because it is not just a single standard. Instead, it is a range of standards suitable for different applications but based on similar principles. MPEG is an acronym for the Moving Picture Experts Group, part of the Joint Technical Committee, JTC1, established by the ISO (International Standards Organization) and IEC (International Electrotechnical Commission). JTC1 is responsible for Information Technology; within JTC1, Sub Group SG29 is responsible for "Coding of Audio, Picture, and Multimedia and Hypermedia Information." There are a number of working groups within SG29, including JPEG (Joint Photographic Experts Group), and Working Group 11 for compression of moving pictures. ISO/IEC JTC1/SG29/WG11 is MPEG.

MPEG can be described as the interaction of acronyms. As ETSI stated, "The CAT is a pointer to enable the IRD to find the EMMs associated with the CA system(s) that it uses." If you can understand that sentence you don't need this book!

1.1 Convergence

Digital techniques have made rapid progress in audio and video for a number of reasons. Digital information is more robust and can be coded to substantially eliminate error. This means that generation-losses in recording and losses in transmission may be eliminated. The compact disc (CD) was the first consumer product to demonstrate this.

While the CD has an improved sound quality with respect to its vinyl predecessor, comparison of quality alone misses the point. The real point is that digital recording and transmission techniques allow content manipulation to a degree that is impossible with analog. Once audio or video is digitized, the content is in the form of data. Such data can be handled in the same way as any other kind of data; therefore, digital video and audio become the province of computer technology.

The convergence of computers and audio/video is an inevitable consequence of the key inventions of computing and pulse code modulation (PCM). Digital media can store any type of information, so it is easy to use a computer storage device for digital video. The nonlinear workstation was the first example of an application of convergent technology that did not have an analog forerunner. Another example, multimedia, combines the storage of audio, video, graphics, text and data on the same medium. Multimedia has no equivalent in the analog domain.

1.2 Why Compression Is Needed

The initial success of digital video was in post-production applications, where the high cost of digital video was offset by its limitless layering and effects capability. However, production-standard digital video generates over 200 megabits per second of data, and this bit rate requires extensive capacity for storage and wide bandwidth for transmission. Digital video could only be used in wider applications if the storage and bandwidth requirements could be eased; this is the purpose of compression.

Compression is a way of expressing digital audio and video by using less data. Compression has the following advantages:

- A smaller amount of storage is needed for a given amount of source material.
- When working in real time, compression reduces the bandwidth needed. Additionally, compression allows faster-than-real-time transfer between media, for example, between tape and disk.
- A compressed recording format can use a lower recording density and this can make the recorder less sensitive to environmental factors and maintenance.

1.3 Principles of Compression

There are two fundamentally different techniques that may be used to reduce the quantity of data used to convey information content. In practical compression systems, these are usually combined, often in very complex ways.

The first technique is to improve coding efficiency. There are many ways of coding any given information, and most simple data representations of video and audio contain a substantial amount of redundancy. The concept of *entropy* is discussed below.

Many coding tricks can be used to reduce or eliminate redundancy; examples include *run-length coding* and variable-length coding systems such as Huffman codes. When properly used, these techniques are completely reversible so that after decompression the data is identical to that at the input of the system. This type of compression is known as *lossless*. Archiving computer programs such as PKZip employ lossless compression.

Obviously, lossless compression is ideal, but unfortunately it does not usually provide the degree of data reduction needed for video and audio applications. However, because it is lossless, it can be applied at any point in the system and is often used on the data output of lossy compressors.

If the elimination of redundancy does not reduce the data as much as needed, some information will have to be discarded. Lossy compression systems achieve data reduction by removing information that is irrelevant, or of lesser relevance. These are not general techniques that can be applied to any data stream; the assessment of relevance can only be made in the context of the application, understanding what the data represents and how it will be used. In the case of television, the application is the presentation of images and sound to the human visual and hearing systems, and the human factors must be well understood to design an effective compression system.

Some information in video signals cannot be perceived by the human visual system and is, therefore, truly irrelevant in this context. A compression system that discards only irrelevant image information is known as *visually lossless.*

1.4 Compression in Television Applications

Television signals, analog or digital, have always represented a great deal of information, and bandwidth reduction techniques have been used from a very early stage. Probably the earliest example is interlace. For a given number of lines, and a given rate of picture refresh, interlace offers a 2:1 reduction in bandwidth requirement. The process is lossy; interlace generates artifacts caused by interference between vertical and temporal information, and reduces the usable vertical resolution of the image. Nevertheless, most of what is given up is largely irrelevant, so interlace represented a simple and very valuable trade-off in its time. Unfortunately interlace and the artifacts it generates are very disruptive to more sophisticated digital compression schemes. Much of the complexity of MPEG-2 results from the need to handle interlaced signals, and there is still a significant loss in coding efficiency when compared to progressive signals.

The next major steps came with the advent of color. Color cameras produce GBR signals, so nominally there is three times the information of a monochrome signal – but there was a requirement to transmit color signals in the same channels used for monochrome.

The first part of the solution was to transform the signals from *GBR* to a brightness signal (normally designated *Y*) plus two color difference signals, *U* and *V*, or *I* and *Q*. Generation of a brightness signal went a long way towards solving the problem of compatibility with monochrome receivers, but the important step for bandwidth minimization came from the color difference signals.

It turns out that the human visual system uses sensors that are sensitive to brightness, and that can “see” a very high-resolution image. Other sensors capture color information, but at much lower resolution. The net result is that, within certain limits, a sharp monochrome image representing scene brightness overlaid with fuzzy (low-bandwidth) color information will appear as a sharp color picture. It is not possible to take advantage of this when dealing with *GBR* signals, as each signal contains both brightness and color information. However, in *YUV* space, most of the brightness information is carried in the *Y* signal, and very little in the color difference signals. So, it is possible to filter the color difference signals and drastically reduce the information to be transmitted.

This is an example of eliminating (mostly) irrelevant information. Under the design viewing conditions, the human visual system does not respond significantly to the high frequency information in the color difference signals, so it may be discarded. NTSC television transmissions carry only about 500 kHz in each color difference signal, but the pictures are adequately sharp for many applications.

The final step in the bandwidth reduction process of NTSC and PAL was to “hide” the color difference signals in unused parts of the spectrum of the monochrome signal. Although the process is not strictly lossless, this can be thought of as increasing the coding efficiency of the signal.

Some of the techniques in the digital world are quite different, but similar principles apply. For example, MPEG transforms signals into a different domain to permit the isolation of irrelevant information. The transform to color-difference space is still employed, but digital techniques permit filtering of the color difference signal to reduce vertical resolution for further savings.

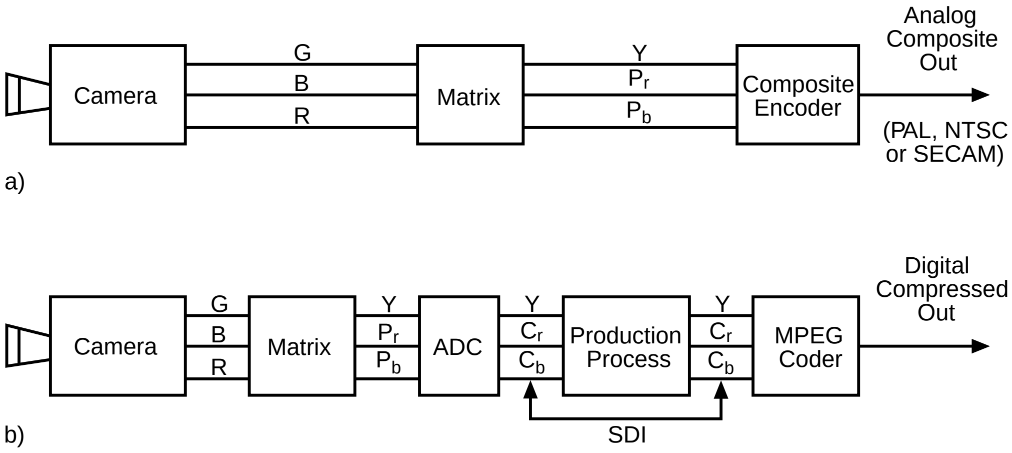


Figure 1-1.

Figure 1-1a shows that in traditional television systems, the GBR camera signal is converted to Y, P_b, P_r components for production and encoded into analog composite for transmission. Figure 1-1b shows the modern equivalent. The Y, P_b, P_r signals are digitized and carried as Y, C_b, C_r signals in SDI form through the production process prior to being encoded with MPEG for transmission. Clearly, MPEG can be considered by the broadcaster as a more efficient replacement for composite video. In addition, MPEG has greater flexibility because the bit rate required can be adjusted to suit the application. At lower bit rates and resolutions, MPEG can be used for video conferencing and video telephones.

Digital Video Broadcasting (DVB) and Advanced Television Systems Committee (ATSC) (the European- and American-originated digital-television broadcasting standards) would not be viable without compression because the bandwidth required would be too great. Compression extends the playing time of DVD (digital video/versatile disk) allowing full-length movies on a single disk. Compression also reduces the cost of ENG and other contributions to television production. DVB, ATSC and digital video disc (DVD) are all based on MPEG-2 compression.

In tape recording, mild compression eases tolerances and adds reliability in Digital Betacam and Digital-S, whereas in SX, DVC, DVCPRO and DVCAM, the goal is miniaturization. In disk-based video servers, compression lowers storage cost. Compression also lowers bandwidth, which allows more users to access a given server. This characteristic is also important for VOD (video on demand) applications.

1.5 Introduction to Digital Video Compression

In all real program material, there are two types of components of the signal: those that are novel and unpredictable and those that can be anticipated. The novel component is called entropy and is the true information in the signal. The remainder is called redundancy because it is not essential. Redundancy may be spatial, as it is in large plain areas of picture where adjacent pixels have almost the same value. Redundancy can also be temporal as it is where similarities between successive pictures are used. All compression systems work by separating entropy from redundancy in the encoder. Only the entropy is recorded or transmitted and the decoder computes the redundancy from the transmitted signal. Figure 1-2a (see next page) shows this concept.

An ideal encoder would extract all the entropy and only this will be transmitted to the decoder. An ideal decoder would then reproduce the original signal. In practice, this ideal cannot be reached. An ideal coder would be complex and cause a very long delay in order to use temporal redundancy. In certain applications, such as recording or broadcasting, some delay is acceptable, but in videoconferencing it is not. In some cases, a very complex coder would be too expensive. It follows that there is no one ideal compression system.

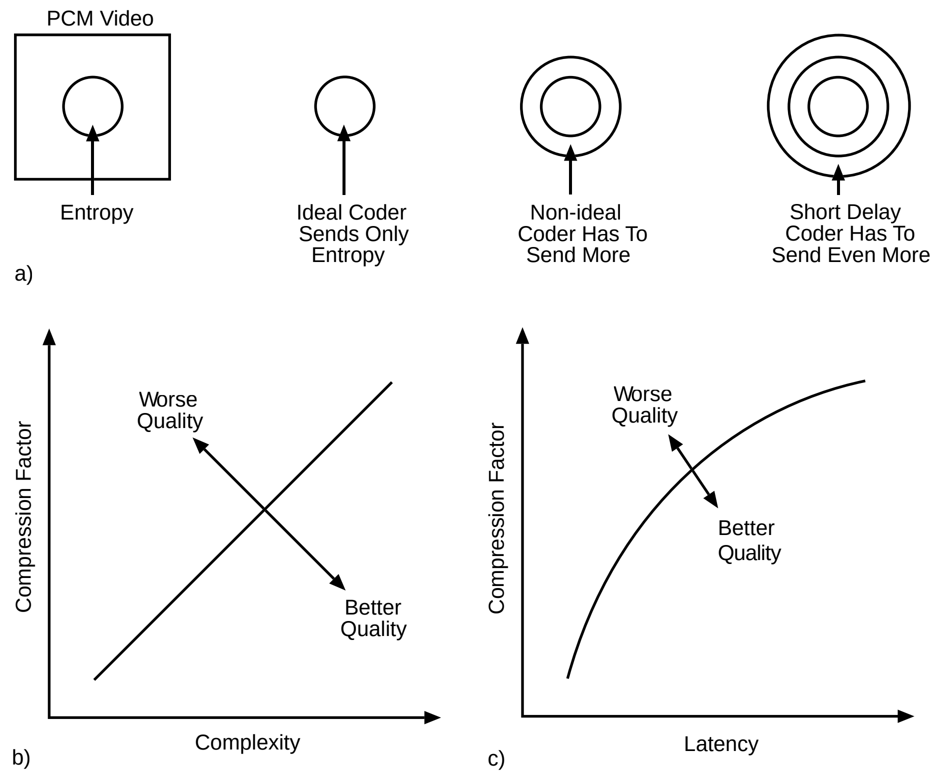


Figure 1-2.

In practice, a range of coders is needed which have a range of processing delays and complexities. The power of MPEG is that it is not a single compression format, but a range of standardized coding tools that can be combined flexibly to suit a range of applications. The way in which coding has been performed is included in the compressed data so that the decoder can automatically handle whatever the coder decided to do.

In MPEG-2 and MPEG-4 coding is divided into several profiles that have different complexity, and each profile can be implemented at a different level depending on the resolution of the input picture. Section 4 considers profiles and levels in detail.

There are many different digital video formats and each has a different bit rate. For example a high definition system might have six times the bit rate of a standard definition system. Consequently, just knowing the bit rate out of the coder is not very useful. What matters is the compression factor, which is the ratio of the input bit rate to the compressed bit rate, for example 2:1, 5:1 and so on.

Unfortunately, the number of variables involved makes it very difficult to determine a suitable compression factor. Figure 1-2a shows that for an ideal coder, if all of the entropy is sent, the quality is good. However, if the compression factor is increased in order to reduce the bit rate, not all of the entropy is sent and the quality falls. Note that in a compressed system when the quality loss occurs, it is steep (Figure 1-2b). If the available bit rate is inadequate, it is better to avoid this area by reducing the entropy of the input picture. This can be done by filtering. The loss of resolution caused by the filtering is subjectively more acceptable than the compression artifacts.

To identify the entropy perfectly, an ideal compressor would have to be extremely complex. A practical compressor may be less complex for economic reasons and must send more data to be sure of carrying all of the entropy. Figure 1-2b shows the relationship between coder complexity and performance. The higher the compression factor required, the more complex the encoder has to be.

The entropy in video signals varies. A recording of an announcer delivering the news has much redundancy and is easy to compress. In contrast, it is more difficult to compress a recording with leaves blowing in the wind or one of a football crowd that is constantly moving and therefore has less redundancy (more information or entropy). In either case, if all the entropy is not sent, there will be quality loss. Thus, we may choose between a constant bit-rate channel with variable quality or a constant quality channel with variable bit rate. Telecommunications network operators tend to prefer a constant bit rate for practical purposes, but a buffer memory can be used to average out entropy variations if the resulting increase in delay is acceptable. In recording, a variable bit rate may be easier to handle and DVD uses variable bit rate, using buffering so that the average bit rate remains within the capabilities of the disk system.

Intra-coding (intra = within) is a technique that exploits spatial redundancy, or redundancy within the picture; inter-coding (inter = between) is a technique that exploits temporal redundancy. Intra-coding may be used alone, as in the JPEG standard for still pictures, or combined with inter-coding as in MPEG.

Intra-coding relies on two characteristics of typical images. First, not all spatial frequencies are simultaneously present, and second, the higher the spatial frequency, the lower the amplitude is likely to be. Intra-coding requires analysis of the spatial frequencies in an image. This analysis is the purpose of transforms such as wavelets and DCT (discrete cosine transform). Transforms produce coefficients that describe the magnitude of each spatial frequency. Typically, many coefficients will be zero, or nearly zero, and these coefficients can be omitted, resulting in a reduction in bit rate.

Inter-coding relies on finding similarities between successive pictures. If a given picture is available at the decoder, the next picture can be created by sending only the picture differences. The picture differences will be increased when objects move, but this magnification can be offset by using motion compensation, since a moving object does not generally change its appearance very much from one picture to the next. If the motion can be measured, a closer approximation to the current picture can be created by shifting part of the previous picture to a new location. The shifting process is controlled by a pair of horizontal and vertical displacement values (collectively known as the *motion vector*) that is transmitted to the decoder. The motion vector transmission requires less data than sending the picture-difference data.

MPEG can handle both interlaced and non-interlaced images. An image at some point on the time axis is called a “picture,” whether it is a field or a frame. Interlace is not ideal as a source for digital compression because it is in itself a compression technique. Temporal coding is made more complex because pixels in one field are in a different position to those in the next.

Motion compensation minimizes but does not eliminate the differences between successive pictures. The picture difference is itself a spatial image and can be compressed using transform-based intra-coding as previously described. Motion compensation simply reduces the amount of data in the difference image.

The efficiency of a temporal coder rises with the time span over which it can act. Figure 1-2c shows that if a high compression factor is required, a longer time span in the input must be considered and thus a longer coding delay will be experienced. Clearly, temporally coded signals are difficult to edit because the content of a given output picture may be based on image data which was transmitted some time earlier. Production systems will have to limit the degree of temporal coding to allow editing and this limitation will in turn limit the available compression factor.

1.6 Introduction to Audio Compression

The bit rate of a PCM digital audio channel is only about 1.5 megabits per second, which is about 0.5% of 4:2:2 digital video. With mild video compression schemes, such as Digital Betacam, audio compression is unnecessary. But, as the video compression factor is raised, it becomes important to compress the audio as well.

Audio compression takes advantage of two facts. First, in typical audio signals, not all frequencies are simultaneously present. Second, because of the phenomenon of masking, human hearing cannot discern every detail of an audio signal. Audio compression splits the audio spectrum into bands by filtering or transforms, and includes less data when describing bands in which the level is low. Where masking prevents or reduces audibility of a particular band, even less data needs to be sent.

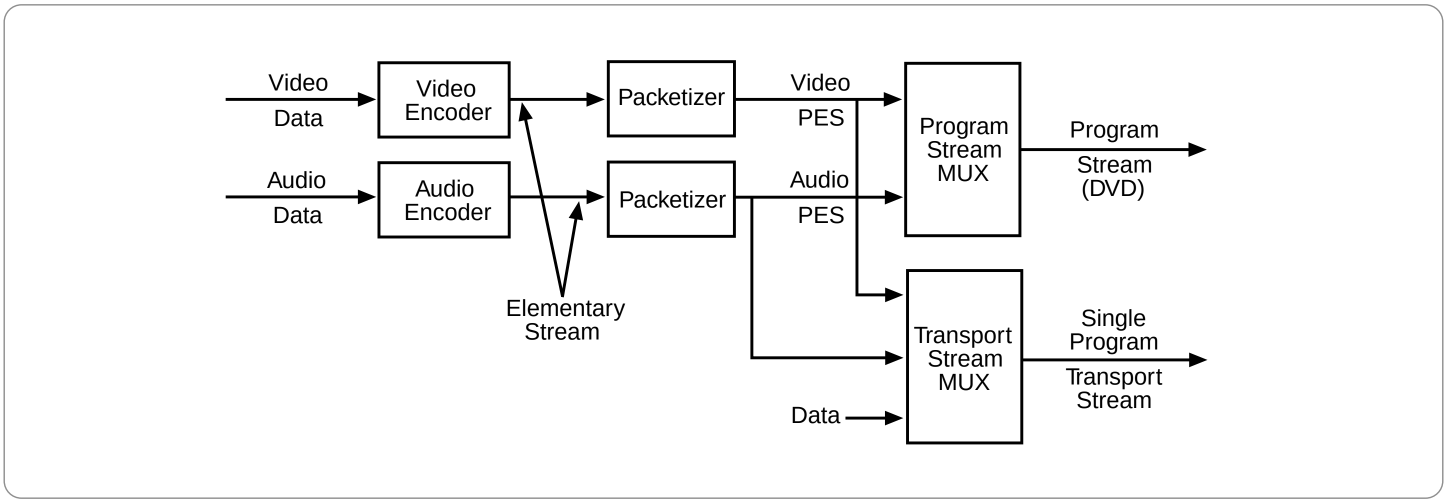


Figure 1-3.

Audio compression is not as easy to achieve as video compression because of the acuity of hearing. Masking only works properly when the masking and the masked sounds coincide spatially. Spatial coincidence is always the case in mono recordings but not in stereo recordings, where low-level signals can still be heard if they are in a different part of the sound stage. Consequently, in stereo and surround sound systems, a lower compression factor is allowable for a given quality. Another factor complicating audio compression is that delayed resonances in poor loudspeakers actually mask compression artifacts. Testing a compressor with poor speakers gives a false result, and signals that are apparently satisfactory may be disappointing when heard on good equipment.

1.7 MPEG Streams

The output of a single MPEG audio or video coder is called an elementary stream. An elementary stream is an endless near real-time signal. For convenience, the elementary stream may be broken into data blocks of manageable size, forming a packetized elementary stream (PES). These data blocks need header information to identify the start of the packets and must include time stamps because packetizing disrupts the time axis.

Figure 1-3 shows that one video PES and a number of audio PES can be combined to form a program stream, provided that all of the coders are locked to a common clock. Time stamps in each PES can be used to ensure lip-sync between the video and audio. Program streams have variable-length packets with headers. They find use in data transfers to and from optical and hard disks, which are essentially error free, and in which files of arbitrary sizes are expected. DVD uses program streams.

For transmission and digital broadcasting, several programs and their associated PES can be multiplexed into a single transport stream. A transport stream differs from a program stream in that the PES packets are further subdivided into short fixed-size packets and in that multiple programs encoded with different clocks can be carried. This is possible because a transport stream has a program clock reference (PCR) mechanism that allows transmission of multiple clocks, one of which is selected and regenerated at the decoder. A single program transport stream (SPTS) is also possible and this may be found between a coder and a multiplexer. Since a transport stream can genlock the decoder clock to the encoder clock, the SPTS is more common than the Program Stream.

A transport stream is more than just a multiplex of audio and video PES. In addition to the compressed audio, video and data, a transport stream includes metadata describing the bit stream. This includes the program association table (PAT) that lists every program in the transport stream. Each entry in the PAT points to a program map table (PMT) that lists the elementary streams making up each program. Some programs will be open, but some programs may be subject to conditional access (encryption) and this information is also carried in the metadata.

The transport stream consists of fixed-size data packets, each containing 188 bytes. Each packet carries a program identifier code (PID). Packets in the same elementary stream all have the same PID, so that the decoder (or a demultiplexer) can select the elementary stream(s) it wants and reject the remainder. Packet continuity counts ensure that every packet that is needed to decode a stream is received. An effective synchronization system is needed so that decoders can correctly identify the beginning of each packet and deserialize the bit stream into words.

1.8 Need for Monitoring and Analysis

The MPEG transport stream is an extremely complex structure using interlinked tables and coded identifiers to separate the programs and the elementary streams within the programs. Within each elementary stream, there is a complex structure, allowing a decoder to distinguish between, for example, vectors, coefficients and quantization tables.

Failures can be divided into two broad categories. In the first category, the transport system correctly delivers information from an encoder/multiplexer to a decoder with no bit errors or added jitter, but the encoder/multiplexer or the decoder has a fault. In the second category, the encoder/multiplexer and decoder are fine, but the transport of data from one to the other is defective. It is very important to know whether the fault lies in the encoder/multiplexer, the transport or the decoder if a prompt solution is to be found.

Synchronizing problems, such as loss or corruption of sync patterns, may prevent reception of the entire transport stream. Transport stream protocol defects may prevent the decoder from finding all of the data for a program, perhaps delivering picture but not sound. Correct delivery of the data but with excessive jitter can cause decoder timing problems.

If a system using an MPEG transport stream fails, the fault could be in the encoder, the multiplexer or in the decoder. How can this fault be isolated? First, verify that a transport stream is compliant with the MPEG-coding standards. If the stream is not compliant, a decoder can hardly be blamed for having difficulty. If the stream is compliant, the decoder may need attention.

Traditional video testing tools, the signal generator, the waveform monitor and vectorscope, are not appropriate in analyzing MPEG systems, except to ensure that the video signals entering and leaving an MPEG system are of suitable quality. Instead, a reliable source of valid MPEG test signals is essential for testing receiving equipment and decoders. With a suitable analyzer, the performance of encoders, transmission systems, multiplexers and remultiplexers can be assessed with a high degree of confidence. As a long standing supplier of high grade test equipment to the video industry, Tektronix continues to provide test and measurement solutions as the technology evolves, giving the MPEG user the confidence that complex compressed systems are correctly functioning and allowing rapid diagnosis when they are not.

1.9 Pitfalls of Compression

MPEG compression is lossy in that what is decoded is not identical to the original. The entropy of the source varies, and when entropy is high, the compression system may leave visible artifacts when decoded. In temporal compression, redundancy between successive pictures is assumed. When this is not the case, the system may fail. An example is video from a press conference where flashguns are firing. Individual pictures containing the flash are totally different from their neighbors, and coding artifacts may become obvious.

Irregular motion or several independently moving objects on screen require a lot of vector bandwidth and this requirement may only be met by reducing the bandwidth available for picture-data. Again, visible artifacts may occur whose level varies and depends on the motion. This problem often occurs in sports-coverage video.

Coarse quantizing results in luminance contouring and posterized color. These can be seen as blotchy shadows and blocking on large areas of plain color. Subjectively, compression artifacts are more annoying than the relatively constant impairments resulting from analog television transmission systems.

The only solution to these problems is to reduce the compression factor. Consequently, the compression user has to make a value judgment between the economy of a high compression factor and the level of artifacts.

In addition to extending the encoding and decoding delay, temporal coding also causes difficulty in editing. In fact, an MPEG bit stream cannot be arbitrarily edited. This restriction occurs because, in temporal coding, the decoding of one picture may require the contents of an earlier picture and the contents may not be available following an edit. The fact that pictures may be sent out of sequence also complicates editing.

If suitable coding has been used, edits can take place, but only at splice points that are relatively widely spaced. If arbitrary editing is required, the MPEG stream must undergo a decode-modify-recode process, which will result in generation loss.

Section 2 – Compression in Video

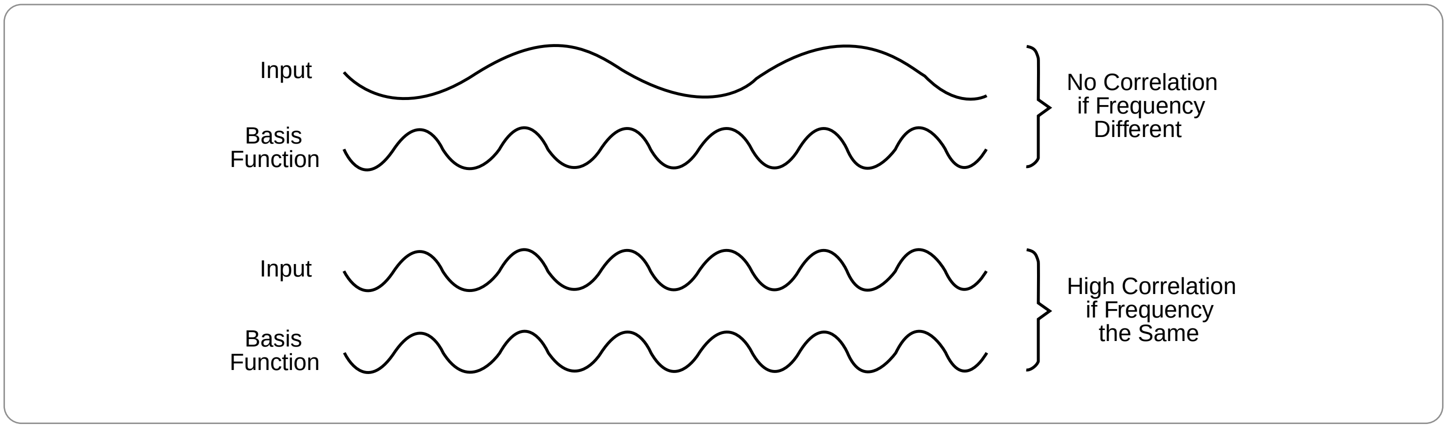

Input
Basis Function
No Correlation if Frequency Different
Input
Basis Function
High Correlation if Frequency the Same

Figure 2-1.

This section shows how video compression is based on the perception of the eye. Important enabling techniques, such as transforms and motion compensation, are considered as an introduction to the structure of an MPEG coder.

2.1 Spatial or Temporal Coding?

As was seen in Section 1, video compression can take advantage of both spatial and temporal redundancy. In MPEG, temporal redundancy is reduced first by using similarities between successive pictures. As much as possible of the current picture is created or “predicted” by using information from pictures already sent. When this technique is used, it is only necessary to send a difference picture, which eliminates the differences between the actual picture and the prediction. The difference picture is then subject to spatial compression. As a practical matter it is easier to explain spatial compression prior to explaining temporal compression.

Spatial compression relies on similarities between adjacent pixels in plain areas of picture and on dominant spatial frequencies in areas of patterning. The JPEG system uses spatial compression only, since it is designed to transmit individual still pictures. However, JPEG may be used to code a succession of individual pictures for video. In the so-called “Motion JPEG” application, the compression factor will not be as good as if temporal coding was used, but the bit stream will be freely editable on a picture-by-picture basis.

2.2 Spatial Coding

The first step in spatial coding is to perform an analysis of spatial frequencies using a transform. A transform is simply a way of expressing a waveform in a different domain, in this case, the frequency domain. The output of a transform is a set of coefficients that describe how much of a given frequency is present. An inverse transform reproduces the original waveform. If the coefficients are handled with sufficient accuracy, the output of the inverse transform is identical to the original waveform.

The most well known transform is the Fourier transform. This transform finds each frequency in the input signal. It finds each frequency by multiplying the input waveform by a sample of a target frequency, called a basis function, and integrating the product. Figure 2-1 shows that when the input waveform does not contain the target frequency, the integral will be zero, but when it does, the integral will be a coefficient describing the amplitude of that component frequency.

The results will be as described if the frequency component is in phase with the basis function. However if the frequency component is in quadrature with the basis function, the integral will still be zero. Therefore, it is necessary to perform two searches for each frequency, with the basis functions in quadrature with one another so that every phase of the input will be detected.

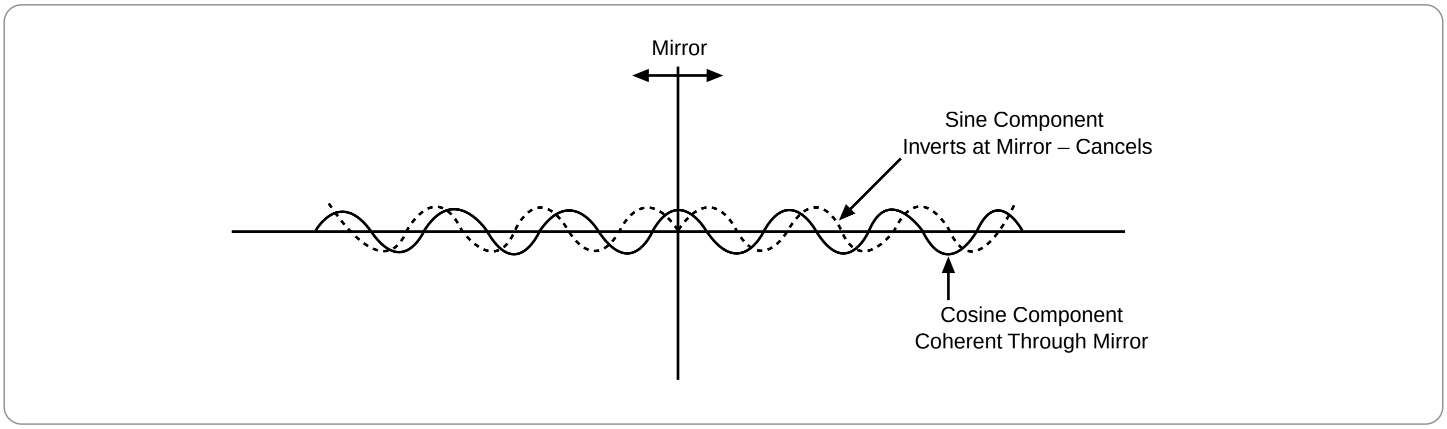


Figure 2-2.

The Fourier transform has the disadvantage of requiring coefficients for both sine and cosine components of each frequency. In the cosine transform, the input waveform is time-mirrored with itself prior to multiplication by the basis functions. Figure 2-2 shows that this mirroring cancels out all sine components and doubles all of the cosine components. The sine basis function is unnecessary and only one coefficient is needed for each frequency.

The discrete cosine transform (DCT) is the sampled version of the cosine transform and is used extensively in two-dimensional form in MPEG. A block of 8x8 pixels is transformed to become a block of 8x8 coefficients. Since the transform requires multiplication by fractions, there is wordlength extension, resulting in coefficients that have longer wordlength than the pixel values. Typically an 8-bit pixel block results in an 11-bit coefficient block. Thus, a DCT does not result in any compression; in fact it results in the opposite. However, the DCT converts the source pixels into a form where compression is easier.

Figure 2-3 shows the results of an inverse transform of each of the individual coefficients of an 8x8 DCT. In the case of the luminance signal, the top-left coefficient is the average brightness or DC component of the whole block. Moving across the top row, horizontal spatial frequency increases. Moving down the left column, vertical spatial frequency increases. In real pictures, different vertical and horizontal spatial frequencies may occur simultaneously and a coefficient at some point within the block will represent all possible horizontal and vertical combinations.

Figure 2-3 also shows 8 coefficients as one-dimensional horizontal waveforms. Combining these waveforms with various amplitudes and either polarity can reproduce any combination of 8 pixels. Thus combining the 64 coefficients of the 2-D DCT will result in the original 8x8 pixel

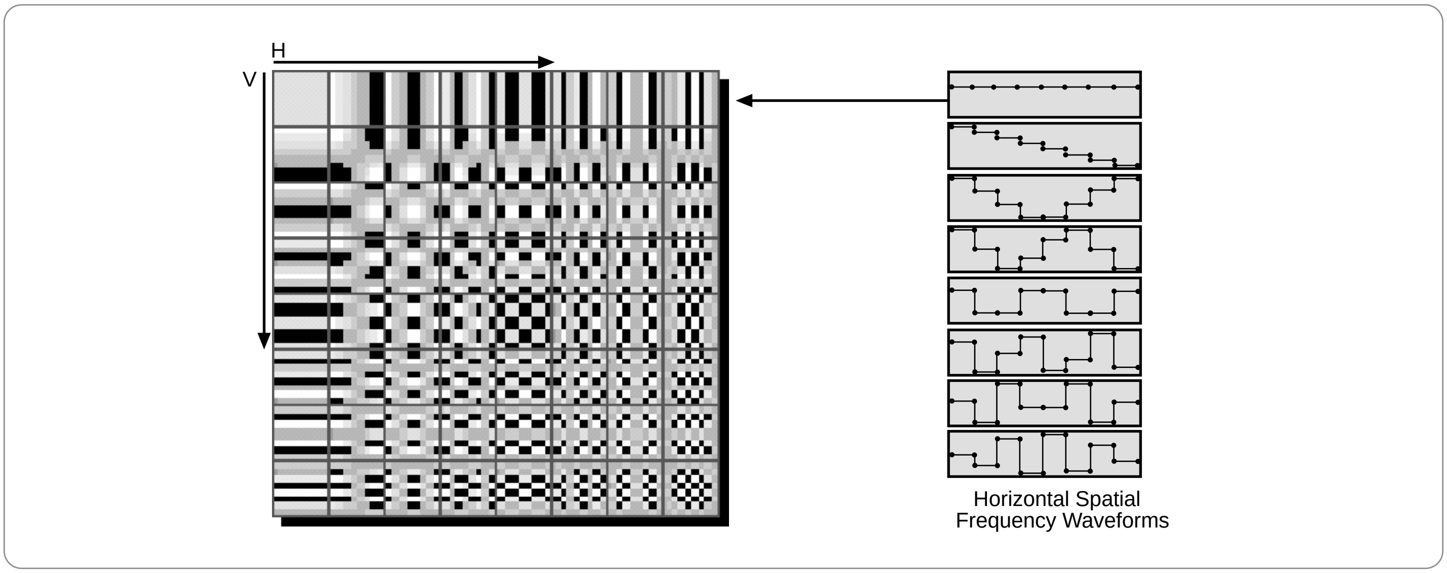


Figure 2-3.

block. Clearly for color pictures, the color difference samples will also need to be handled. Y, C_b and C_r data are assembled into separate 8x8 arrays and are transformed individually.

In much real program material, many of the coefficients will have zero or near-zero values and, therefore, will not be transmitted. This fact results in significant compression that is virtually lossless. If a higher compression factor is needed, then the wordlength of the non-zero coefficients must be reduced. This reduction will reduce accuracy of these coefficients and will introduce losses into the process. With care, the losses can be introduced in a way that is least visible to the viewer.

2.3 Weighting

Figure 2-4 shows that the human perception of noise in pictures is not uniform but is a function of the spatial frequency. More noise can be tolerated at high spatial frequencies. Also, video noise is effectively masked by fine detail in the picture, whereas in plain areas it is highly visible. The reader will be aware that traditional noise measurements are frequently weighted so that technical measurements relate more closely to the subjective result.

Compression reduces the accuracy of coefficients and has a similar effect to using shorter wordlength samples in PCM; that is, the noise level rises. In PCM, the result of shortening the word-length is that the noise level rises equally at all frequencies. As the DCT splits the signal into different frequencies, it becomes possible to control the spectrum of the noise. Effectively, low-frequency coefficients are rendered more accurately than high-frequency coefficients by a process of weighting.

Figure 2-5 shows that, in the weighting process, the coefficients from the DCT are divided by constants that are a function of two-dimensional frequency. Low-frequency coefficients will be divided by small numbers, and high-frequency coefficients will be divided by large numbers. Following the division, the result is truncated to the nearest integer. This truncation is a form of requantizing. In the absence of weighting, this requantizing would have the effect of uniformly increasing the size of the quantizing step, but with weighting, it increases the step size according to the division factor.

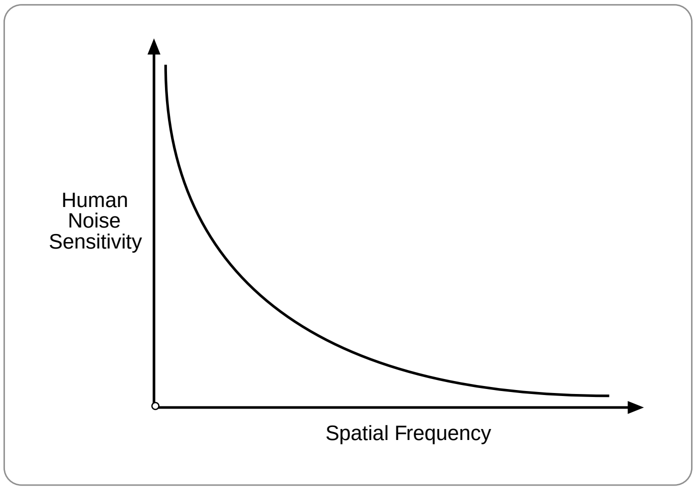


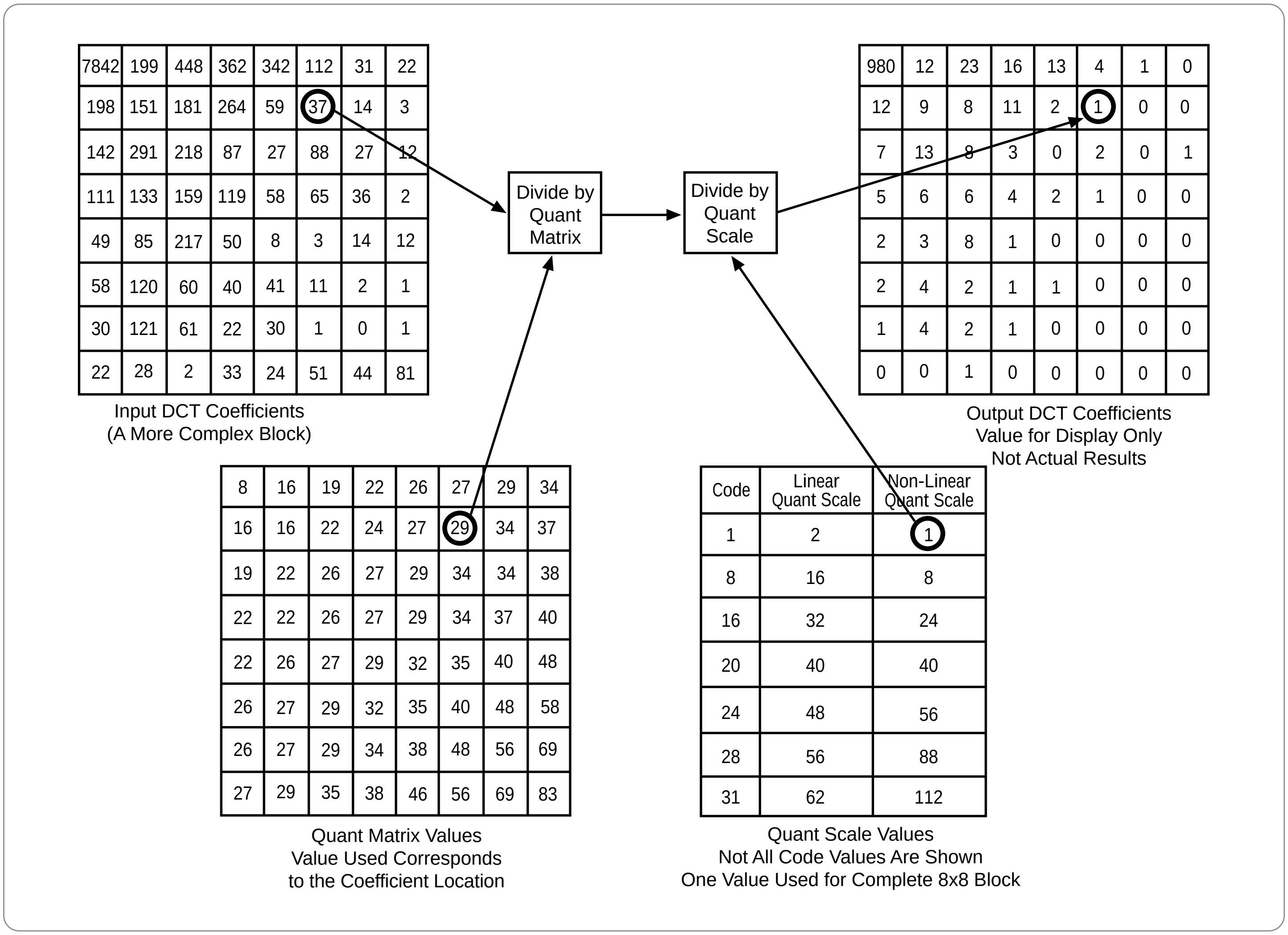
Figure 2-4.

As a result, coefficients representing low spatial frequencies are requantized with relatively small steps and suffer little increased noise. Coefficients representing higher spatial frequencies are requantized with large steps and suffer more noise. However, fewer steps means that fewer bits are needed to identify the step and compression is obtained.

In the decoder, low-order zeros will be added to return the weighted coefficients to their correct magnitude. They will then be multiplied by inverse weighting factors. Clearly, at high frequencies the multiplication factors will be larger, so the requantizing noise will be greater. Following inverse weighting, the coefficients will have their original DCT output values, plus requantizing error, which will be greater at high frequency than at low frequency.

As an alternative to truncation, weighted coefficients may be nonlinearly requantized so that the quantizing step size increases with the magnitude of the coefficient. This technique allows higher compression factors but worse levels of artifacts.

Clearly, the degree of compression obtained and, in turn, the output bit rate obtained, is a function of the severity of the requantizing process. Different bit rates will require different weighting tables. In MPEG, it is possible to use various different weighting tables and the table in use can be transmitted to the decoder, so that correct decoding is ensured.



7842	199	448	362	342	112	31	22
198	151	181	264	59	37	14	3
142	291	218	87	27	88	27	12
111	133	159	119	58	65	36	2
49	85	217	50	8	3	14	12
58	120	60	40	41	11	2	1
30	121	61	22	30	1	0	1
22	28	2	33	24	51	44	81

Input DCT Coefficients
(A More Complex Block)

980	12	23	16	13	4	1	0
12	9	8	11	2	1	0	0
7	13	8	3	0	2	0	1
5	6	6	4	2	1	0	0
2	3	8	1	0	0	0	0
2	4	2	1	1	0	0	0
1	4	2	1	0	0	0	0
0	0	1	0	0	0	0	0

Output DCT Coefficients
Value for Display Only
Not Actual Results

8	16	19	22	26	27	29	34
16	16	22	24	27	29	34	37
19	22	26	27	29	34	34	38
22	22	26	27	29	34	37	40
22	26	27	29	32	35	40	48
26	27	29	32	35	40	48	58
26	27	29	34	38	48	56	69
27	29	35	38	46	56	69	83

Quant Matrix Values
Value Used Corresponds
to the Coefficient Location

Code	Linear Quant Scale	Non-Linear Quant Scale
1	2	1
8	16	8
16	32	24
20	40	40
24	48	56
28	56	88
31	62	112

Quant Scale Values
Not All Code Values Are Shown
One Value Used for Complete 8x8 Block

Figure 2-5.

2.4 Scanning

In typical program material, the most significant DCT coefficients are generally found in or near the top-left corner of the matrix. After weighting, low-value coefficients might be truncated to zero. More efficient transmission can be obtained if all of the non-zero coefficients are sent first, followed by a code indicating that the remainder are all zero. Scanning is a technique that increases the probability of achieving this result, because it sends coefficients in descending order of magnitude probability. Figure 2-6a (see next page) shows that in a non-interlaced system, the probability of a coefficient having a high value is highest in the top-left corner and lowest in the bottom-right corner. A 45 degree diagonal zigzag scan is the best sequence to use here.

In Figure 2-6b, an alternative scan pattern is shown that may be used for interlaced sources. In an interlaced picture, an 8x8 DCT block from one field extends over twice the vertical screen area, so that for a given picture detail, vertical frequencies will appear to be twice as great as horizontal frequencies. Thus, the ideal scan for an interlaced picture will be on a diagonal that is twice as steep. Figure 2-6b shows that a given vertical spatial frequency is scanned before scanning the same horizontal spatial frequency.

2.5 Entropy Coding

In real video, not all spatial frequencies are present simultaneously; therefore, the DCT coefficient matrix will have zero terms in it. Requantization will increase the number of zeros by eliminating small values. Despite the

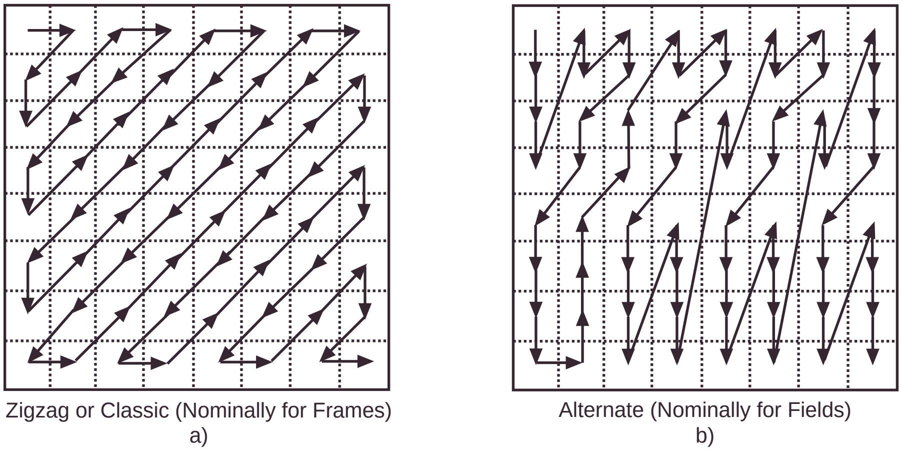

Zigzag or Classic (Nominally for Frames)
a)
Alternate (Nominally for Fields)
b)

Figure 2-6.

use of scanning, zero coefficients will still appear between the significant values. Run length coding (RLC) allows these coefficients to be handled more efficiently. Where repeating values, such as a string of zeros, are present, RLC simply transmits the number of zeros rather than each individual bit.

The probability of occurrence of particular coefficient values in real video can be studied. In practice, some values occur very often; others occur less often. This statistical information can be used to achieve further compression using variable length coding (VLC). Frequently occurring values are converted to short code words, and infrequent values are converted to long code words. To aid decoding, no code word can be the prefix of another.

2.6 A Spatial Coder

Figure 2-7 ties together all of the preceding spatial coding concepts. The input signal is assumed to be 4:2:2 SDI (Serial Digital Interface), which may have 8- or 10-bit wordlength. MPEG uses only 8-bit resolution; therefore, a rounding stage will be needed when the SDI signal contains 10-bit words. Most MPEG profiles operate with 4:2:0 sampling; therefore, a vertical low-pass filter/interpolation stage will be needed. Rounding and color subsampling introduces a small irreversible loss of information and a proportional reduction in bit rate. The raster scanned input format will need to be stored so that it can be converted to 8x8 pixel blocks.

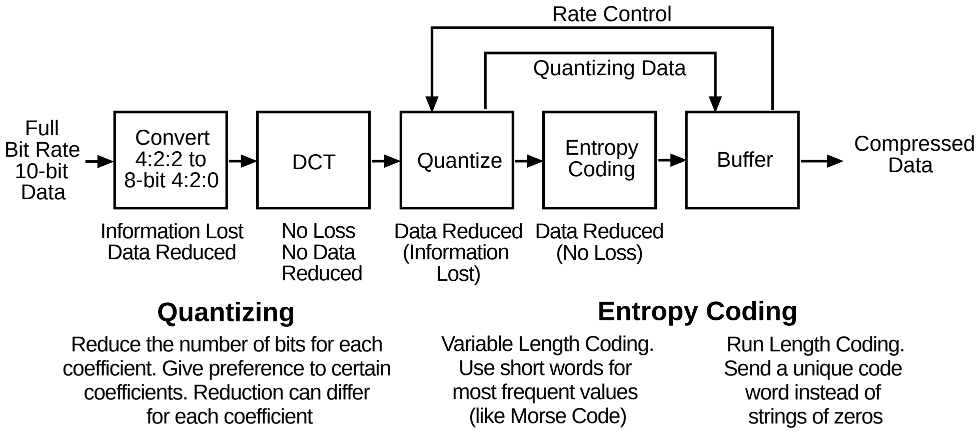

Rate Control
Quantizing Data
Full
Bit Rate
10-bit
Data
Convert
4:2:2 to
8-bit 4:2:0
DCT
Quantize
Entropy
Coding
Buffer
Compressed
Data
Information Lost
Data Reduced
No Loss
No Data
Reduced
Data Reduced
(Information
Lost)
Data Reduced
(No Loss)
Quantizing
Reduce the number of bits for each coefficient. Give preference to certain coefficients. Reduction can differ for each coefficient
Entropy Coding
Variable Length Coding. Use short words for most frequent values (like Morse Code)
Run Length Coding. Send a unique code word instead of strings of zeros

Figure 2-7.

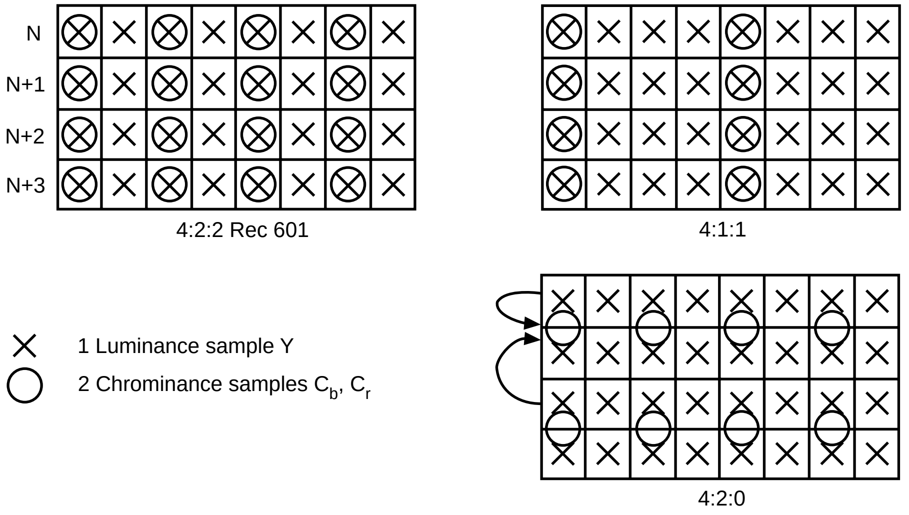


Figure 2-8.

The DCT stage transforms the picture information to the frequency domain. The DCT itself does not achieve any compression. Following DCT, the coefficients are weighted and truncated, providing the first significant compression. The coefficients are then zigzag scanned to increase the probability that the significant coefficients occur early in the scan. After the last non-zero coefficient, an EOB (end of block) code is generated.

Coefficient data are further compressed by run-length and variable-length coding. In a variable bit-rate system, the quantizing may be fixed, but in a fixed bit-rate system, a buffer memory is used to absorb variations in coding difficulty. Highly detailed pictures will tend to fill the buffer, whereas plain pictures will allow it to empty. If the buffer is in danger of overflowing, the requantizing steps will have to be made larger, so that the compression factor is raised.

In the decoder, the bit stream is deserialized and the entropy coding is reversed to reproduce the weighted coefficients. The coefficients are placed in the matrix according to the zigzag scan, and inverse weighting is applied to recreate the block of DCT coefficients. Following an inverse transform, the 8x8 pixel block is recreated. To obtain a raster-scanned output, the blocks are stored in RAM, which is read a line at a time. To obtain a 4:2:2 output from 4:2:0 data, a vertical interpolation process will be needed as shown in Figure 2-8.

The chroma samples in 4:2:0 are positioned half way between luminance samples in the vertical axis so that they are evenly spaced when an interlaced source is used.

2.7 Temporal Coding

Temporal redundancy can be exploited by inter-coding or transmitting only the differences between pictures. Figure 2-9 shows that a one-picture

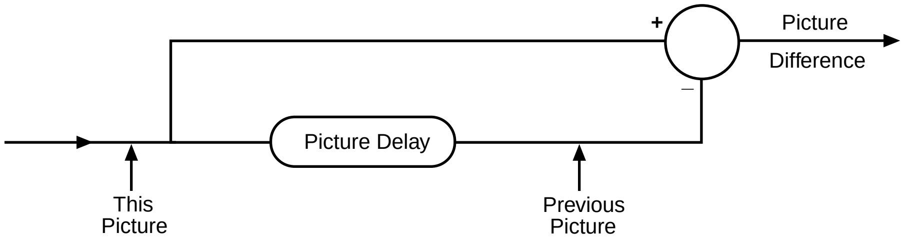


Figure 2-9.

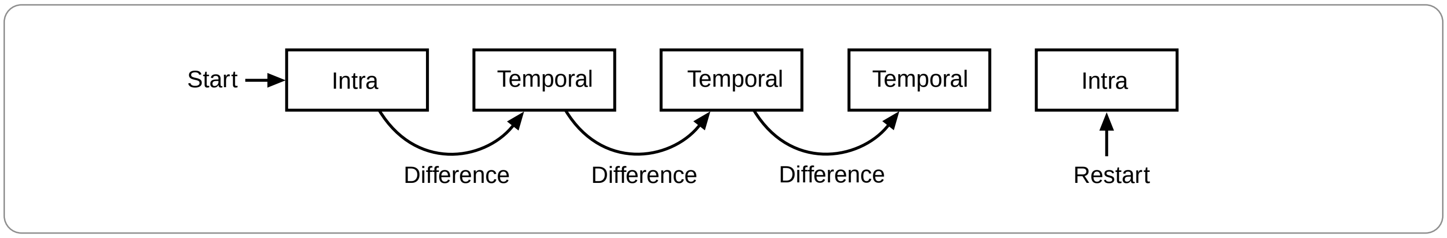


Figure 2-10.

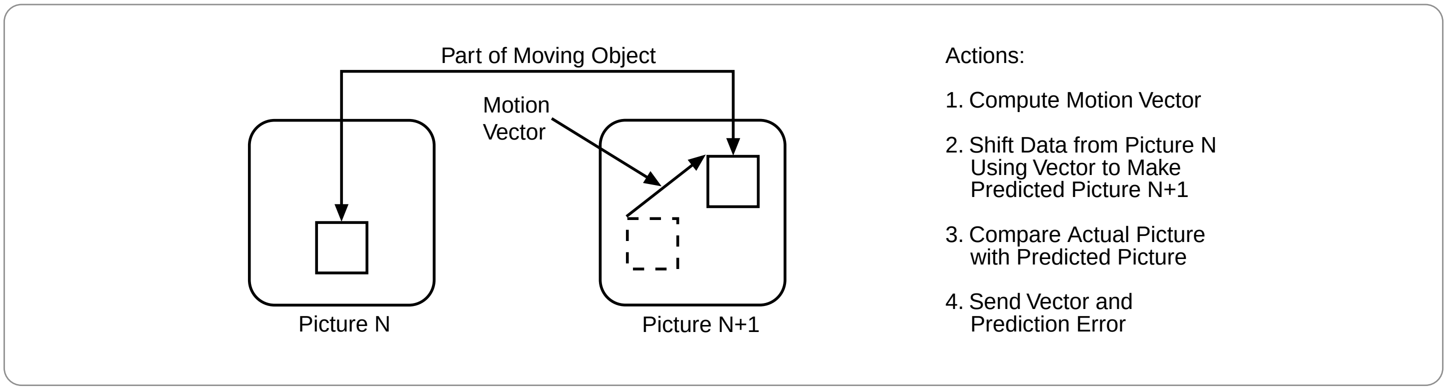
delay combined with a subtracter can compute the picture differences. The picture difference is an image in its own right and can be further compressed by the spatial coder as was previously described. The decoder reverses the spatial coding and adds the difference picture to the previous picture to obtain the next picture.

There are some disadvantages to this simple system. First, as only differences are sent, it is impossible to begin decoding after the start of the transmission. This limitation makes it difficult for a decoder to provide pictures following a switch from one bit stream to another (as occurs when the viewer changes channels). Second, if any part of the difference data is incorrect, the error in the picture will propagate indefinitely.

The solution to these problems is to use a system that is not completely differential. Figure 2-10 shows that periodically complete pictures are sent. These are called Intra-coded pictures (or I-pictures), and they are obtained by spatial compression only. If an error or a channel switch occurs, it will be possible to resume correct decoding at the next I-picture.

2.8 Motion Compensation

Motion reduces the similarities between pictures and increases the data needed to create the difference picture. Motion compensation is used to increase the similarity. Figure 2-11 shows the principle. When an object moves across the TV screen, it may appear in a different place in each picture, but it does not change in appearance very much. The picture difference can be reduced by measuring the motion at the encoder. This is sent to the decoder as a vector. The decoder uses the vector to shift part of the previous picture to a more appropriate place in the new picture.



Actions:

1. Compute Motion Vector
2. Shift Data from Picture N Using Vector to Make Predicted Picture N+1
3. Compare Actual Picture with Predicted Picture
4. Send Vector and Prediction Error

Figure 2-11.

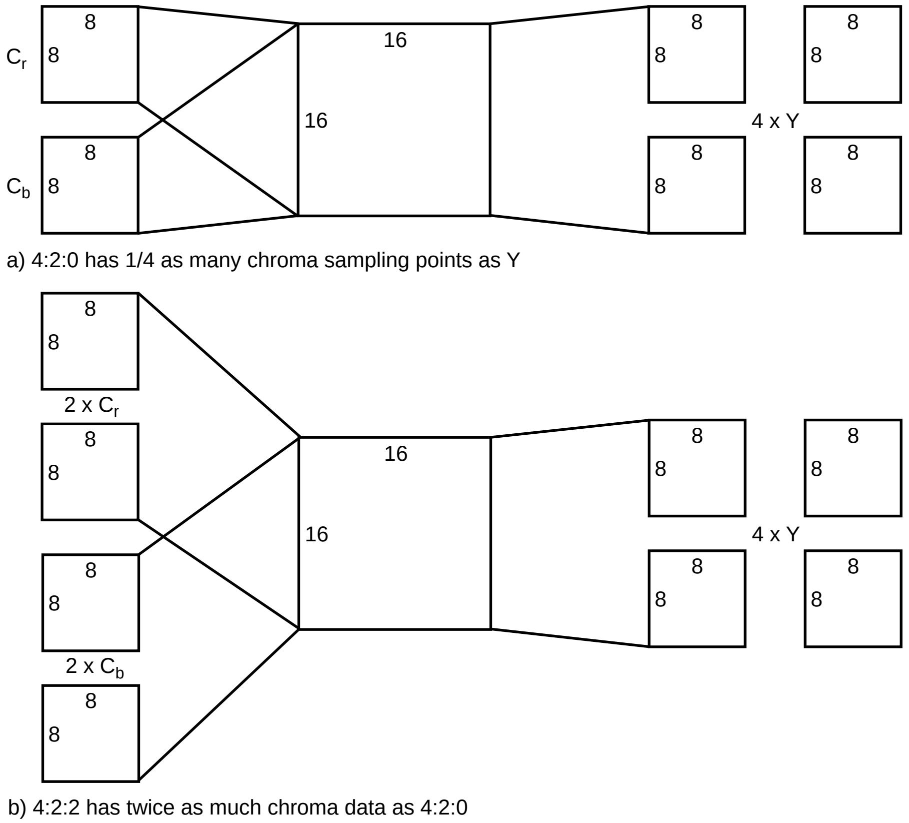

8
C_r 8
16
16
8
8
8
8
4 x Y
8
C_b 8
8
8
8
8
a) 4:2:0 has 1/4 as many chroma sampling points as Y
8
8
2 x C_r
8
8
16
16
8
8
8
8
4 x Y
8
8
8
8
8
8
2 x C_b
8
8
b) 4:2:2 has twice as much chroma data as 4:2:0

Figure 2-12.

One vector controls the shifting of an entire area of the picture that is known as a macroblock. The size of the macroblock is determined by the DCT coding and the color subsampling structure. Figure 2-12a shows that, with a 4:2:0 system, the vertical and horizontal spacing of color samples is exactly twice the spacing of luminance. A single 8x8 DCT block of color samples extends over the same area as four 8x8 luminance blocks; therefore this is the minimum picture area that can be shifted by a vector. One 4:2:0 macroblock contains four luminance blocks: one C_b block and one C_r block.

In the 4:2:2 profile, color is only subsampled in the horizontal axis. Figure 2-12b shows that in 4:2:2, a single 8x8 DCT block of color samples extends over two luminance blocks. A 4:2:2 macroblock contains four luminance blocks: two C_b blocks and two C_r blocks.

The motion estimator works by comparing the luminance data from two successive pictures. A macroblock in the first picture is used as a reference. The correlation between the reference and the next picture is measured at all possible displacements with a resolution of half a pixel over the entire search range. When the greatest correlation is found, this correlation is assumed to represent the correct motion.

The motion vector has a vertical and horizontal component. In typical program material, a moving object may extend over a number of macroblocks. A greater compression factor is obtained if the vectors are transmitted differentially. When a large object moves, adjacent macroblocks have the same vectors and the vector differential becomes zero.

Motion vectors are associated with macroblocks, not with real objects in the image and there will be occasions where part of the macroblock moves and part of it does not. In this case, it is impossible to compensate properly. If the motion of the moving part is compensated by transmitting a vector, the stationary part will be incorrectly shifted, and it will need difference data to be corrected. If no vector is sent, the stationary part will be correct, but difference data will be needed to correct the moving part. A practical compressor might attempt both strategies and select the one that required the least data.

2.9 Bidirectional Coding

When an object moves, it conceals the background at its leading edge and reveals the background at its trailing edge. The revealed background requires new data to be transmitted because the area of background was previously concealed and no information can be obtained from a previous picture. A similar problem occurs if the camera pans; new areas come into view and nothing is known about them. MPEG helps to minimize this problem by using bidirectional coding, which allows information to be taken from pictures before and after the current picture. If a background is being revealed, it will be present in a later picture, and the information can be moved backwards in time to create part of an earlier picture.

Figure 2-13 shows the concept of bidirectional coding. On an individual macroblock basis, a bidirectionally-coded picture can obtain motion-compensated data from an earlier or later picture, or even use an average of earlier and later data. Bidirectional coding significantly reduces the amount of difference data needed by improving the degree of prediction possible. MPEG does not specify how an encoder should be built, only what constitutes a compliant bit stream. However, an intelligent compressor could try all three coding strategies and select the one that results in the least data to be transmitted.

2.10 I-, P- and B-pictures

In MPEG, three different types of pictures are needed to support differential and bidirectional coding while minimizing error propagation:

I-pictures are intra-coded pictures that need no additional information for decoding. They require a lot of data compared to other picture types, and therefore they are not transmitted any more frequently than necessary. They consist primarily of transform coefficients and have no vectors. I-pictures are decoded without reference to any other pictures, so they allow the viewer to switch channels, and they arrest error propagation.

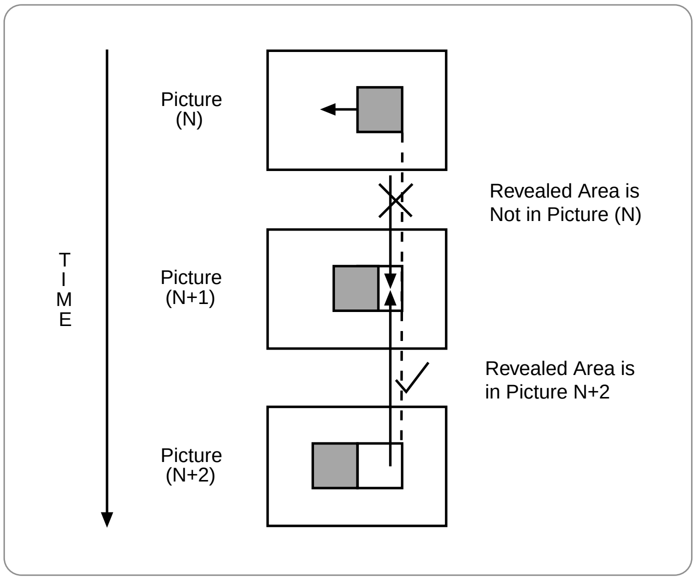

TIME
Picture (N)
Picture (N+1)
Picture (N+2)
Revealed Area is Not in Picture (N)
Revealed Area is in Picture N+2

Figure 2-13.

P-pictures are forward predicted from an earlier picture, which could be an I-picture or a P-picture. P-picture data consists of vectors describing where, in the previous picture, each macro-block should be taken from, and transform coefficients that describe the correction or difference data that must be added to that macroblock. Where no suitable match for a macroblock could be found by the motion compensation search, intra data is sent to code that macroblock. P-pictures require roughly half the data of an I-picture.

B-pictures are bidirectionally predicted from earlier and/or later I- or P-pictures. B-picture data consists of vectors describing where in earlier or later pictures data should be taken from. It also contains the intracoded data that provide necessary corrections. Again, when no suitable match for a macroblock is found by the motion compensation search, intra data is sent to code that macroblock. Bidirectional prediction is quite effective, so most macroblocks in a B-picture will be coded largely by motion vectors. Also, a B-picture is never used as a reference for coding other pictures, so there is no possibility of error propagation. This permits encoders to use more aggressive requantization for correction data. A typical B-picture requires about one quarter the data of an I-picture.

Note that a B-picture does not have to use both directions of prediction; in some circumstances only one direction is employed. This option may be used when constructing closed groups of pictures (GOP).

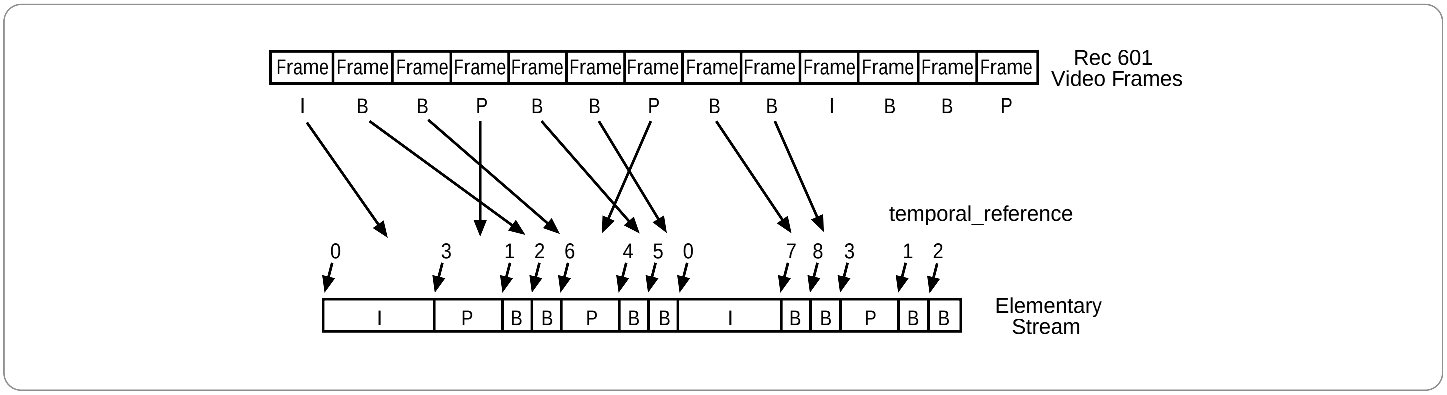

Frame Frame Frame Frame Frame Frame Frame Frame Frame Frame Frame Frame Frame
Rec 601
Video Frames
I B B P B B P B B I B B P
temporal_reference
0 3 1 2 6 4 5 0 7 8 3 1 2
I P B B P B B I B B P B B
Elementary
Stream

Figure 2-14.

Figure 2-14 introduces the concept of the GOP. The GOP represents the structure of I-, P-, and B-pictures in the sequence. Generally the GOP structure repeats through the sequence, but the GOP length and structure may be changed at any time. There are no formal limits on the length of a GOP, but for transmission purposes a typical length is 12 or 15 pictures.

The nature of MPEG's temporal compression means that the transmission order of pictures is not the same as the display order. A P-picture naturally follows the I- or P-picture from which it is predicted, so there are no special requirements. A bidirectionally-coded B-picture, however, cannot be decoded until both of its reference pictures have been received and decoded. Figure 2-14 shows the pictures of a GOP in display order at the top, and in transmission order below. Note that, in transmission order, the B-pictures always follow the two reference pictures from which they are predicted.

There are two types of GOP, open and closed. A *closed GOP* requires no reference outside itself. In display order, it may begin with an I-picture and end with a P-picture. In transmission order there will usually be B-pictures following the last P-picture, but these are pictures that will be displayed before that last P-picture.

It is possible to start and or end a closed GOP with B-pictures (in display order), but in this case the starting and ending B-pictures must be coded using only a single direction of prediction. B-pictures at the start of a closed GOP must use backward prediction only. B-pictures at the end of a closed GOP may use forward prediction only – similar to a P-picture, but B-picture rules would be used for requantization etc.

An *open GOP* does not have these restrictions on prediction vectors. For example, B-pictures at the end of the GOP can use forward prediction from the last P-picture and backward prediction from the first I-picture of the next GOP. This structure is slightly more efficient, but predictions can cross any picture boundary. It is much more difficult to splice bit streams, and events such as channel changes are more likely to cause picture errors.

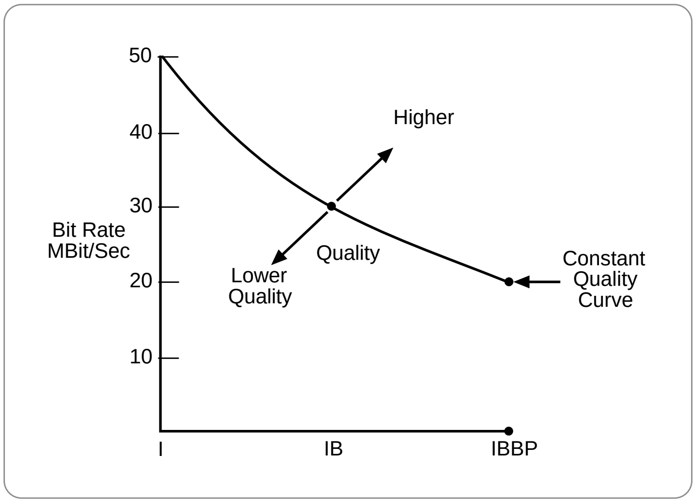


Figure 2-15.

The GOP structure may be altered by the encoder when there are scene changes. Predictions across a scene change will usually fail, since there will be large entropy between the two pictures either side of the scene change. An encoder may choose to detect the scene change, use a closed GOP leading up to the scene change, and start a new GOP (open or closed) with an I-picture representing the first picture of the new scene.

Sending picture data out of sequence requires additional memory at the encoder and decoder and also causes delay. The number of bidirectionally-coded pictures between intra- or forward-predicted pictures must be restricted to reduce cost, and to minimize delay if this is an issue.

Figure 2-15 shows the trade-off that must be made between compression factor and coding delay. For a given quality, sending only I-pictures requires more than twice the bit rate of an IBBP sequence.

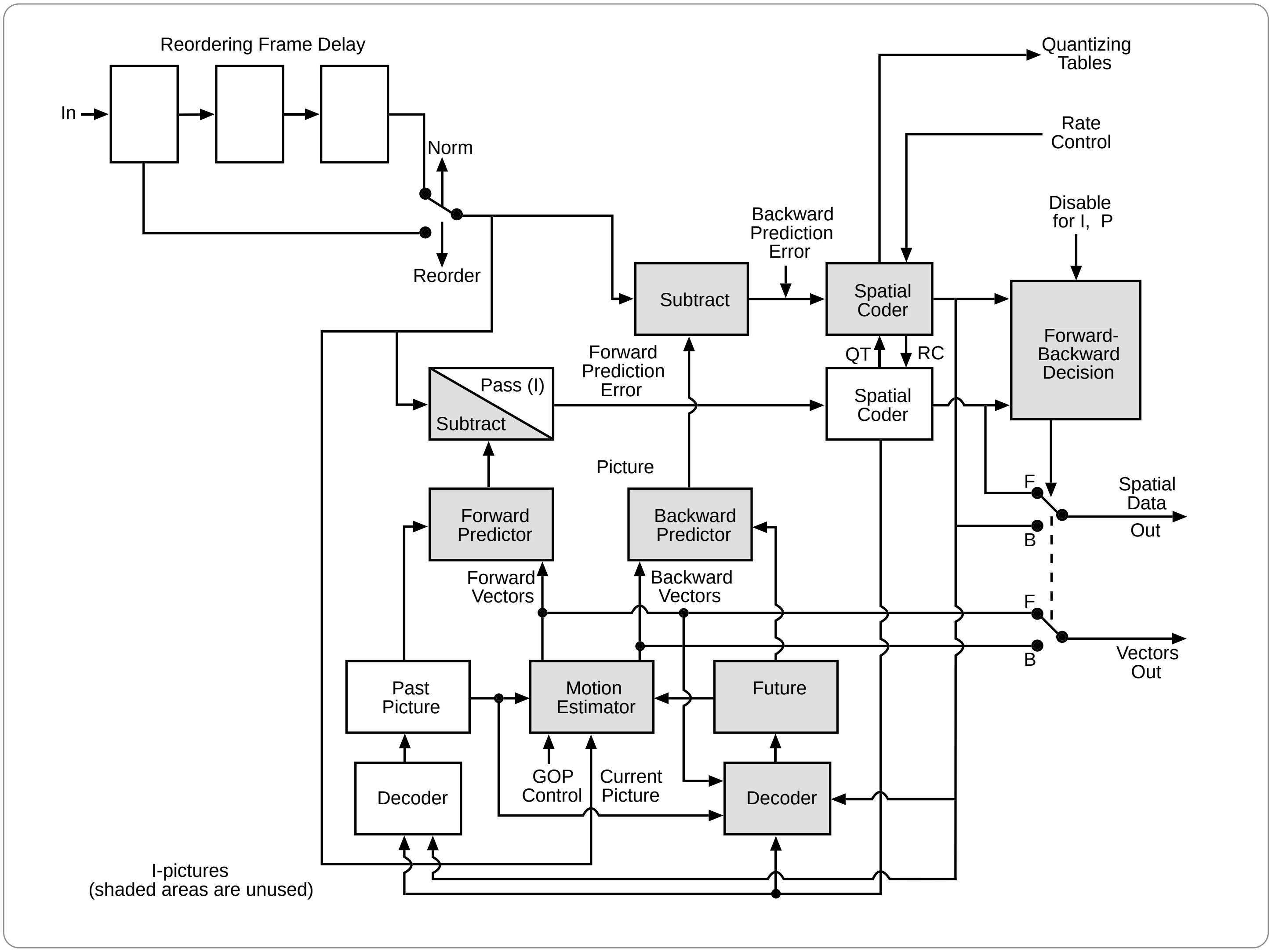


Figure 2-16a.

2.11 An MPEG Compressor

Figures 2-16a, b, and c show a typical bidirectional motion compensator structure. Pre-processed input video enters a series of frame stores that can be bypassed to change the picture order. The data then enter the subtracter and the motion estimator. To create an I-picture, the end of the input delay is selected and the subtracter is turned off so that the data pass straight through to be spatially coded (see Figure 2-16a). Subtracter output data also pass to a frame store that can hold several pictures. The I-picture is held in the store.

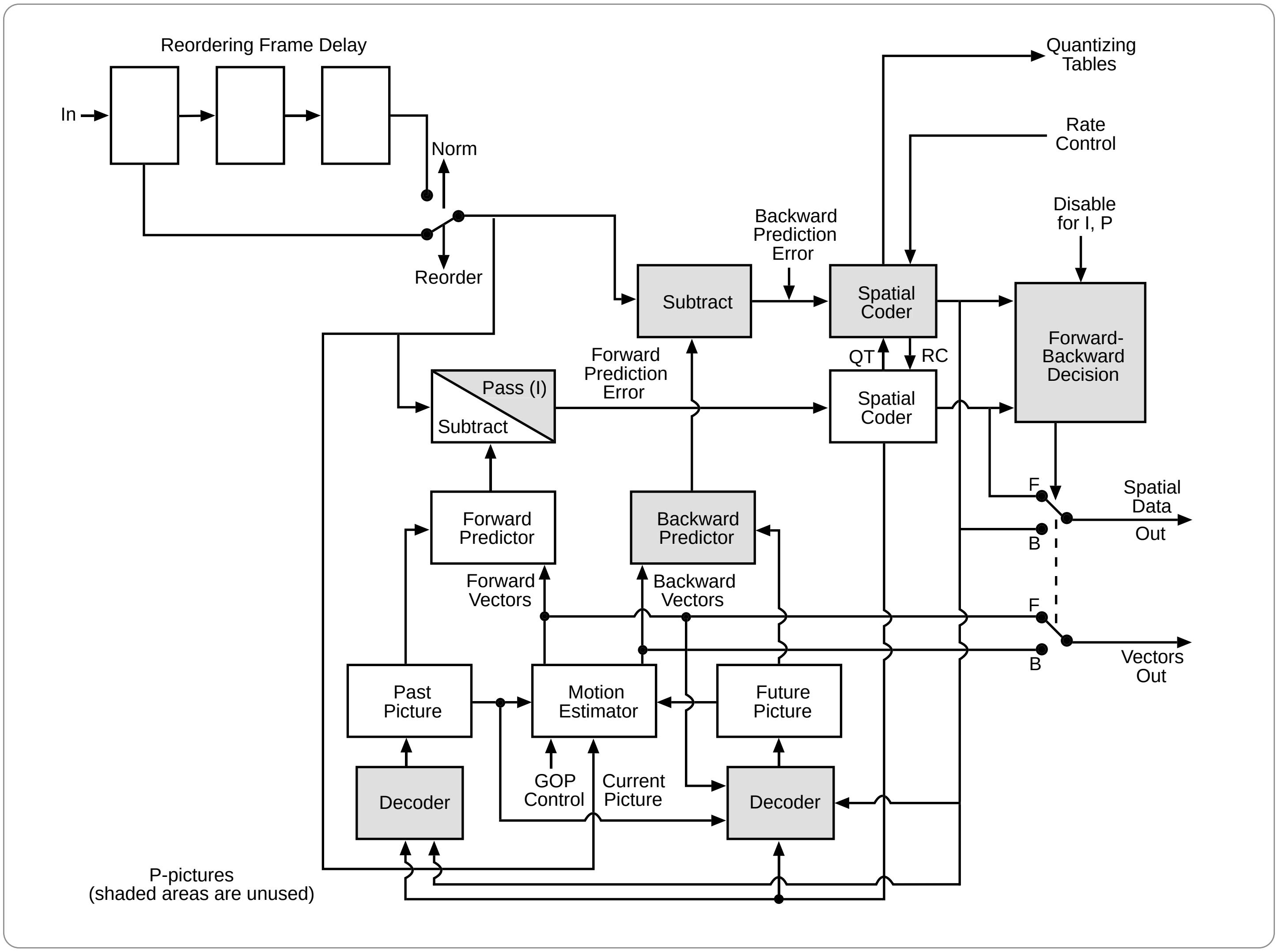


Figure 2-16b.

To encode a P-picture, the B-pictures in the input buffer are bypassed, so that the future picture is selected (see Figure 2-16b). The motion estimator compares the I-picture in the output store with the P-picture in the input store to create forward motion vectors. The I-picture macroblocks are shifted by these vectors to make a predicted P-picture. The predicted P-picture is subtracted from the actual P-picture to produce the prediction error, which is spatially coded and sent along with the vectors. The prediction error is also added to the predicted P-picture to create a locally decoded P-picture that also enters the output store.

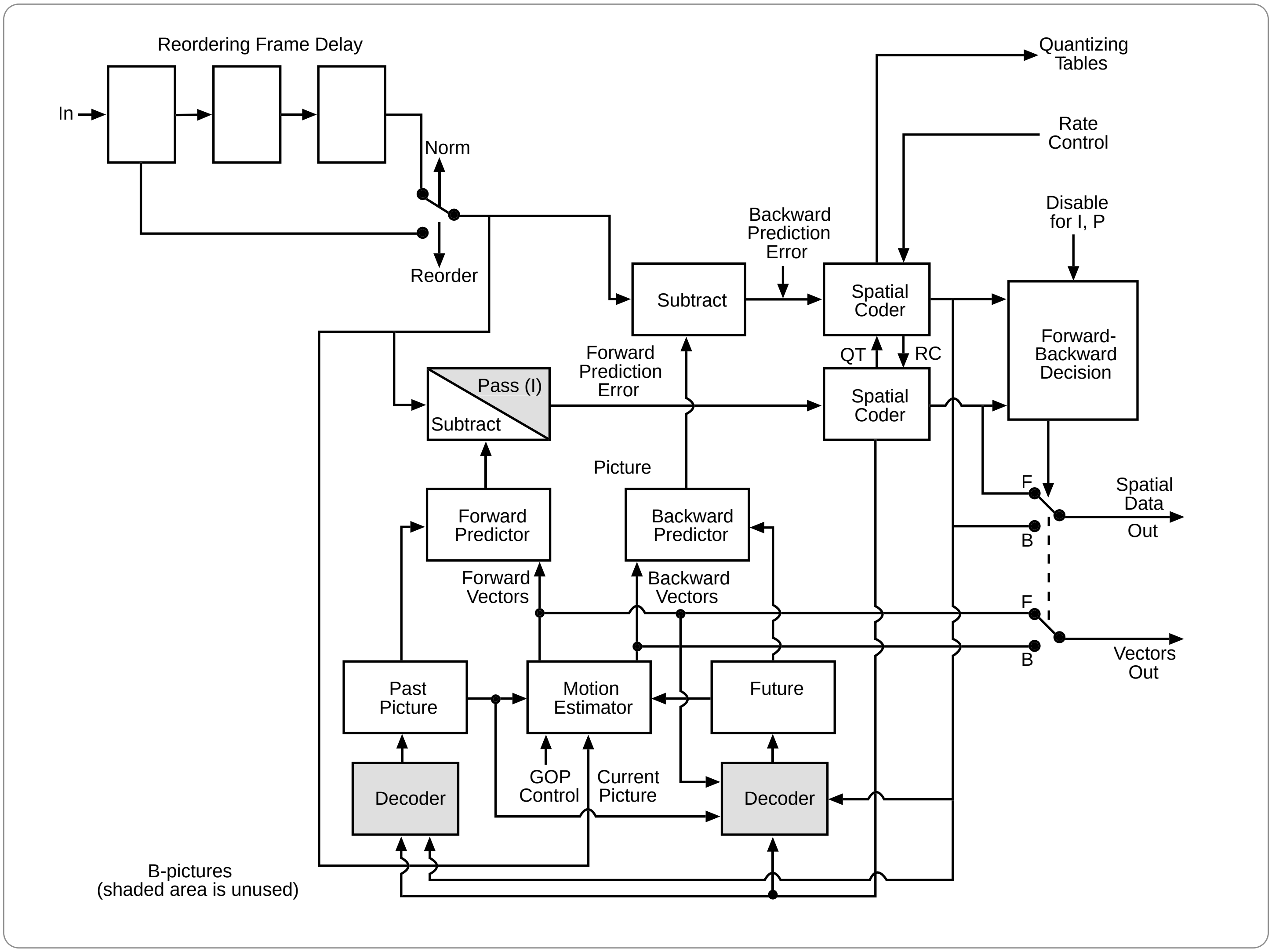


Figure 2-16c.

The output store then contains an I-picture and a P-picture. A B-picture from the input buffer can now be selected. The motion compensator will compare the B-picture with the I-picture that precedes it and the P-picture that follows it to obtain bidirectional vectors (see Figure 2-16c). Forward and backward motion compensation is performed to produce two predicted B-pictures. These are subtracted from the current B-picture. On a macroblock-by-macroblock basis, the forward or backward data are selected according to which represent the smallest differences. The picture differences are then spatially coded and sent with the vectors.

When all of the intermediate B-pictures are coded, the input memory will once more be bypassed to create a new P-picture based on the previous P-picture.

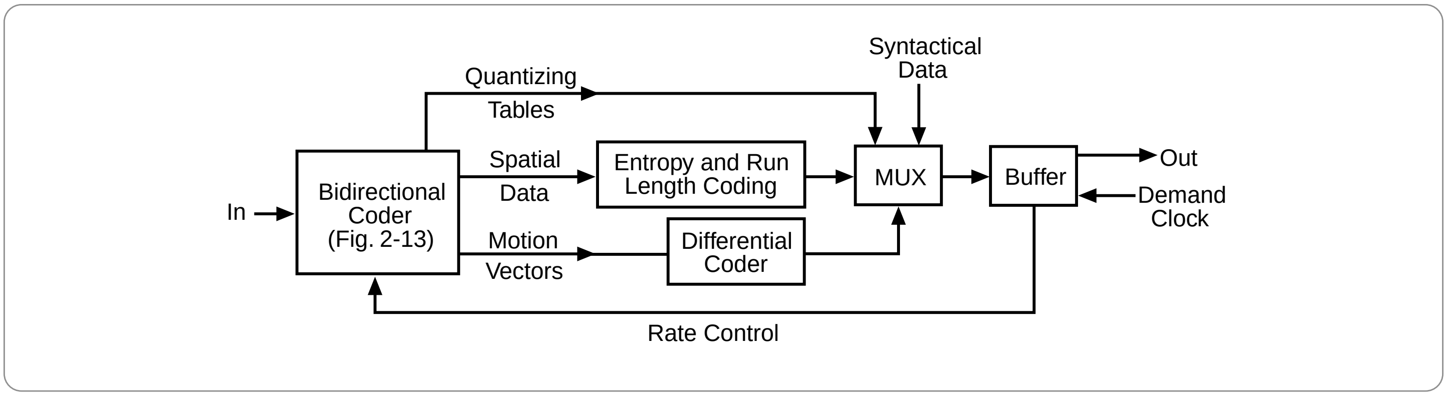


Figure 2-17.

Figure 2-17 shows an MPEG coder. The motion compensator output is spatially coded and the vectors are added in a multiplexer. Syntactical data is also added which identifies the type of picture (I, P, or B) and provides other information to help a decoder (see Section 5 – Elementary Streams). The output data are buffered to allow temporary variations in bit rate. If the mean bit rate is too high, the buffer will tend to fill up. To prevent overflow, quantization will have to be made more severe. Equally, should the buffer show signs of underflow, the quantization will be relaxed to maintain the average bit rate.

2.12 Preprocessing

A compressor attempts to eliminate redundancy within the picture and between pictures. Anything that reduces that apparent redundancy, that is not picture content, is undesirable. Noise and film grain are particularly problematic because they generally occur over the entire picture. After the DCT process, noise results in more non-zero coefficients, and the coder cannot distinguish this information from genuine picture data. Heavier quantizing will be required to encode all of the coefficients, reducing picture quality. Noise also reduces similarities between successive pictures, increasing the difference data needed.

Residual subcarrier in video decoded from composite video is a serious problem because it results in high, spatial frequencies that are normally at a low level in component programs. Subcarrier also alternates in phase from picture to picture causing an increase in difference data. Naturally, any composite decoding artifact that is visible in the input to the MPEG coder is likely to be reproduced at the decoder.

Any practice that causes unwanted motion is to be avoided. Unstable camera mountings, in addition to giving a shaky picture, increase picture differences and vector transmission requirements. This will also happen with telecine material if sprocket hole damage results in film weave or hop. In general, video that is to be compressed must be of the highest quality possible. If high quality cannot be achieved, then noise reduction and other stabilization techniques will be desirable.

If a high compression factor is required, the level of artifacts can increase, especially if input quality is poor. In this case, it may be better to reduce the entropy presented to the coder by using pre-filtering. The video signal is subject to two-dimensional, low-pass filtering, which reduces the number of coefficients needed and reduces the level of artifacts. The picture will be less sharp, but less sharpness is preferable to a high level of artifacts.

In most MPEG-2 applications, 4:2:0 sampling is used, which requires a chroma downsampling process if the source is 4:2:2. In MPEG-1, the luminance and chroma are further downsampled to produce an input picture or CIF (common image format) that is only 352-pixels wide. This technique reduces the entropy by a further factor. For very high compression, the QCIF (quarter common image format) picture, which is 176-pixels wide, is used. Downsampling is a process that combines a spatial low-pass filter with an interpolator. Downsampling interlaced signals is problematic because vertical detail is spread over two fields that may de-correlate due to motion.

When the source material is telecine, the video signal has different characteristics than normal video. In 50-Hz video, pairs of fields represent the same film frame, and there is no motion between them. Thus, the motion between fields alternates between zero and the motion between frames. In 60-Hz video, 3:2 pulldown is used to obtain 60 Hz from 24 Hz film. One frame is made into two fields; the next is made into three fields, and so on.

Consequently, one field in five is completely redundant. MPEG handles film material best by discarding the third field in 3:2 systems. A 24-Hz code in the transmission alerts the decoder to recreate the 3:2 sequence by re-reading a field store. In 50- and 60-Hz telecine, pairs of fields are deinterlaced to create frames, and then motion is measured between frames. The decoder can recreate interlace by reading alternate lines in the frame store.

A cut is a difficult event for a compressor to handle because it often results in an almost complete prediction failure, requiring a large amount of correction data. If a coding delay can be tolerated, a coder may detect cuts in advance and modify the GOP structure dynamically, so that an I-picture is inserted to coincide with the cut. In this case, the cut is handled with very little extra data. The last B-pictures before the I frame will almost certainly need to use forward prediction. In some applications that are not real-time, such as DVD mastering, a coder could take two passes at the input video: one pass to identify the difficult or high entropy areas and create a coding strategy, and a second pass to actually compress the input video.

2.13 Wavelets

All transforms suffer from uncertainty because the more accurately the frequency domain is known, the less accurately the time domain is known (and vice versa). In most transforms such as discreet Fourier transport (DFT) and discreet cosine transform (DCT), the block length is fixed, so the time and frequency resolution is fixed. The frequency coefficients represent evenly spaced values on a linear scale. Unfortunately, because human senses are logarithmic, the even scale of the DFT and DCT gives inadequate frequency resolution at one end and excess resolution at the other.

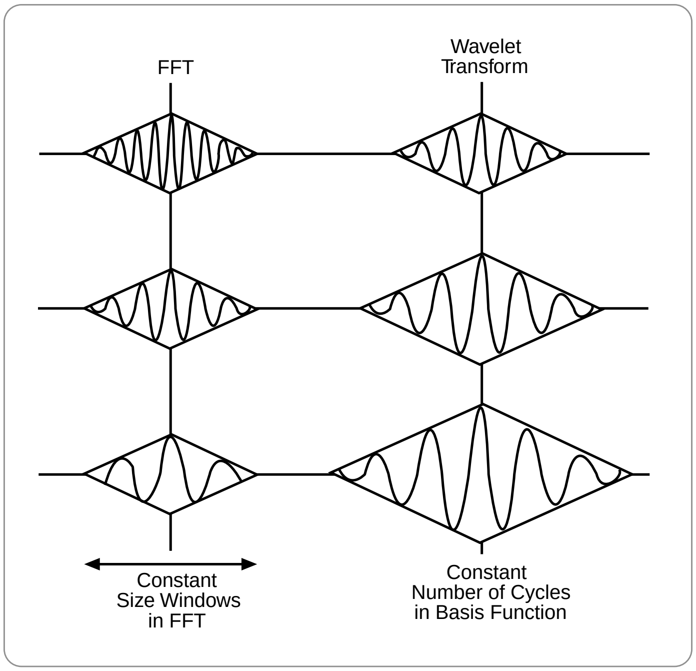

FFT
Wavelet Transform
Constant Size Windows in FFT
Constant Number of Cycles in Basis Function

Figure 2-18.

The wavelet transform is not affected by this problem because its frequency resolution is a fixed fraction of an octave and therefore has a logarithmic characteristic. This is done by changing the block length as a function of frequency. As frequency goes down, the block becomes longer. Thus, a characteristic of the wavelet transform is that the basis functions all contain the same number of cycles, and these cycles are simply scaled along the time axis to search for different frequencies. Figure 2-18 contrasts the fixed block size of the DFT/DCT with the variable size of the wavelet.

Wavelets are especially useful for audio coding because they automatically adapt to the conflicting requirements of the accurate location of transients in time and the accurate assessment of pitch in steady tones.

For video coding, wavelets have the advantage of producing resolution-scalable signals with almost no extra effort. In moving video, the advantages of wavelets are offset by the difficulty of assigning motion vectors to a variable size block, but in still-picture or I-picture coding this difficulty is not an issue. Wavelet coding has shown particular benefits for very-low bit rate applications. The artifacts generated by excessive quantization of wavelet coefficients generally appear as “smearing,” and these are much less objectionable than the “blockiness” that results from excessive quantization of DCT coefficients.

Section 3 – Audio Compression

Lossy audio compression is based entirely on the characteristics of human hearing, which must be considered before any description of compression is possible. Surprisingly, human hearing, particularly in stereo, is actually more critically discriminating than human vision, and consequently audio compression should be undertaken with care. As with video compression, audio compression requires a number of different levels of complexity according to the required compression factor.

3.1 The Hearing Mechanism

Hearing comprises physical processes in the ear and nervous/mental processes that combine to give us an impression of sound. The impression we receive is not identical to the actual acoustic waveform present in the ear canal because some entropy is lost. Audio compression systems that lose only that part of the entropy that will be lost in the hearing mechanism will produce good results.

The physical hearing mechanism consists of the outer, middle, and inner ears. The outer ear comprises the ear canal and the eardrum. The eardrum converts the incident sound into a vibration, in much the same way as a microphone diaphragm. The inner ear works by sensing vibrations transmitted through a fluid. The impedance of fluid is much higher than that of air and the middle ear acts as an impedance-matching transformer that improves power transfer.

Figure 3-1 shows that vibrations are transferred to the inner ear by the stirrup bone, which acts on the oval window. Vibrations in the fluid in the ear travel up the cochlea, a spiral cavity in the skull (shown unrolled in Figure 3-1 for clarity). The basilar membrane is stretched across the cochlea. This membrane varies in mass and stiffness along its length. At the end near the oval window, the membrane is stiff and light, so its resonant frequency is high. At the distant end, the membrane is heavy and soft and resonates at low frequency. The range of resonant frequencies available determines the frequency range of human hearing, which in most people is from 20 Hz to about 15 kHz.

Different frequencies in the input sound cause different areas of the membrane to vibrate. Each area has different nerve endings to allow pitch discrimination. The basilar membrane also has tiny muscles controlled by the nerves that together act as a kind of positive-feedback system that improves the Q-factor of the resonance.

The resonant behavior of the basilar membrane is an exact parallel with the behavior of a transform analyzer. According to the uncertainty theory of transforms, the more accurately the frequency domain of a signal is known, the less accurately the time domain is known. Consequently, the more able a transform is to discriminate between two frequencies, the less able it is to discriminate between the time of two events. Human hearing has evolved with a certain compromise that balances time-uncertainty discrimination and frequency discrimination; in the balance, neither ability is perfect.

The imperfect frequency discrimination results in the inability to separate closely spaced frequencies. This inability is known as auditory masking, defined as the reduced sensitivity to sound in the presence of another.

Figure 3-2a (see next page) shows that the threshold of hearing is a function of frequency. The greatest sensitivity is, not surprisingly, in the speech range. In the presence of a single tone, the threshold is modified as in Figure 3-2b. Note that the threshold is raised for tones at higher frequency and to some extent at lower frequency. In the presence of a complex input spectrum, such as music, the threshold is raised at nearly all frequencies. One consequence of this behavior is that the hiss from an analog audio cassette is only audible during quiet passages in music. Companding makes use of this principle by amplifying low-level audio signals prior to recording or transmission and returning them to their correct level afterwards.

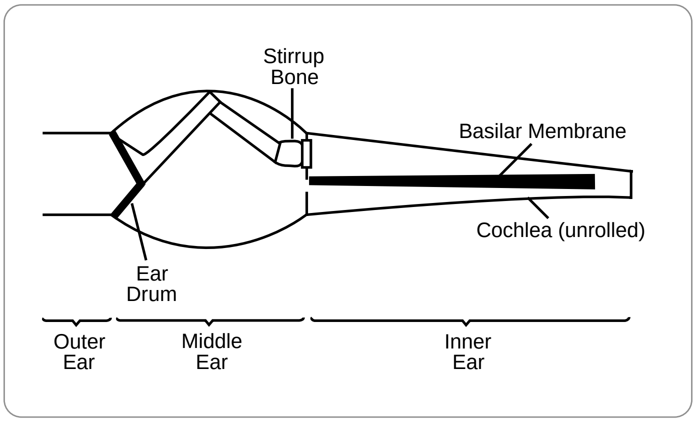


Figure 3-1.

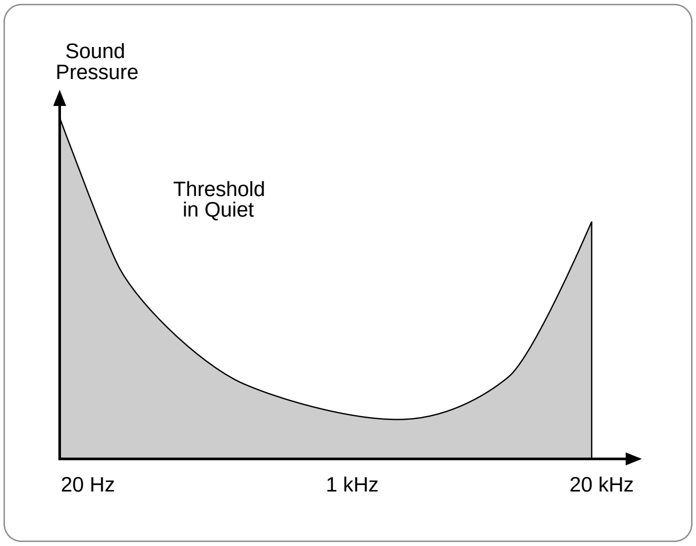


Figure 3-2a.

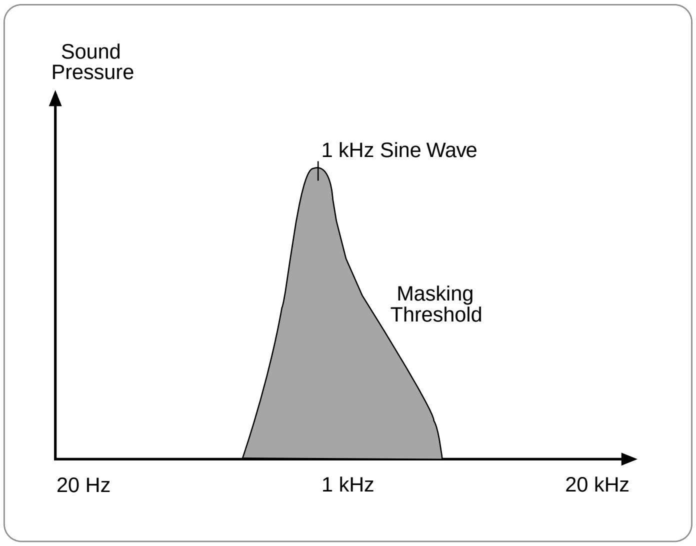


Figure 3-2b.

The imperfect time discrimination of the ear is due to its resonant response. The Q-factor is such that a given sound has to be present for at least about 1 millisecond before it becomes audible. Because of this slow response, masking can still take place even when the two signals involved are not simultaneous. Forward and backward masking occur when the masking sound continues to mask sounds at lower levels before and after the masking sound's actual duration. Figure 3-3 shows this concept.

Masking raises the threshold of hearing, and compressors take advantage of this effect by raising the noise floor, which allows the audio waveform to be expressed with fewer bits. The noise floor can only be raised at frequencies at which there is effective masking. To maximize effective masking, it is necessary to split the audio spectrum into different frequency bands to allow introduction of different amounts of companding and noise in each band.

3.2 Subband Coding

Figure 3-4 shows a band-splitting compander. The band-splitting filter is a set of narrow-band, linear-phase filters that overlap and all have the same bandwidth. The output in each band consists of samples representing a waveform. In each frequency band, the audio input is amplified up to maximum level prior to transmission. Afterwards, each level is returned to its correct value. Noise picked up in the transmission is reduced in each band. If the noise reduction is compared with the threshold of hearing, it can be seen that greater noise can be tolerated in some bands because of masking. Consequently, in each band, after companding it is possible to reduce the wordlength of samples. This technique achieves compression because the quantization noise introduced by the loss of resolution is masked.

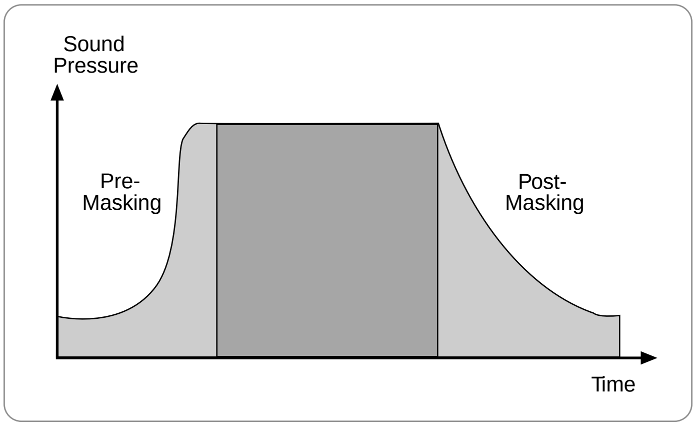


Figure 3-3.

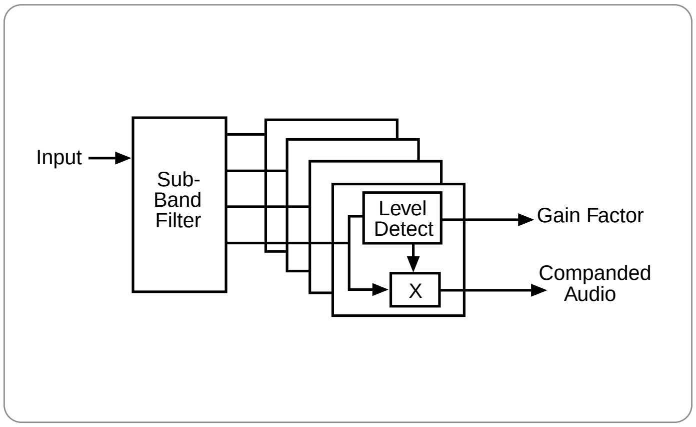


Figure 3-4.

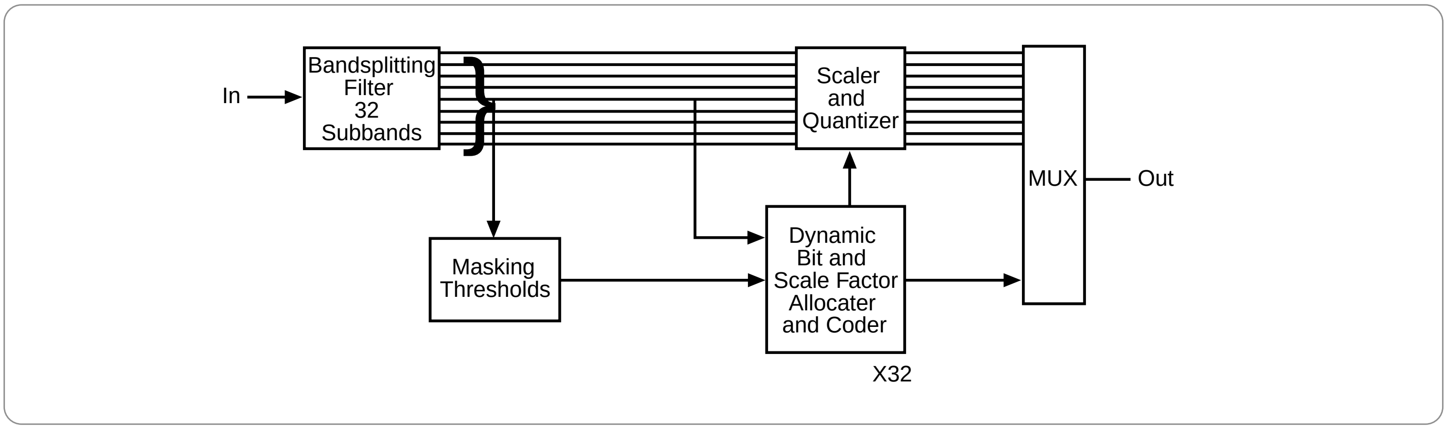

In
Bandsplitting Filter 32 Subbands
Scaler and Quantizer
MUX
Out
Masking Thresholds
Dynamic Bit and Scale Factor Allocater and Coder
X32

Figure 3-5.

Figure 3-5 shows a simple band-splitting coder as is used in MPEG layer 1. The digital audio input is fed to a band-splitting filter that divides the spectrum of the signal into a number of bands. In MPEG, this number is 32. The time axis is divided into blocks of equal length. In MPEG layer 1, there are 384 input samples, so there are 12 samples in each of 32 bands in the output of the filter. Within each band, the level is amplified by multiplication to bring the level up to maximum. The gain required is constant for the duration of a block, and a single scale factor is transmitted with each block for each band in order to allow the process to be reversed at the decoder.

The filter bank output for MPEG layer 1 is analyzed using a 512-point FFT to determine the spectrum of the input signal. This analysis drives a masking model that determines the degree of masking that can be expected in each band. The more masking available, the less accurate the samples in each band need to be. The sample accuracy is reduced by requantizing to reduce wordlength. This reduction is also constant for every word in a band, but different bands can use different wordlengths. The wordlength needs to be transmitted as a bit allocation code for each band to allow the decoder to deserialize the bit stream properly.

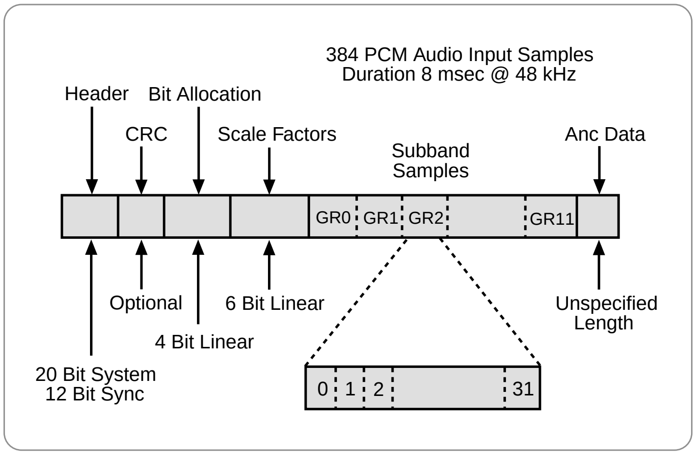


Figure 3-6.

3.3 MPEG Layer 1

Figure 3-6 shows an MPEG layer 1 audio bit stream. Following the synchronizing pattern and the header, there are 32-bit allocation codes of four bits each. These codes describe the wordlength of samples in each subband. Next come the 32 scale factors used in the companding of each band. These scale factors determine the gain needed in the decoder to return the audio to the correct level. The scale factors are followed, in turn, by the audio data in each band.

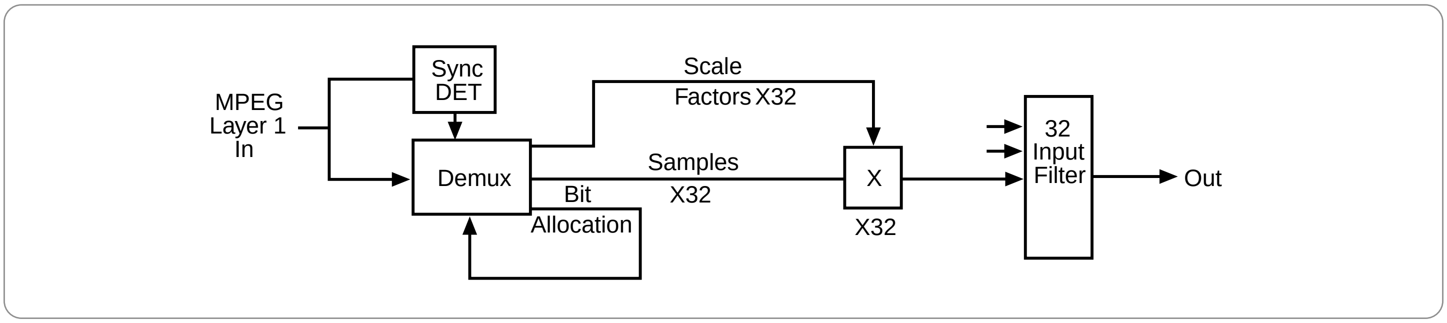


Figure 3-7.

Figure 3-7 shows the layer 1 decoder. The synchronization pattern is detected by the timing generator, which deserializes the bit allocation and scale factor data. The bit allocation data then allows deserialization of the variable length samples. The requantizing is reversed and the compression is reversed by the scale factor data to put each band back to the correct level. These 32 separate bands are then combined in a combiner filter that produces the audio output.

3.4 MPEG Layer 2

Figure 3-8 shows that when the band-splitting filter is used to drive the masking model, the spectral analysis is not very accurate, since there are only 32 bands and the energy could be anywhere in the band. The noise floor cannot be raised very much because, in the worst case shown, the masking may not operate. A more accurate spectral analysis would allow a higher compression factor. In MPEG layer 2, the spectral analysis is performed by a separate process. A 1024-point FFT (Fast Fourier Transform) working directly from the input is used to drive the masking model instead. To resolve frequencies more accurately, the time span of the transform has to be increased, which is done by raising the block size to 1152 samples.

While the block-companding scheme is the same as in layer 1, not all of the scale factors are transmitted, since they contain a degree of redundancy on real program material. The scale factor of successive blocks in the same band differs by 2 dB less than 10 percent of the time, and advantage is taken of this characteristic by analyzing sets of three successive scale factors. When the subband content is approximately constant (known as stationary or quasi-stationary program), only one scale factor out of three is sent. As transient content increases in a given subband, two or three scale factors will be sent. A scale factor select code is also sent to allow the decoder to determine what has been sent in each subband. This technique effectively halves the scale factor bit rate.

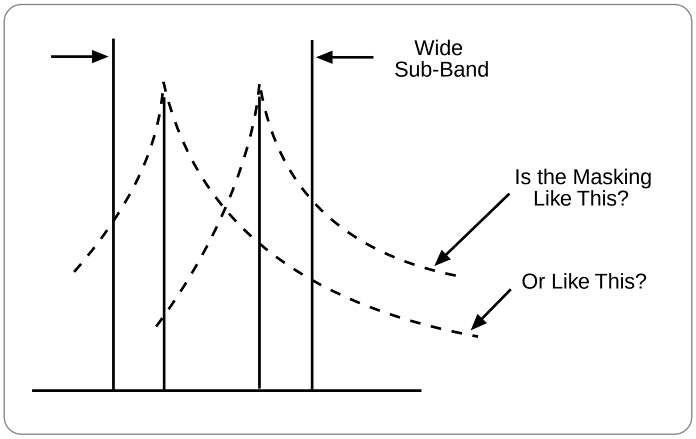


Figure 3-8.

3.5 Transform Coding

Layers 1 and 2 are based on band-splitting filters in which the signal is still represented as a waveform. However, layer 3 adopts transform coding similar to that used in video coding. As was mentioned above, the ear performs a kind of frequency transform on the incident sound and, because of the Q-factor of the basilar membrane, the response cannot increase or reduce rapidly. Consequently, if an audio waveform is transformed into the frequency domain, the coefficients do not need to be sent very often. This principle is the basis of transform coding. For higher compression factors, the coefficients can be requantized, making them less accurate. This process produces noise that will be placed at frequencies where the masking is the greatest. A by-product of a transform coder is that the input spectrum is accurately known, so a precise masking model can be created.

3.6 MPEG Layer 3

This complex level of coding is really only required when the highest compression factor is needed. It has a degree of commonality with layer 2. A discrete cosine transform is used having 384 output coefficients per block. This output can be obtained by direct processing of the input samples, but in a multi-level coder, it is possible to use a hybrid transform incorporating the 32-band filtering of layers 1 and 2 as a basis. If this is done, the 32 subbands from the QMF (quadrature mirror filter) are each further processed by a 12-band MDCT (modified discreet cosine transform) to obtain 384 output coefficients.

Two window sizes are used to avoid pre-echo on transients. The window switching is performed by the psycho-acoustic model. It has been found that pre-echo is associated with the entropy in the audio rising above the average value. To obtain the highest compression factor, non-uniform quantizing of the coefficients is used along with Huffman coding. This technique allocates the shortest wordlengths to the most common code values.

3.7 MPEG-2 Audio

Although originally designated MPEG audio levels 1, 2 and 3, the systems are now more accurately known as MPEG-1 Level 1, etc. MPEG-2 defined extensions to MPEG-1 audio, and a new advanced coding system.

MPEG-2 permits the use of sampling at lower rates than MPEG-1. This is not strictly backward compatible, but requires only additional tables in an MPEG-1 decoder for interoperability.

MPEG-2 BC (backward compatible) audio provides for 5.1 channels (five full-bandwidth channels plus a low-bandwidth low frequency effects channel). MPEG-2 BC has an MPEG-1 (2 channel) bit stream at its core and adds the multi-channel extensions in a form that will be ignored by an MPEG-1 decoder.

MPEG-2 AAC (advanced audio coding) is a more sophisticated system with higher resolution filter banks and additional coding tools. It offers significantly higher coding efficiency, but is not backward compatible.

3.8 MPEG-4 Audio

MPEG-4 coding is based on objects. (See Section 4.4.2.) MPEG-4 audio objects can represent natural or synthetic sounds. For natural audio coding, the MPEG-4 toolkit includes MPEG-2 AAC as well as a variety of other tools. These include parametric encoding for very low bit rates and a technique known as code excited linear predictive (CELP) coding for speech coding in the mid range of bit rates. Various forms of scalability are supported, including bit stream scalability that can be applied at points in the transmission system.

The use of object coding permits choices to be made at the decoding point. For example, a concerto may be transmitted as two objects, orchestra and solo. Normal decoding would present the complete work, but an instrumentalist could decode only the orchestra object and perform the solo part “live.” Similar approaches could permit coding of programs so that listeners could select a “mix minus” mode to eliminate commentary from, say, a sporting event.

MPEG-4’s synthetic audio capabilities will, no doubt, be used extensively in the future. These include “text-to-speech” capabilities and “score driven” techniques where music is synthesized with downloaded instruments using the structured audio orchestra language (SAOL).

3.9 AC-3

The AC-3 audio coding technique, invented by Dolby Laboratories, is used with the ATSC system instead of one of the MPEG audio coding schemes. It is documented as ATSC Standard a/52. Subsequently, AC-3 was adopted as an optional component of DVB, and by the Motorola Digicypher II system. AC-3 is a transform-based system that obtains coding gain by requantizing frequency coefficients.

The PCM input to an AC-3 coder is divided into overlapping windowed blocks as shown in Figure 3-9. These blocks contain 512 samples each, but because of the complete overlap, there is 100 percent redundancy. After the transform, there are 512 coefficients in each block, but because of the redundancy, these coefficients can be decimated to 256 coefficients using a technique called time domain aliasing cancellation (TDAC).

The input waveform is analyzed, and if there is a significant transient in the second half of the block, the waveform will be split into two to prevent pre-echo. In this case, the number of coefficients remains the same, but the frequency resolution will be halved and the temporal resolution will be doubled. A flag is set in the bit stream to indicate to the decoder that this has been done.

The coefficients are output in floating-point notation as a mantissa and an exponent. The representation is the binary equivalent of scientific notation. Exponents are effectively scale factors. The set of exponents in a block produce a spectral analysis of the input to a finite accuracy on a logarithmic scale called the spectral envelope. This spectral analysis is the input to the masking model that determines the degree to which noise can be raised at each frequency.

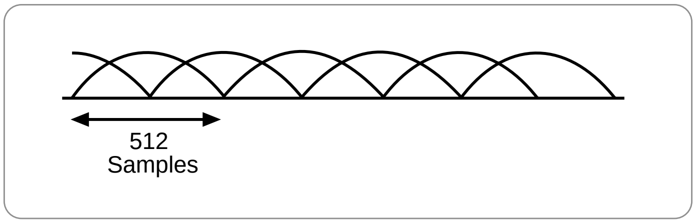


Figure 3-9.

The masking model drives the requantizing process, which reduces the accuracy of each coefficient by rounding the mantissa. A significant proportion of the transmitted data consist of mantissa values.

The exponents are also transmitted, but not directly as there is further redundancy within them that can be exploited. Within a block, only the first (lowest frequency) exponent is transmitted in absolute form. The remaining exponents are transmitted differentially and the decoder adds the difference to the previous value. Where the input audio has a smooth spectrum, the exponents in several frequency bands may be the same. Exponents can be grouped into sets of two or four with flags that describe what has been done.

Sets of six blocks are assembled into an AC-3 sync frame. The first block of the frame always has full exponent data, but in cases of stationary signals, later blocks in the frame can use the same exponents.

Section 4 – The MPEG Standards

Sections 2 and 3 introduced the technologies of video and audio compression, and many of the tools used in the MPEG standards. This section examines the history and structure of MPEG, and the evolution of the various MPEG standards.

4.1 What is MPEG

MPEG is the *Moving Pictures Experts Group,* a committee that comes under the joint control of the *International Standards Organization* (ISO) and the *International Electrotechnical Commission* (IEC). IEC handles international standardization for electrical and electronic technologies; ISO handles virtually everything else. At the start of the information technology age, ISO and IEC formed a joint technical committee (JTC1) to address IT issues. JTC1 has a number of working groups, including JPEG *(Joint Photographic Experts Group)* and WG11, which is MPEG.

The committee was formed in 1988 under the leadership of MPEG convener Dr. Leonardo Chiariglione of Italy. Attendance at MPEG meetings, normally held four times each year, has grown from about 15 delegates in 1988 to some 300 in 2002. It established an enviable track record of generating standards that achieve widespread adoption, MPEG-1, MPEG-2, and the MP3 audio compression standard (MPEG-1 Audio, layer 3). This reputation was somewhat tarnished by MPEG-4, not because of deficiencies in the standard, but as a result of the long delay in publishing licensing terms, and the strong adverse reaction to the first terms that were eventually published in early 2002.

It should be noted that MPEG itself has no role in licensing. As a committee under ISO and IEC, it requires that technology included in its standards be licensable under “reasonable and non-discriminatory terms,” but there is no accepted definition of “reasonable.” Licensing is the responsibility of the holders of the relevant patents, and typically this means many organizations throughout the world that have contributed research and development, and wish to see some recompense.

For MPEG-2, the patent holders grouped together and formed MPEG-LA (MPEG licensing authority). All the essential patents are certified by this group, and are licensed as a block to any organization wishing to implement the standards. This worked well for MPEG-2, but as noted above, greater difficulties are being experienced with MPEG-4, and many hold the delays in publishing licensing terms responsible for the current lack of commercial success of MPEG-4. (This, of course, may change. The MPEG-4 Industry Forum is working hard to find solutions acceptable to patent holders and potential users, and revised proposals released in mid-2002 are likely to be accepted more readily.)

4.2 MPEG-1

The MPEG-1 system, ISO/IEC 11172, is the first international compression standard for motion imagery and was developed between 1988 and 1992. It uses DCT transforms, coefficient quantization, and variable length coding in a similar manner to JPEG, but also includes motion compensation for temporal compression.

It is in three parts:

- System ISO/IEC 11172-1, the multiplex structure.
- ISO/IEC 11172-2, video coding.
- ISO/IEC 11172-3, audio coding.

MPEG-1 represented a remarkable technical achievement. It was designed to compress image streams with SIF picture size, 352x288 (25-Hz PAL) or 352x240 (30Hz NTSC), and associated audio, to approximately 1.5 Mbits/s total compressed data rate. This rate is suitable for transport over T1 data circuits and for replay from CD-ROM, and corresponds approximately to the resolution of a consumer video recorder. A measure of this achievement may be seen by comparing the numbers for an audio CD. A normal audio CD, carrying two-channel audio, at 16-bit resolution with a sampling rate of 44.1 kHz, has a data transfer rate of up to 1.5 Mbit/s. MPEG-1 succeeds in compressing video and audio so that both may be transmitted within the same data rate!

The CIF format is a compromise between European and American SIF (source input format) formats: spatial resolution for 625 SIF (352x288) and Temporal Resolution 525 SIF (29.97 Hz). This is the basis for video conferencing.

MPEG-1 can was designed for CIF images, and has no tools to handle interlaced images, so it had little obvious impact in the world of broadcast television.

Before leaving MPEG-1, it is important to note what is actually included in the standard, and how interoperability is achieved. The standard defines a tool set, the syntax of the bit stream, and the operation of the decoder. It does not define the operation of the encoder – any device that produces a syntactically valid bit stream that can be decoded by a compliant decoder is a valid MPEG encoder. Also, it does not define the quality of the picture, nor encoding quality. This allows for the evolution of encoding technology without change to the standard, and without rendering existing decoders obsolete. This model is used throughout the MPEG standards. The success of this strategy is obvious; although MPEG-2 is used for video, MPEG-1, layer 2 audio is still in use as the principal audio compression system in the DVB transmission systems today.

4.3 MPEG-2

MPEG-1 was frozen (i.e., subsequent changes were allowed to be editorial only) in 1991. In the same year the MPEG-2 process was started, and MPEG-2 eventually became a standard in 1994. The initial goals were simple; there was a need for a standard that would accommodate broadcast quality video width. This required the coding of "full size" standard definition images (704x480 at 29.97 Hz, and 704x576 at 25 Hz), and the ability to code interlaced video efficiently.

In many ways, MPEG-2 represents the "coming of age" of MPEG. The greater flexibility of MPEG-2, combined with the increased availability of large-scale integrated circuits, meant that MPEG-2 could be used in a vast number of applications. The success of MPEG-2 is best highlighted by the demise of MPEG-3, intended for high-definition television. MPEG-3 was soon abandoned when it became clear that MPEG-2 could accommodate this application with ease. MPEG-2 is, of course, the basis for both the ATSC and DVB broadcast standards, and the compression system used by DVD.

MPEG-2 was also permitted to be a moving target. By the use of profiles and levels, discussed below, it was possible to complete the standard for one application, but then to move on to accommodate more demanding applications in an evolutionary manner. Work on extending MPEG-2 continues into 2002.

MPEG-2 is documented as ISO/IEC 13818, currently in 10 parts. The most important parts of this standard are:

- ISO/IEC 13818-1 Systems (transport and programs streams), PES, T-STD buffer model and the basic PSI tables: CAT, PAT, PMT and NIT.
- ISO/IEC 13818-2 video coding.
- ISO/IEC 13818-3 audio coding.
- ISO/IEC 13818-4 MPEG test and conformance.
- ISO/IEC 13818-6 data broadcast and DSMCC.

One of the major achievements of MPEG-2 defined in 13818-1, the transport stream, is described in Section 8. The flexibility and robustness of this design have permitted it to be used for many applications, including transport of MPEG-4 and MPEG-7 data.

Note: DVB and ATSC transport streams carry video and audio PES within "program" groupings, which are entirely different than "program streams" (these are used on DVD & CD).

MPEG Transport Streams are normally constant bit rate but program streams are normally variable bit rate.

4.3.1 Profiles and Levels in MPEG-2

With certain minor exceptions, MPEG-1 was designed for one task; the coding of fixed size pictures and associated audio to a known bit rate of 1.5 Mbits/sec. The MPEG-1 tools and syntax can and have been used for other purposes, but such use is outside the standard, and requires proprietary encoders and decoders. There is only one type of decoder compliant to the MPEG-1 standard.

At the outset, there was a similar goal for MPEG-2. It was intended for coding of broadcast pictures and sound, nominally the 525/60 and 625/50 interlaced television systems. However, as the design work progressed, it was apparent that the tools being developed were capable of handling many picture sizes and a wide range of bit rates. In addition, more complex tools were developed for scalable coding systems. This meant that in practice there could not be a single MPEG-2 decoder. If a compliant decoder had to be capable of handling high-speed bit streams encoded using all possible tools, it would no longer be an economical decoder for mainstream applications. As a simple example, a device capable of decoding high-definition signals at, say, 20 Mbits/sec would be substantially more expensive than one limited to standard-definition signals at around 5 Mbits/sec. It would be a poor standard that required the use of an expensive device for the simple application.

MPEG devised a two-dimensional structure of profiles and levels for classifying bit streams and decoders. Profiles define the tools that may be used. For example, bidirectional encoding (B-frames) may be used in the main profile, but not in simple profile. Levels relate just to scale. A high level decoder must be capable of receiving a faster bit stream, and must have more decoder buffer and larger frame stores than a main level decoder. However, main profile at high level (MP@HL) and main profile at main level (MP@ML) use exactly the same encoding/decoding tools and syntax elements.

Figure 4-1 shows the pairings of profile and level that are defined by MPEG-2 (Profiles on the horizontal axis, Levels on the vertical axis). Note that not all combinations are valid; only the completed pairings are defined in the standard. It is a requirement of conformity to the standard that a decoder at any profile/level must be able to decode lower profiles and levels. For example, an MP@ML decoder must be able to decode main profile at low level (MP@LL) and simple profile at main level (SP@ML) bit streams.

HIGH		4:2:0 1920x1152 80 Mbps I, P, B	4:2:2 1920x1080 300 Mbps I, P, B			4:2:0, 4:2:2 1920x1152 100 Mbps I, P, B
HIGH – 1440		4:2:0 1440x1152 60 Mbps I, P, B			4:2:0 1440x1152 60 Mbps I, P, B	4:2:0, 4:2:2 1440x1152 80 Mbps I, P, B
MAIN	4:2:0 720x576 15 Mbps I, P	4:2:0 720x576 15 Mbps I, P, B	4:2:2 720x608 50 Mbps I, P, B	4:2:0 720x576 15 Mbps I, P, B		4:2:0, 4:2:2 720x576 20 Mbps I, P, B
LOW		4:2:0 352x288 4 Mbps I, P, B		4:2:0 352x288 4 Mbps I, P, B		
LEVEL PROFILE	SIMPLE	MAIN	4:2:2 PROFILE	SNR	SPATIAL	HIGH

Figure 4-1.

The simple profile does not support bidirectional coding, and so only I- and P-pictures will be output. This reduces the coding and decoding delay and allows simpler hardware. The simple profile has only been defined at main level.

The Main Profile is designed for a large proportion of uses. The low level uses a low-resolution input having only 352 pixels per line. The majority of broadcast applications will require the MP@ML subset of MPEG, which supports SDTV (standard definition TV).

The high-1440 level is a high definition scheme that doubles the definition compared to the main level. The high level not only doubles the resolution but maintains that resolution with 16:9 format by increasing the number of horizontal samples from 1440 to 1920.

In compression systems using spatial transforms and requantizing, it is possible to produce scalable signals. A scalable process is one in which the input results in a main signal and a "helper" signal. The main signal can be decoded alone to give a picture of a certain quality, but if the information from the helper signal is added, some aspect of the quality can be improved.

For example, a conventional MPEG coder, by heavily requantizing coefficients, encodes a picture with moderate signal-to-noise ratio results. If, however, that picture is locally decoded and subtracted pixel-by-pixel from the original, a quantizing noise picture results. This picture can be compressed and transmitted as the helper signal. A simple decoder only decodes the main, noisy bit stream, but a more complex decoder can decode both bit streams and combine them to produce a low-noise picture. This is the principle of SNR (signal-to-noise ratio) scalability.

As an alternative, coding only the lower spatial frequencies in a HDTV picture can produce a main bit stream that an SDTV receiver can decode. If the lower definition picture is locally decoded and subtracted from the original picture, a definition-enhancing picture would result. This picture can be coded into a helper signal. A suitable decoder could combine the main and helper signals to recreate the HDTV picture. This is the principle of spatial scalability.

The high profile supports both SNR and spatial scalability as well as allowing the option of 4:2:2 sampling.

The 4:2:2 profile has been developed for improved compatibility with digital production equipment. This profile allows 4:2:2 operation without requiring the additional complexity of using the high profile. For example, an HP@ML decoder must support SNR scalability, which is not a requirement for production. The 4:2:2 profile has the same freedom of GOP structure as other profiles, but in practice it is commonly used with short GOPs making editing easier. 4:2:2 operation requires a higher bit rate than 4:2:0, and the use of short GOPs requires an even higher bit rate for a given quality.

The concept of profiles and levels is another development of MPEG-2 that has proved to be robust and extensible; MPEG-4 uses a much more complex array of profiles and levels, to be discussed later.

4.4 MPEG-4

International standardization is a slow process, and technological advances often occur that could be incorporated into a developing standard. Often this is desirable, but continual improvement can mean that the standard never becomes final and usable. To ensure that a standard is eventually achieved there are strict rules that prohibit substantive change after a certain point in the standardization process. So, by the time a standard is officially adopted there is often a backlog of desired enhancements and extensions. So it was with MPEG-2. As discussed above, MPEG-3 had been started and abandoned, so the next project became MPEG-4. Two versions of MPEG-4 are already complete, and work is continuing on further extensions.

At first the main focus of MPEG-4 was the encoding of video and audio at very low rates. In fact, the standard was explicitly optimized for three bit rate ranges:

- Below 64 kbits/s.
- 64 to 384 kbits/s.
- 384 kbits/s to 4 Mbits/s.

Performance at low bit rates remained a major objective and some very creative ideas contributed to this end. Great attention was also paid to error resilience, making MPEG-4 very suitable for use in the error-prone environments, such as transmission to personal handheld devices. However, other profiles and levels use bit rates up to 38.4 Mbits/s, and work is still proceeding on studio-quality profiles and levels using data rates up to 1.2 Gbits/s.

More importantly, MPEG-4 became vastly more than just another compression system – it evolved into a totally new concept of multimedia encoding with powerful tools for interactivity and a vast range of applications. Even the official “overview” of this standard spans 67 pages, so only a brief introduction to the system is possible here.

4.4.1 MPEG-4 Standards Documents

The principal parts of the MPEG-4 standards are:

- ISO/IEC 14496-1 Systems.
- ISO/IEC 14496-2 Visual.
- ISO/IEC 14496-3 Audio.
- ISO/IEC 14496-4 Conformance Testing.
- ISO/IEC 14496-6 Delivery Multimedia Integration Framework (DMIF).

4.4.2 Object Coding

The most significant departure from conventional transmission systems is the concept of objects. Different parts of the final scene can be coded and transmitted separately as video objects and audio objects to be brought together, or composited, by the decoder. Different object types may each be coded with the tools most appropriate to the job. The objects may be generated independently, or a scene may be analyzed to separate, for example, foreground and background objects. In one interesting demonstration, video coverage of a soccer game was processed to separate the ball from the rest of the scene. The background (the scene without the ball) was transmitted as a “teaser” to attract a pay-per-view audience. Anyone could see the players and the field, but only those who paid could see the ball!

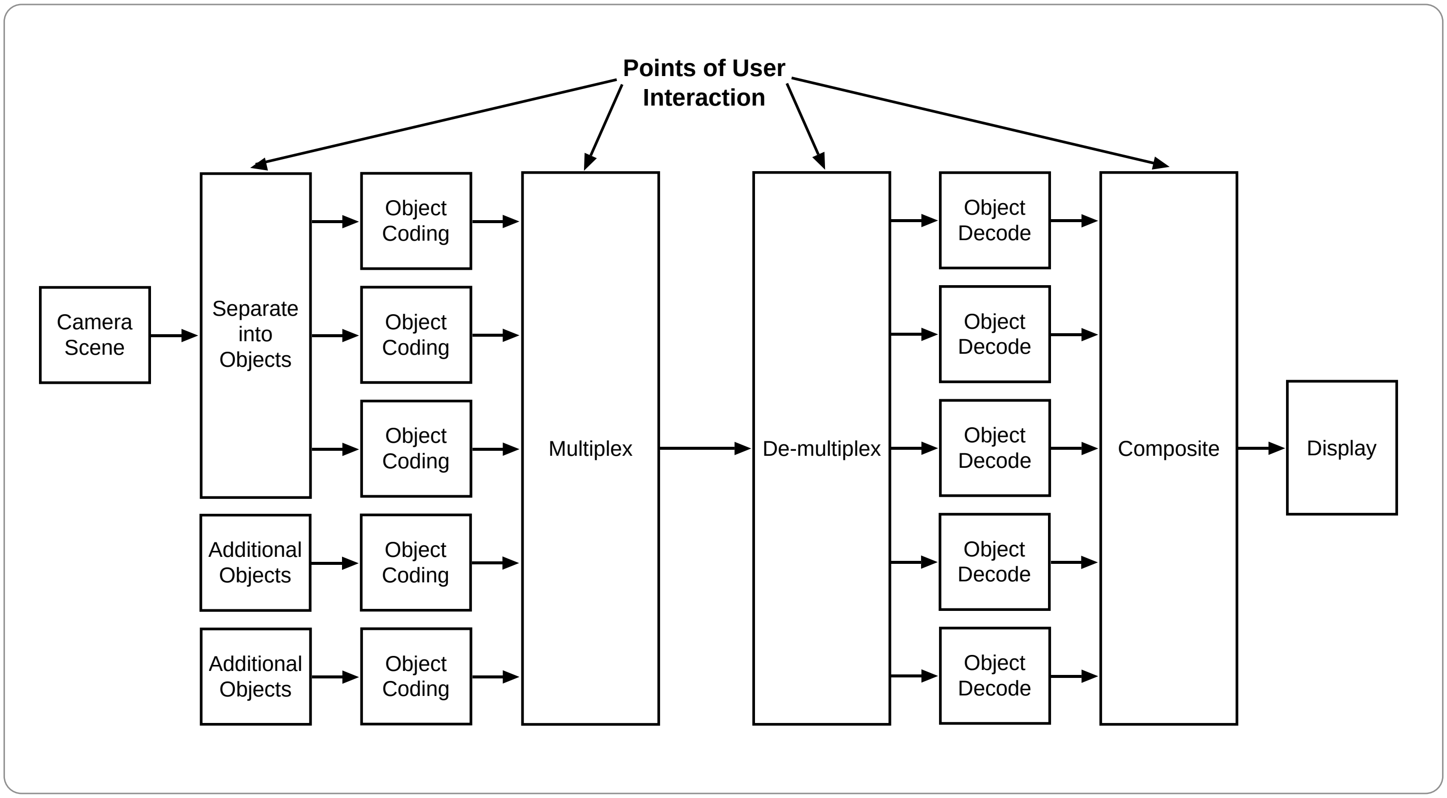


Figure 4-2.

The object-oriented approach leads to three key characteristics of MPEG-4 streams:

- Multiple objects may be encoded using different techniques, and composited at the decoder.
- Objects may be of natural origin, such as scenes from a camera, or synthetic, such as text.
- Instructions in the bit stream, and/or user choice, may enable several different presentations from the same bit stream.

The generalized system for object coding in MPEG-4 is shown in Figure 4-2. This diagram also emphasizes the opportunities for user interaction within MPEG-4 systems – a powerful feature, particularly for video game designers.

These capabilities do not have to be used – MPEG-4 provides traditional coding of video and audio, and improves on MPEG-2 by offering improved efficiency and resilience to errors. However, the true power of MPEG-4 comes from the architecture described above. The coding of objects independently offers a number of advantages. Each object may be coded on the most efficient manner, and different spatial or temporal scaling (see 4.4.3) may be used as appropriate.

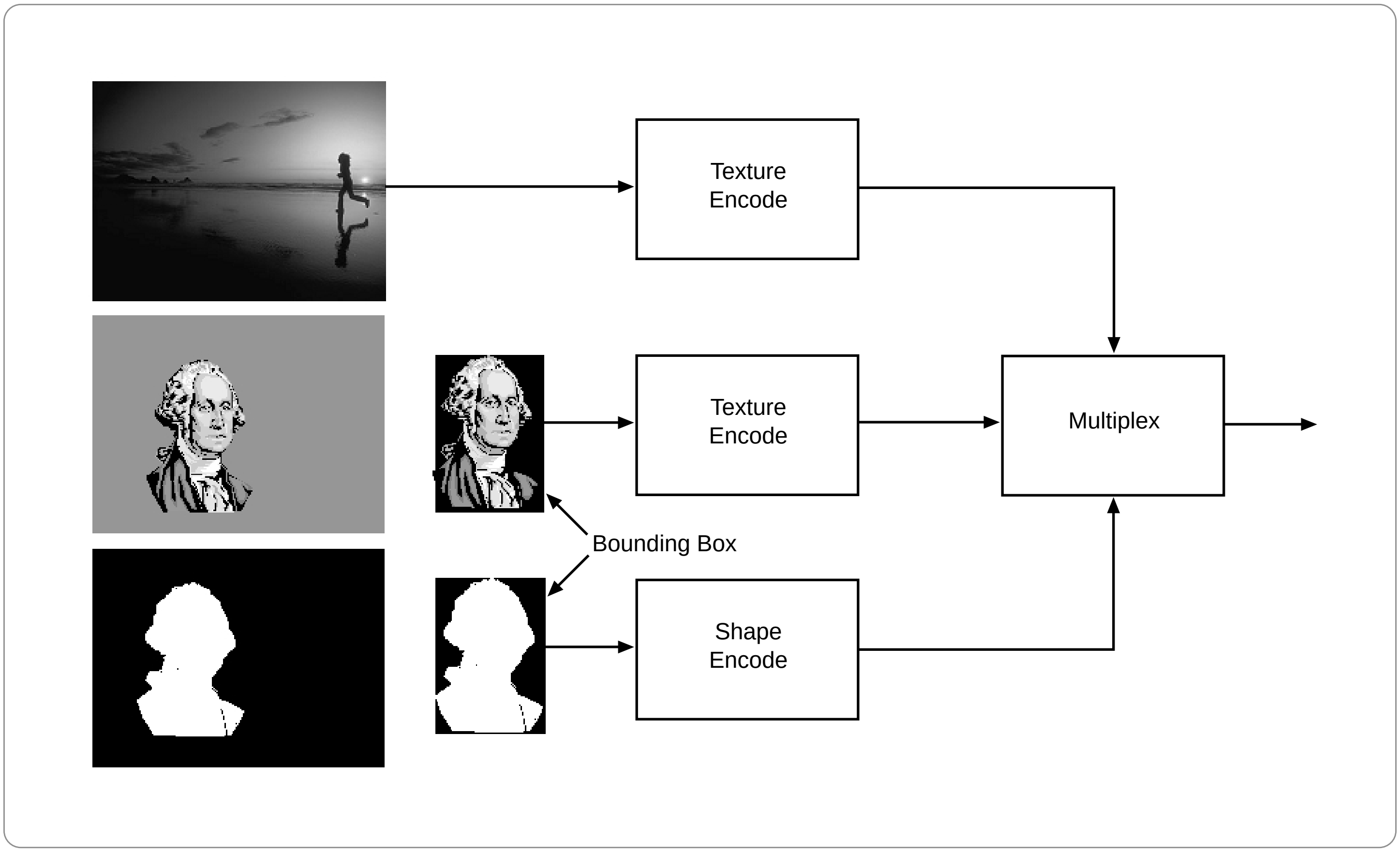


Figure 4-3.

4.4.3 Video and Audio Coding

Many of the video coding tools in MPEG-4 are similar to those of MPEG-2, but enhanced by better use of predictive coding and more efficient entropy coding. However, the application of the tools may differ significantly from earlier standards.

MPEG-4 codes video objects. In the simplest model a video is coded in much the same way as in MPEG-2, but it is described as a single video object with a rectangular shape. The representation of the image is known as texture coding. Where there is more than one video object, some may have irregular shapes, and generally all will be smaller than a full-screen background object. This means that only the active area of the object need be coded, but the shape and position must also be represented. The standard includes tools for shape coding of rectangular and irregular objects, in either binary or gray-scale representations (similar to an alpha channel). The concept is shown in Figure 4-3.

Similarly, MPEG-4 uses tools similar to those of MPEG-1 and MPEG-2 for coding live audio, and AAC offers greater efficiency. Multiple audio “objects” may be encoded separately and composited at the decoder. As with video, audio objects may be natural or synthetic.

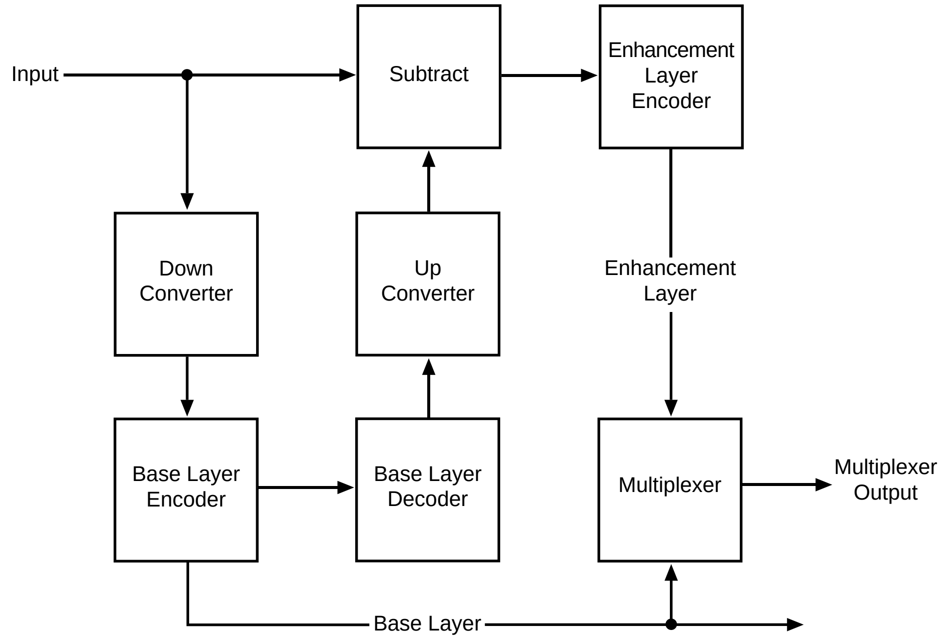


Figure 4-4.

4.4.4 Scalability

In the context of media compression, scalability means the ability to distribute content at more than one quality level within the same bit stream. MPEG-2 and MPEG-4 both provide scalable profiles using a conventional model; the encoder generates a base-layer and one or more enhancement layers, as shown in Figure 4-4. The enhancement layer(s) may be discarded for transmission or decoding if insufficient resources are available. This approach works, but all decisions about quality levels have to be made at the time of encoding, and in practice the number of enhancement layers is severely limited (usually to one).

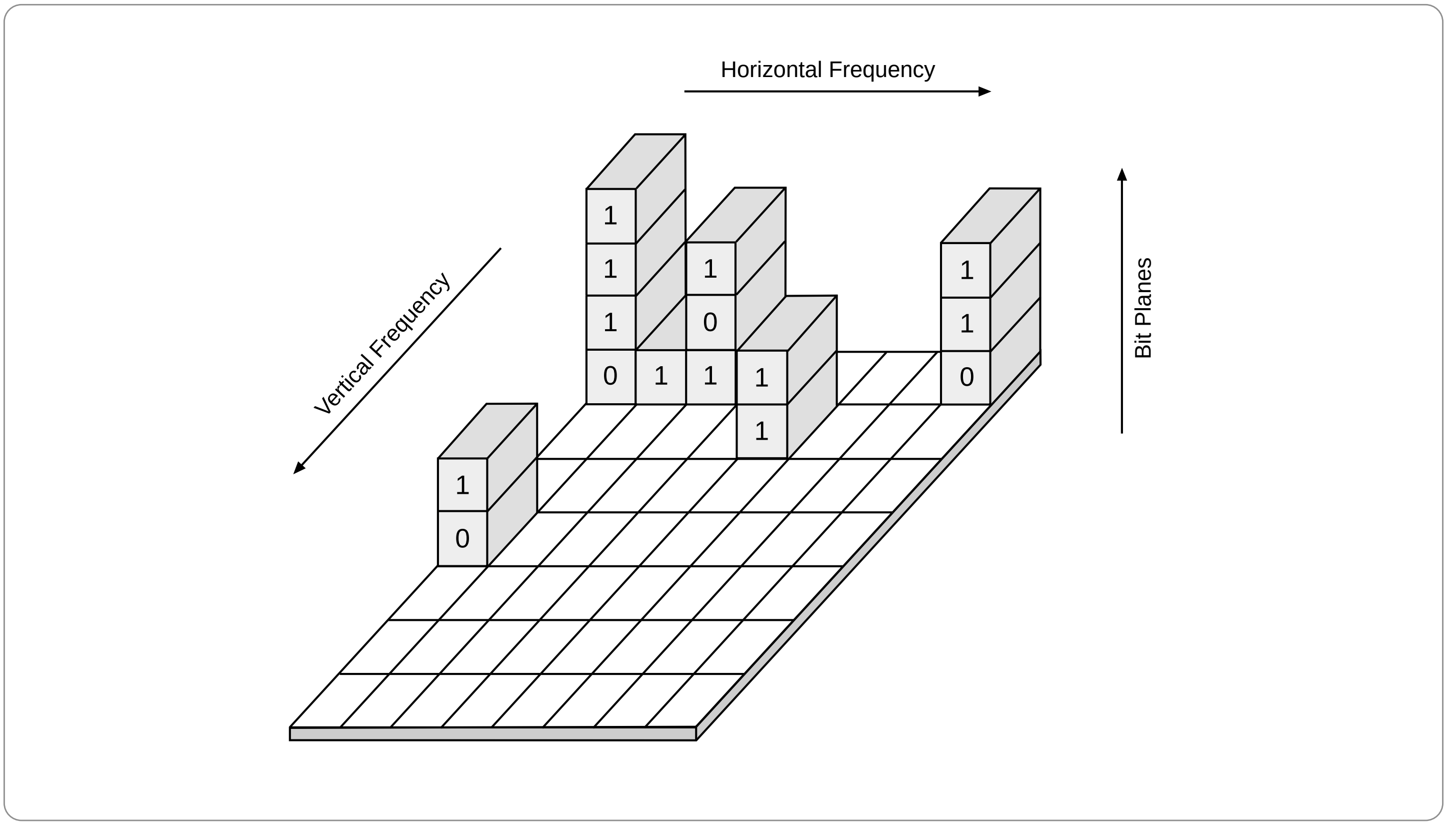


Figure 4-5.

Later versions of MPEG-4 include the fine grain scalability (FGS) profile. This technique generates a single bit stream representing the highest quality level, but that allows for lower quality versions to be extracted downstream. FGS uses bit-plane encoding, shown in concept in Figure 4-5. The quantized coefficients are "sliced" one bit at a time, starting with the most significant bit. This provides a coarse representation of the largest (and most significant) coefficient(s). Subsequent slices provide more accurate representations of these most-significant coefficients, and coarse approximations of the next most significant – and so on.

Spatial scaling, including FGS, may be combined with temporal scaling that permits the transmission and/or decoding of lower frame rates when resources are limited. As mentioned above, objects may be scaled differently; it may be appropriate to retain full temporal resolution for an important foreground object, but to update to the background as a lower rate.

4.4.5 Other Aspects of MPEG-4

MPEG-4 is enormous, and the comments above just touch on a few of the many aspects of the standard. There are studio profiles for high-quality encoding which, in conjunction with object coding, will permit structured storage of all the separate elements of a video composite. Further extensions of MPEG-4 may even provide quality levels suitable for digital cinema. Figure 4-6 shows the MPEG-4 profiles defined today. (Note that this diagram shows only the profiles; generally multiple levels are defined for each profile.)

Some of the object types defined within MPEG-4 are interesting. One example is a *sprite*. A sprite is a static background object, generally larger than the viewing port or display device. For example, the action of a video game may take place in front of a background scene. If a sprite is used, a large static background may be transmitted once, and as the game action proceeds the appropriate part of the background will be seen, according to the motion of the viewport.

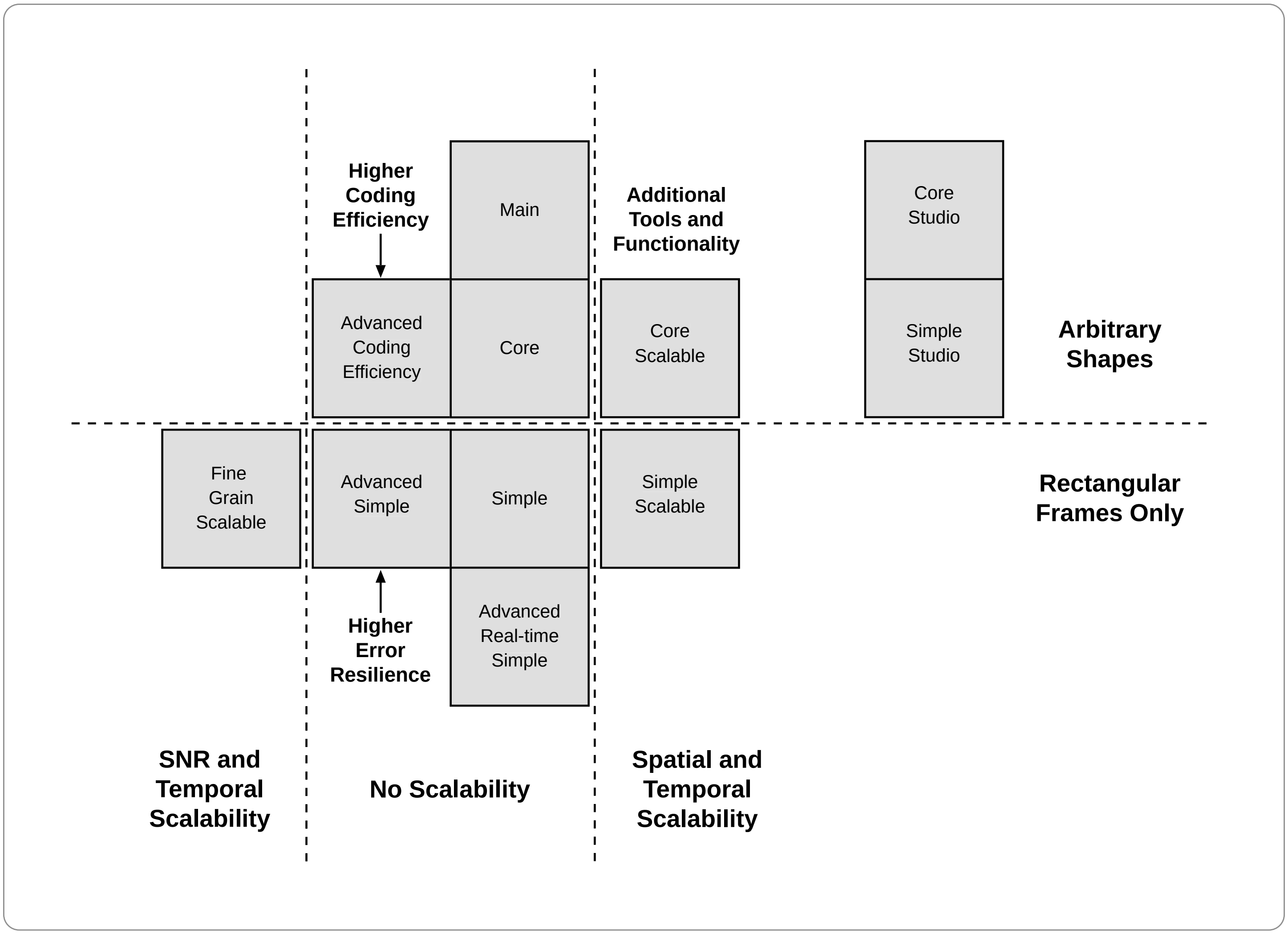


Figure 4-6.

MPEG-4 defines both facial and body animation profiles. In each case a default face or body may be used, and commands sent to animate this object. Alternatively, the default object may be modified by the bit stream; for example, a specific face may be transmitted and then animated. Sophisticated animation commands related to language will permit a stored face to "read" text in many languages.

Some describe MPEG-4 as the standard for video games, and certainly many of the constructs are ideally suited to that industry. However, even a cursory examination of the standard reveals such a wealth of capabilities, and such depth in every aspect, that the potential applications are endless.

4.4.6 The Future of MPEG-4

As discussed above, MPEG-4 is a wide-ranging set of standards with a rich offering of capabilities for many applications. This is the theory; in practice, MPEG-4 can show few successes. In particular, many observers expected that MPEG-4 would quickly become the dominant coding mechanism for audio-visual material transmitted over the Internet and replace the various proprietary codecs in use today. This has not happened, nor is there any likelihood that it will happen in the near future. There are two main reasons for this failure.

The first is technology, and the resulting performance. MPEG-4 uses video compression technology based on the ITU-developed H.26x, dating from the early 1990s. The distribution of audio and video over the Internet is a fiercely competitive business, and all the major players, Apple, Microsoft and RealNetworks, have implemented proprietary video encoding schemes that outperform the current MPEG-4 codec.

The other reason for the failure (to date) of MPEG-4 is the patent licensing situation. Until early 2002, companies wishing to implement MPEG-4 did not know what royalties they would need to pay to the patent holders. The proposed licensing scheme for the basic levels of MPEG-4 has now been published, and met with strong adverse reaction from the industry. Licensing terms for the more sophisticated levels are still unknown. Certainly the initial offering of licensing terms has done nothing to increase mainstream implementation of the standard.

There is hope for the future. A joint effort of ITU and MPEG, known as the joint video team (JVT) is working on a codec known as H.26L. To quote the ITU, “The H.26L design is a block-based motion-compensated hybrid transform coder – similar in spirit but different in many specifics relative to prior designs. … H.26L significantly increases the number of available block sizes and the number of available reference pictures for performing motion estimation.” The new codec also offers much greater precision in motion estimation (1/8 pixel in some implementations), and is based on a principal block size of 4x4, rather than the 8x8 used in most MPEG systems.

H.26L is expected to show substantial improvements in coding efficiency, and it is the goal of the participants that the base level, suitable for Internet streaming, will be royalty-free. The first stage of the work of JVT is expected to be completed in 2002, and published as MPEG-4 Part 10.

4.5 MPEG-7

Because MPEG-3 was cancelled, the sequence of actual standards was MPEG-1, MPEG-2, and MPEG-4. Some committee participants wanted the next standard to be MPEG-5; others were attracted by the binary nature of the sequence and preferred MPEG-8. Finally, it was concluded that any simple sequence would fail to signal the fundamental difference from the work of MPEG-1 through MPEG-4, and MPEG-7 was chosen.

MPEG-7 is not about compression; it is about *metadata*, also known as the “bits about the bits.” Metadata is digital information that describes the content of other digital data. In modern parlance, the program material or content, the actual image, video, audio or data objects that convey the information, are known as *data essence*. The metadata tells the world all it needs to know about what is in the essence.

Anyone who has been involved with the storage of information, be it videotapes, books, music, whatever, knows the importance and the difficulty of accurate cataloging and indexing. Stored information is useful only if its existence is known, and if it can be retrieved in a timely manner when needed.

This problem has always been with us, and is addressed in the analog domain by a combination of labels, catalogs, card indexes, etc. More recently, the computer industry has given us efficient, cost-effective, relational databases that permit powerful search engines to access stored information in remarkable ways. Provided, that is, the information is present in a form that the search engine can use.

Here is the real problem. The world is generating new media content at an enormous and ever-increasing rate. With the increasing quantity and decreasing cost of digital storage media, more and more of this content can be stored. Local and wide-area networks can make the content accessible and deliverable if it can be found. The search engines can find what we want, and the databases can be linked to the material itself, but we need to get all the necessary indexing information into the database in a form suitable for the search engine.

We might guess from knowledge of earlier standards that the MPEG committee would not concern itself unduly with mechanisms for generating data. MPEG rightly takes the view that if it creates a standardized structure, and if there is a market need, the technological gaps will be filled. In previous MPEG standards, the syntax and the decoder were specified by the standard. In MPEG-7, *only* the syntax is standardized, as shown in Figure 4-7. The generation of the metadata is unspecified, as are the applications that may use it. MPEG-7 specifies how metadata should be expressed. This means that the fields that should go into a database are specified, and anyone designing a search engine knows what descriptive elements may be present, and how they will be encoded.

MPEG-7 defines a structure of *descriptors* and *description schemes* that can characterize almost anything. In theory at least, primitive elements such as color histograms and shapes can be combined to represent complex entities such as individual faces. It may be possible to index material automatically such that the database can be searched for scenes that show, for example, President Clinton and U.S. Federal Reserve Chairman Greenspan together. The constructs are not confined to images. It should be possible to use a voice sample to search for recordings by, or images of, Pavarotti, or to play a few notes on a keyboard to find matching or similar melodies.

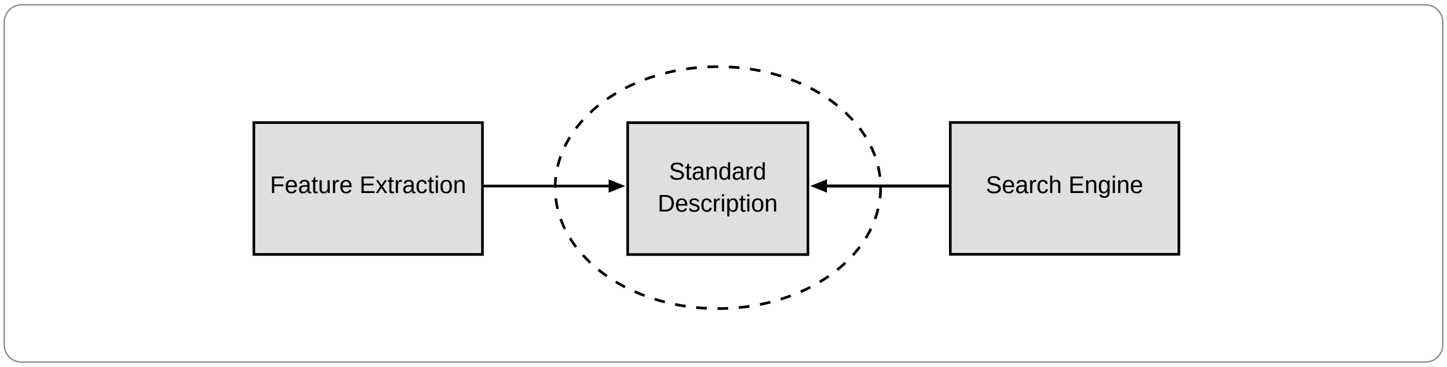


Figure 4-7.

The rapid advance of storage and networking systems will enable access to vast quantities of digital content. As technology advances to satisfy the needs of MPEG-7, we will be able to index and retrieve items in ways unimaginable a few years ago. We will then need a system to control access, privacy, and commercial transactions associated with this content. This is where MPEG-21 is targeted.

4.6 MPEG-21

MPEG-21 again differs in kind from the earlier work of the committee. The basic concept is fairly simple – though wide reaching. MPEG-21 seeks to create a complete structure for the management and use of digital assets, including all the infrastructure support for the commercial transactions and rights management that must accompany this structure. The vision statement is "to enable transparent and augmented use of multimedia resources across a wide range of networks and devices."

The scope of the MPEG-21 work is indicated by the seven architectural elements defined in the draft technical report.

1. The *digital item declaration* is expected to "establish a uniform and flexible abstraction and interoperable schema for defining digital items." The scheme must be open and extensible to any and all media resource types and description schemes, and must support a hierarchical structure that is easy to search and navigate.
2. The *digital item representation* of MPEG-21 is the technology that will be used to code the content and to provide all the mechanisms needed to synchronize all the elements of the content. It is expected that this layer will reference at least MPEG-4.
3. *Digital item identification & description* will provide the framework for the identification and description of digital items (linking all content elements). This will likely include the description schemes of MPEG-7, but must also include "[a] new generation of identification systems to support effective, accurate and automated event management and reporting (license transactions, usage rules, monitoring and tracking, etc)." It must satisfy the needs of all classes of MPEG-21 users.
4. *Content management and usage* must define interfaces and protocols for storage and management of MPEG-21 digital items and descriptions. It must support archiving and cataloguing of content while preserving usage rights, and the ability to track changes to items and descriptions. This element of MPEG-21 will likely also support a form of "trading" where consumers can exchange personal information for the right to access content, and formalization of mechanisms for "personal channels" and similar constructs.
5. *Intellectual property management and protection* is an essential component. The current controversies surrounding MP3 audio files demonstrate the need for new copyright mechanisms cognizant of the digital world. It can be argued that content has no value unless it is protected. MPEG-21 will build on the ongoing work in MPEG-4 and MPEG-7, but will need extensions to accommodate new types of digital items, and new delivery mechanisms.
6. MPEG-21 *terminals and networks* will address delivery of items over a wide range of networks, and the ability to render the content on a wide range of terminals. Conceptually a movie should be deliverable in full digital-cinema quality to a movie theater, or at a lower quality over a slower network to a consumer device (at a different price). In either case there will be some restriction on the type and or number of uses. The user should not need to be aware of any issues or complexities associated with delivery or rendering.
7. Finally, there is a need for *event reporting* to "standardize metrics and interfaces for performance of all reportable events." The most obvious example here is that if the system allows a user access to a protected item, it must also ensure that the appropriate payment is made!

Section 5 – Elementary Streams

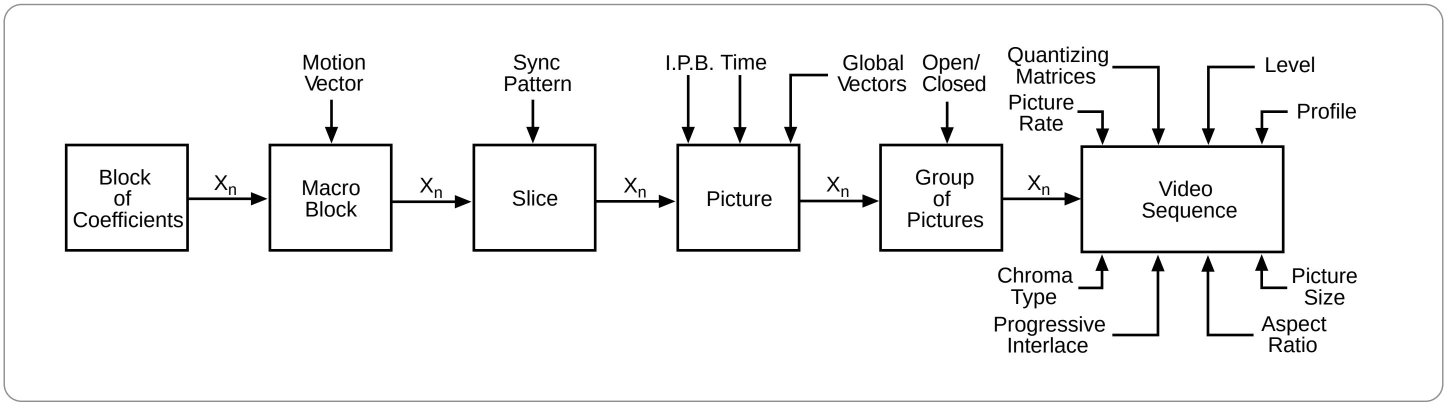


Figure 5-1.

An elementary stream is basically the raw output of an encoder and contains no more than is necessary for a decoder to approximate the original picture or audio. The syntax of the compressed signal is rigidly defined in MPEG so that decoders can be guaranteed to operate on it. The encoder is not defined except that it must somehow produce the right syntax.

The advantage of this approach is that it suits the real world in which there are likely to be many more decoders than encoders. By standardizing the decoder, they can be made at low cost. In contrast, the encoder can be more complex and more expensive without a great cost penalty, but with the potential for better picture quality as complexity increases. When the encoder and the decoder are different in complexity, the coding system is said to be asymmetrical.

The MPEG approach also allows for the possibility that quality will improve as coding algorithms are refined while still producing bit streams that can be understood by earlier decoders. The approach also allows the use of proprietary coding algorithms, which need not enter the public domain.

5.1 Video Elementary Stream Syntax

Figure 5-1 shows the construction of the elementary video stream. The fundamental unit of picture information is the DCT (discrete cosine transform) block, which represents an 8x8 array of pixels that can be Y, C_b or C_r. The DC coefficient is sent first and is represented more accurately than the other coefficients. Following the remaining coefficients, an end of block (EOB) code is sent.

Blocks are assembled into macroblocks, which are the fundamental units of a picture and which can be motion compensated. Each macroblock has a two-dimensional motion vector in the header. In B-pictures, the vectors can be backward as well as forward. The motion compensation can be field or frame based and this is indicated. The scale used for coefficient requantizing is also indicated. Using the vectors, the decoder obtains information from earlier and later pictures to produce a predicted picture. The blocks are inverse-transformed to produce a correction picture that is added to the predicted picture to produce the decoded output. In 4:2:0 coding, each macroblock will have four Y blocks and two color-difference blocks. To make it possible to identify which block describes which component, the blocks are sent in a specified order.

Macroblocks are assembled into slices that must always represent horizontal strips of picture from left to right. In MPEG, slices can start anywhere and be of arbitrary size, but in ATSC they must start at the left-hand edge of the picture. Several slices can exist across the screen width. The slice is the fundamental unit of synchronization for variable length and differential coding. The first vectors in a slice are sent absolutely, whereas the remaining vectors are transmitted differentially. In I-pictures, the first DC coefficients in the slice are sent absolutely and the remaining DC coefficients are transmitted differentially. In difference pictures, correlation of these coefficients is not to be expected, and this technique is not appropriate.

In the case of a bit error in the elementary stream, either the deserialization of the variable length symbols will break down, or subsequent differentially coded coefficients or vectors will be incorrect. The slice structure allows recovery by providing a resynchronizing point in the bit stream.

A number of slices are combined to make a picture that is the active part of a field or a frame. The picture header defines whether the picture was I, P or B coded and includes a temporal reference so that the picture can be presented at the correct time. In the case of pans and tilts, the vectors in every macroblock will be the same. A global vector can be sent for the whole picture, and the individual vectors then become differences from this global value.

Pictures may be combined to produce a GOP that must begin (in transmission order) with an I-picture. The GOP is the fundamental unit of temporal coding. In the MPEG standard, the use of a GOP is optional, but it is a practical necessity. Between I-pictures, a variable number of P- and/or B-pictures may be placed as was described in Section 2. A GOP may be open or closed. In a closed GOP, the last B-pictures do not require the I-picture in the next GOP for decoding and the bit stream could be cut at the end of the GOP.

If GOPs are used, several GOPs may be combined to produce a video sequence. The sequence begins with a sequence start code, followed by a sequence header and ends with a sequence end code. Additional sequence headers can be placed throughout the sequence. This approach allows decoding to begin part way through the sequence, as might happen in playback of digital video disks and tape cassettes. The sequence header specifies the vertical and horizontal size of the picture, the aspect ratio, the chroma subsampling format, the picture rate, the use of progressive scan or interlace, the profile, level, and bit rate, and the quantizing matrices used in intra and inter-coded pictures.

Without the sequence header data, a decoder cannot understand the bit stream, and therefore sequence headers become entry points at which decoders can begin correct operation. The spacing of entry points influences the delay in correct decoding that may occur when the viewer switches from one television channel to another.

5.2 Audio Elementary Streams

Various types of audio can be embedded in an MPEG-2 multiplex. These types include audio coded according to MPEG layers 1, 2, 3, or AC-3. The type of audio encoding used must be included in a descriptor that a decoder will read in order to invoke the appropriate type of decoding.

The audio compression process is quite different from the video process. There is no equivalent to the different I, P, and B frame types, and audio frames always contain the same amount of audio data. There is no equivalent of bidirectional coding and audio frames are not transmitted out of sequence.

In MPEG-2 audio, the descriptor in the sequence header contains the layer that has been used to compress the audio and the type of compression used (for example, joint stereo), along with the original sampling rate. The audio sequence is assembled from a number of access units (AUs) that will be coded audio frames.

If AC-3 coding is used, as in ATSC, this usage will be reflected in the sequence header. The audio access unit (AU) is an AC-3 sync frame as described in Section 3.7. The AC-3 sync frame represents a time span equivalent of 1536 audio samples and will be 32 ms for 48-kHz sampling and 48 ms for 32 kHz.

Section 6 – Packetized Elementary Streams (PES)

For practical purposes, the continuous elementary streams carrying audio or video from compressors need to be broken into packets. These packets are identified by headers that contain time stamps for synchronizing. PES packets can be used to create Program Streams or Transport Streams.

6.1 PES Packets

In the PES, an endless elementary stream is divided into packets of a convenient size for the application. This size might be a few hundred kilobytes, although this would vary with the application.

Each packet is preceded by a PES packet header. Figure 6-1 shows the contents of a header. The packet begins with a start-code prefix of 24 bits and a stream ID that identifies the contents of the packet as video or audio and further specifies the type of audio coding. These two parameters (start code prefix and stream ID) comprise the packet start code that identifies the beginning of a packet. It is important not to confuse the packet in a PES with the much smaller packet used in transport streams that, unfortunately, shares the same name.

Because MPEG only defines the transport stream, not the encoder, a designer might choose to build a multiplexer that converts from elementary streams to a transport stream in one step. In this case, the PES packets may never exist in an identifiable form, but instead, they are logically present in the Transport Stream payload.

6.2 Time Stamps

After compression, pictures are sent out of sequence because of bidirectional coding. They require a variable amount of data and are subject to variable delay due to multiplexing and transmission. In order to keep the audio and video locked together, time stamps are periodically incorporated in each picture.

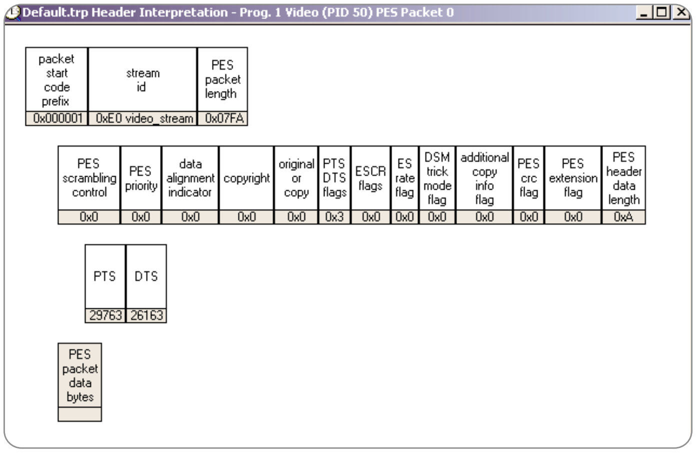
A time stamp is a 33-bit number that is a sample of a counter driven by a 90-kHz clock. This clock is obtained by dividing the 27-MHz program clock by 300. Since presentation times are evenly spaced, it is not essential to include a time stamp in every presentation unit. Instead, time stamps can be interpolated by the decoder, but they must not be more than 700 ms apart in either program streams or transport streams.

Time stamps indicate where a particular access unit belongs in time. Lip sync is obtained by incorporating time stamps into the headers in both video and audio PES packets. When a decoder receives a selected PES packet, it decodes each access unit and buffers it into RAM. When the time-line count reaches the value of the time stamp, the RAM is read out. This operation has two desirable results. First, effective timebase correction is obtained in each elementary stream. Second, the video and audio elementary streams can be synchronized together to make a program.

6.3 PTS/DTS

When bidirectional coding is used, a picture may have to be decoded some time before it is presented so that it can act as the source of data for a B-picture. Although, for example, pictures can be presented in the order IBBP, they will be transmitted in the order IPBB. Consequently, two types of time stamp exist. The decode time stamp (DTS) indicates the time when a picture must be decoded, whereas a presentation time stamp (PTS) indicates when it must be presented to the decoder output.

B-pictures are decoded and presented simultaneously so that they only contain PTS. When an IPBB sequence is received, both I- and P- must be decoded before the first B-picture. A decoder can only decode one picture at a time; therefore the I-picture is decoded first and stored. While the P-picture is being decoded, the decoded I-picture is output so that it can be followed by the B-pictures.



Default.trp Header Interpretation - Prog. 1 Video (PID 50) PES Packet 0

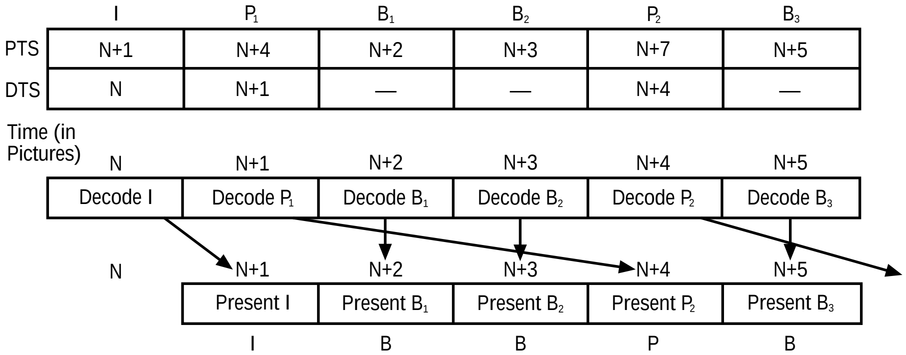
packet start code prefix	stream id	PES packet length
0x000001	0xE0 video_stream	0x07FA

PES scrambling control	PES priority	data alignment indicator	copyright	original or copy	PTS DTS flags	ESCR flags	ES rate flag	DSM trick mode flag	additional copy info flag	PES crc flag	PES extension flag	PES header data length
0x0	0x0	0x0	0x0	0x0	0x3	0x0	0x0	0x0	0x0	0x0	0x0	0xA

PTS	DTS
29763	26163

PES packet data bytes

Figure 6-1.



	I	P_1	B_1	B_2	P_2	B_3
PTS	N+1	N+4	N+2	N+3	N+7	N+5
DTS	N	N+1	—	—	N+4	—

Figure 6-2.

Figure 6-2 shows that when an access unit containing an I-picture is received, it will have both DTS and PTS in the header and these time stamps will be separated by one picture period. If bidirectional coding is being used, a P-picture must follow and this picture also has a DTS and a PTS time stamp, but the separation between the two stamp times is three picture periods to allow for the intervening B-pictures. Thus, if IPBB is received, I is delayed one picture period, P is delayed three picture periods, the two Bs are not delayed at all, and the presentation sequence becomes IBBP. Clearly, if the GOP structure is changed such that there are more B-pictures between I and P, the difference between DTS and PTS in the P-pictures will be greater.

The PTS/DTS flags in the packet header are set to indicate the presence of PTS alone or both PTS and DTS time stamp. Audio packets may contain several access units and the packet header contains a PTS. Because audio packets are never transmitted out of sequence, there is no DTS in an audio packet.

Section 7 – Program Streams

Program streams are one way of combining several PES packet streams and are advantageous for recording applications such as DVD.

7.1 Recording vs. Transmission

For a given picture quality, the data rate of compressed video will vary with picture content. A variable bit rate channel will give the best results. In transmission, most practical channels are fixed and the overall bit rate is kept constant by the use of stuffing (meaningless data).

In a DVD, the use of stuffing is a waste of storage capacity. However, a storage medium can be slowed down or speeded up, either physically or, in the case of a disk drive, by changing the rate of data transfer requests. This approach allows a variable-rate channel to be obtained without capacity penalty. When a medium is replayed, the speed can be adjusted to keep a data buffer approximately half full, irrespective of the actual bit rate, which can change dynamically. If the decoder reads from the buffer at an increased rate, it will tend to empty the buffer, and the drive system will simply increase the access rate to restore balance. This technique only works if the audio and video were encoded from the same clock; otherwise, they will slip over the length of the recording.

To satisfy these conflicting requirements, program streams and transport streams have been devised as alternatives. A program stream works well on a single program with variable bit rate in a recording environment; a transport stream works well on multiple programs in a fixed bit rate transmission environment.

The problem of genlocking to the source does not occur in a DVD player. The player determines the time base of the video with a local synchronizing pulse generator (internal or external) and simply obtains data from the disk in order to supply pictures on that time base. In transmission, the decoder has to recreate the time base at the encoder or it will suffer overflow or underflow. Thus, a transport stream uses program clock reference (PCR), whereas a program stream has no need for the program clock.

7.2 Introduction to Program Streams

A program stream is a PES packet multiplex that carries several elementary streams that were encoded using the same master clock or system time clock (STC). This stream might be a video stream and its associated audio streams, or a multichannel audio-only program. The elementary video stream is divided into access units (AUs), each of which contains compressed data describing one picture. These pictures are identified as I, P, or B and each carries an AU number that indicates the correct display sequence. One video AU becomes one program-stream packet. In video, these packets vary in size. For example, an I-picture packet will be much larger than a B-picture packet. Digital audio access units are generally of the same size and several are assembled into one program-stream packet. These packets should not be confused with transport-stream packets that are smaller and of fixed size. Video and audio AU boundaries rarely coincide on the time axis, but this lack of coincidence is not a problem because each boundary has its own time-stamp structure.

Section 8 – Transport Streams

A transport stream is more than a multiplex of many PES packets. In program streams, time stamps are sufficient to recreate the time axis because the audio and video are locked to a common clock. For transmission down a data network over distance, there is an additional requirement to recreate the clock for each program at the decoder. This requires an additional layer of syntax to provide PCR signals.

8.1 The Job of a Transport Stream

The Transport Stream carries many different programs and each may use a different compression factor and a bit rate that can change dynamically even though the overall bit rate stays constant. This behavior is called statistical multiplexing and it allows a program that is handling difficult material to borrow bandwidth from a program handling easy material. Each video PES can have a different number of audio and data PESs associated with it. Despite this flexibility, a decoder must be able to change from one program to the next and correctly select the appropriate audio and data channels. Some of the programs can be protected so that they can only be viewed by those who have paid a subscription or fee. The transport stream must contain CA information to administer this protection. The transport stream contains PSI to handle these tasks.

The transport layer converts the PES data into small packets of constant size (adding stuffing bits if necessary) that are self-contained. When these packets arrive at the decoder, there may be jitter in the timing. The use of time division multiplexing also causes delay, but this factor is not fixed because the proportion of the bit stream allocated to each program need not be fixed. Time stamps are part of the solution, but they only work if a stable clock is available. The transport stream must contain further data allowing the re-creation of a stable clock.

The operation of digital video production equipment is heavily dependent on the distribution of a stable system clock for synchronization. In video production, genlocking is used, but over long distances, the distribution of a separate clock is not practical. In a transport stream, the different programs may have originated in different places that are not necessarily synchronized. As a result, the transport stream has to provide a separate means of synchronizing for each program.

This additional synchronization method is called a PCR and it recreates a stable reference clock that can be divided down to create a time line at the decoder, so that the time stamps for the elementary streams in each program become useful. Consequently, one definition of a program is a set of elementary streams sharing the same timing reference.

In a single program transport stream (SPTS), there will be one PCR channel that recreates one program clock for both audio and video. The SPTS is often used as the communication between an audio/video coder and a multiplexer.

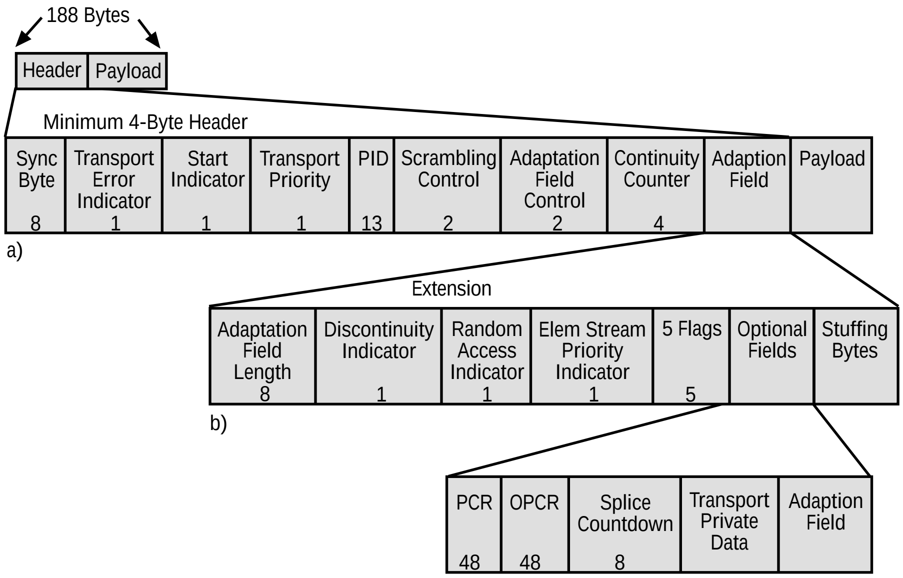


Figure 8-1.

8.2 Packets

Figure 8-1 shows the structure of a transport stream packet. The size is a constant 188 bytes and it is always divided into a header and a payload. Figure 8-1a shows the minimum header of 4 bytes. In this header, the most important information is:

- The sync byte. This byte is recognized by the decoder so that the header and the payload can be deserialized.
- The transport error indicator. This indicator is set if the error correction layer above the transport layer is experiencing a raw-bit error rate (BER) that is too high to be correctable. It indicates that the packet may contain errors. See Section 10 – Introduction to DVB & ATSC for details of the error correction layer.
- The packet identification (PID). This thirteen-bit code is used to distinguish between different types of packets. More will be said about PID later.
- The continuity counter. This four-bit value is incremented by the multiplexer as each new packet having the same PID is sent. It is used to determine if any packets are lost, repeated, or out of sequence.

In some cases, more header information is needed, and if this is the case, the adaptation field control bits are set to indicate that the header is larger than normal. Figure 8-1b shows that when this happens the extra header length is described by the adaptation field length code. Where the header is extended, the payload becomes smaller to maintain constant packet length.

8.3 Program Clock Reference (PCR)

The encoder used for a particular program will have a 27-MHz program clock. In the case of an SDI (serial digital interface) input, the bit clock can be divided by 10 to produce the encoder program clock. Where several programs originate in the same production facility, it is possible that they will all have the same clock. In case of an analog video input, the H-sync period will need to be multiplied by a constant in a phase-locked loop to produce 27 MHz.

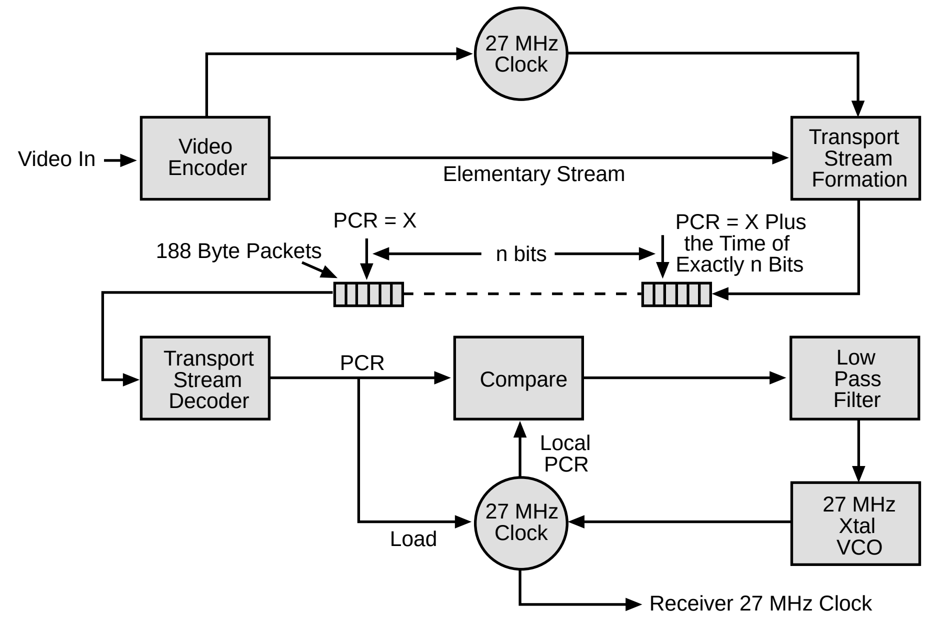


Figure 8-2.

The adaptation field in the packet header is used periodically to include the PCR code that allows generation of a locked clock at the decoder. If the encoder or a remultiplexer has to switch sources, the PCR may have a discontinuity. The continuity count can also be disturbed. This event is handled by the discontinuity indicator, which tells the decoder to expect a disturbance. Otherwise, a discontinuity is an error condition.

Figure 8-2 shows how the PCR is used by the decoder to recreate a remote version of the 27-MHz clock for each program. The encoder clocks drive a constantly running binary counter, and the value of these counters are sampled periodically and placed in the header adaptation fields as the PCR. The PCR is a 42-bit number that is represented by a 33-bit PCR base, plus a 9-bit PCR extension to provide higher resolution. (The PCR base, like the PTS, is a 33-bit number that is a sample of a counter driven by a 90-kHz clock). The packets generated by each encoder are given a different PID. The decoder recognizes the packets with the correct PID for the selected program and ignores others. At the decoder, a voltage controlled oscillator (VCO) generates a nominal 27 MHz clock and this drives a local PCR counter. The local PCR is compared with the PCR from the packet header and the difference is the PCR phase error. This error is filtered to control the VCO that eventually will bring the local PCR count into step with the header PCRs. Heavy VCO filtering ensures that jitter in PCR transmission does not modulate the clock. The discontinuity indicator will reset the local PCR count and, optionally, may be used to reduce the filtering to help the system quickly lock to the new timing.

MPEG requires that PCRs be sent at a rate of at least 10 PCRs per second, whereas DVB specifies a minimum of 25 PCRs per second.

8.4 Packet Identification (PID)

A 13-bit field in the transport packet header contains the Packet Identification code (PID). The PID is used by the demultiplexer to distinguish between packets containing different types of information. The transport-stream bit rate must be constant, even though the sum of the rates of all of the different streams it contains can vary. This requirement is handled by the use of null packets. If the real payload rate falls, more null packets are inserted. Null packets always have the same PID, which is 8191 (thirteen ones in the binary representation).

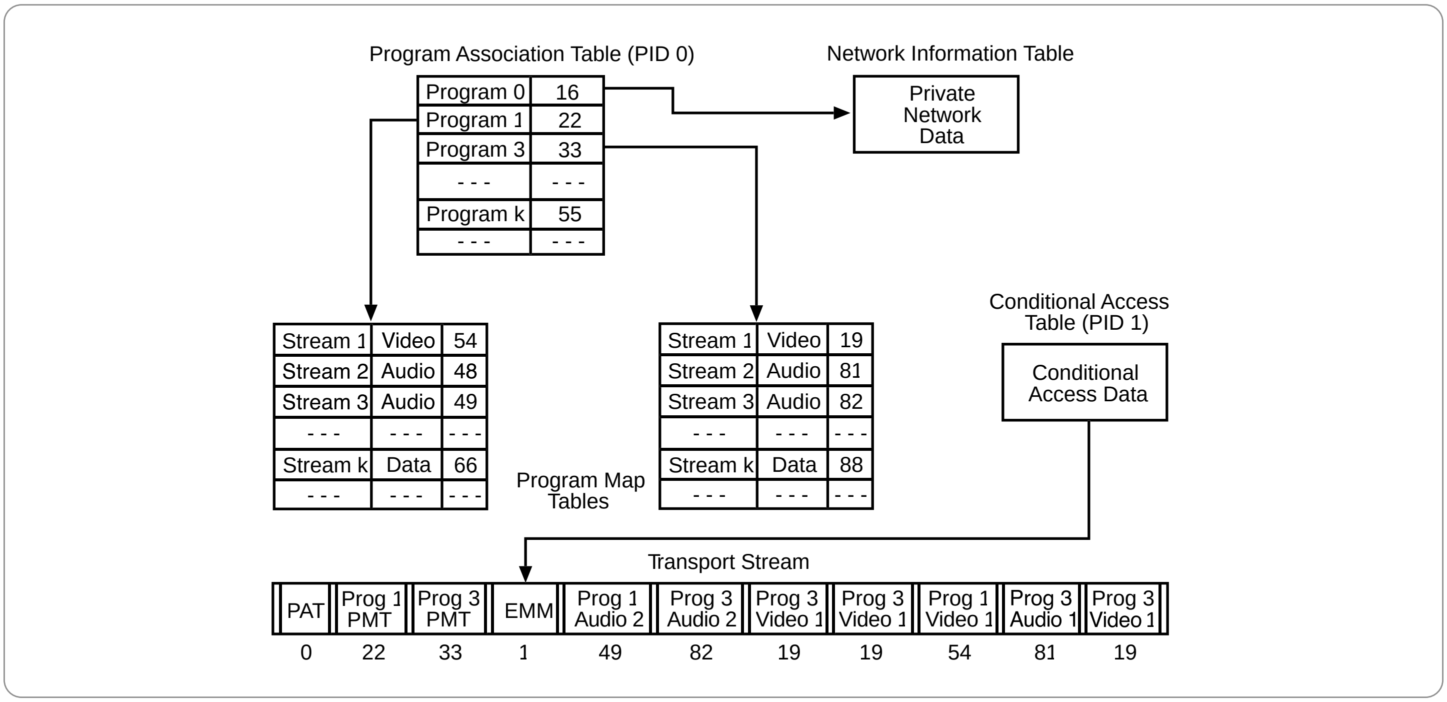


Figure 8-3.

In a given transport stream, all packets belonging to a given elementary stream will have the same PID. The demultiplexer can easily select all data for a given elementary stream simply by accepting only packets with the right PID. Data for an entire program can be selected using the PIDs for video, audio and data streams such as subtitles or teletext. The demultiplexer can correctly select packets only if it can correctly associate them with the elementary stream to which they belong. The demultiplexer can do this task only if it knows what the right PIDs are. This is the function of the PSI.

8.5 Program Specific Information (PSI)

PSI is carried in packets having unique PIDs, some of which are standardized and some of which are specified by the program association table (PAT), conditional access table (CAT) and the transport stream description table (TSDT). These packets must be included periodically in every transport stream. The PAT always has a PID of 0, the CAT always has a PID of 1, and the TSDT always has a PID of 2. These values and the null-packet PID of 8191 are the only PIDs fixed by the MPEG standard. The demultiplexer must determine all of the remaining PIDs by accessing the appropriate tables. However, there are some constraints in the use of PIDs in ATSC and DVB. In this respect (and in some others), MPEG and DVB/ATSC are not fully interchangeable. All DVB and ATSC transport stream must be MPEG-2 compliant (ISO/IEC 13818-1), but not all MPEG-2 transport streams will be compliant with the ATSC (A/65A) or DVB (EN 300 468) standards.

The programs that exist in the transport stream are listed in the program association table (PAT) packets (PID = 0) that carries the PID of each PMT packet. The first entry in the PAT, program 0, is reserved for network data and contains the PID of network information table (NIT) packets. Usage of the NIT is optional in MPEG-2, but is mandatory in DVB.

The PIDs for entitlement control messages (ECM) and entitlement management messages (EMM) are listed in the conditional access table (CAT) packets (PID = 1).

As Figure 8-3 shows, the PIDs of the video, audio, and data elementary streams that belong in the same program are listed in the Program Map Table (PMT) packets. Each PMT packet normally has its own PID, but MPEG-2 does not mandate this. The program number within each PMT will uniquely define each PMT.

A given network information table (NIT) contains details of more than just the transport stream carrying it. Also included are details of other transport streams that may be available to the same decoder, for example, by tuning to a different RF channel or steering a dish to a different satellite. The NIT may list a number of other transport streams and each one must have a descriptor that specifies the radio frequency, orbital position, and so on. In DVB, additional metadata, known as DVB-SI, is included, and the NIT is considered to be part of DVB-SI. This operation is discussed in Section 10 – Introduction to DVB & ATSC. When discussing the subject in general, the term PSI/SI is used.

Upon first receiving a transport stream, the demultiplexer must look for PIDs 0 and 1 in the packet headers. All PID 0 packets contain the PAT. All PID 1 packets contain CAT data.

By reading the PAT, the demultiplexer can find the PIDs of the NIT and of each program map table (PMT). By finding the PMTs, the demultiplexer can find the PIDs of each elementary stream.

Consequently, if the decoding of a particular program is required, reference to the PAT and then the PMT is all that is needed to find the PIDs of all of the elementary streams in the program. If the program is encrypted, access to the CAT will also be necessary. As demultiplexing is impossible without a PAT, the lockup speed is a function of how often the PAT packets are sent. MPEG specifies a maximum interval of 0.5 seconds for the PAT packets and the PMT packets that are referred to in those PAT packets. In DVB and ATSC, the NIT may reside in packets that have a specific PID.

Section 9 – Digital Modulation

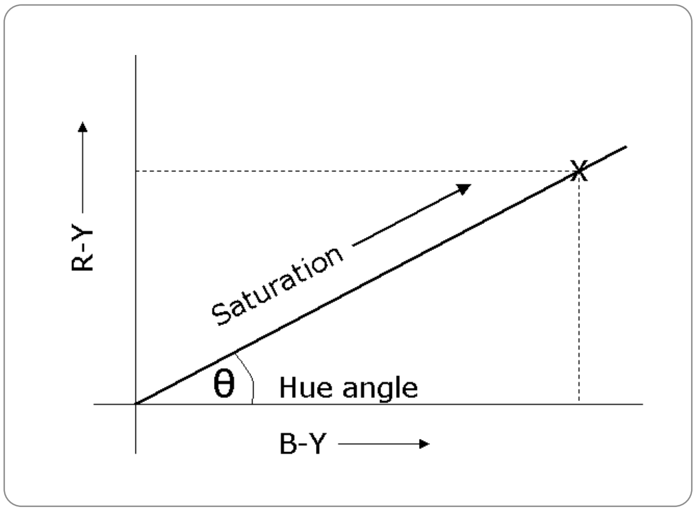


Figure 9-1.

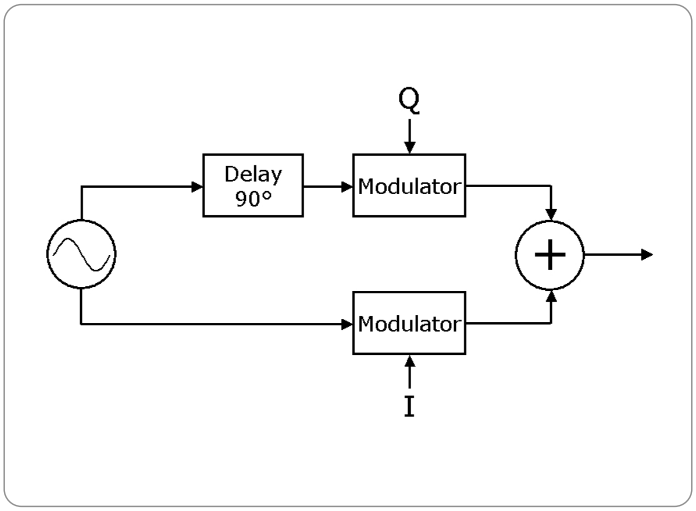


Figure 9-2.

MPEG systems encode and package video, audio, and other data. For storage, the resulting data stream may be recorded on a hard drive or perhaps DVD. For other applications, the MPEG data, usually in the form of a transport stream, has to be sent from one place to another, or to many places. Television systems use cable, satellite and terrestrial transmission in various ways for contribution, distribution and broadcast. All of these transport mechanisms require that the data be modulated on some carrier. This section provides a brief introduction to the digital modulation schemes that are used to achieve this.

This section discusses just the modulation techniques. Other processes are necessary to make a data stream suitable for transmission, and will depend on both the modulation chosen and the channel characteristics. These techniques will be discussed in Section 10.

9.1 Principles of Modulation

A continuous carrier conveys no information unless it is modified in some way by the information to be transmitted. A carrier can be modified in three ways, by changing its amplitude, frequency or phase. Frequency and phase are, of course, closely related. Although generally treated as separate modulation types, the distinction can become very blurred; some "frequency modulation" systems are implemented by "phase modulators."

9.2 Analog Modulation

In the analog world, the amplitude or frequency of a carrier is changed (modulated), according to the amplitude of an audio or video signal, usually according to a linear law. Phase modulation is also used in analog systems, the most obvious example being the modulation of color difference signals onto a color subcarrier in the NTSC and PAL television systems.

This familiar operation provides a useful example. We can think of the color information in two ways. It can be expressed as a phase angle (related to hue) plus an amplitude (related to saturation), or as values of two color difference signals, B-Y and R-Y. If the color difference components are treated as *"x"* and *"y"* values on a graph, the two representations are seen to be closely related, as illustrated in Figure 9.1. This figure is similar to the well-known vectorscope display.

9.3 Quadrature Modulation

Figure 9.2 shows how we can modulate a carrier with two different signals, using the technique known as "quadrature modulation." A single carrier is split into two paths, and one path is delayed by a time equal to one-quarter of the cycle time of the carrier. This generates a carrier of identical frequency, but phase shifted from the original by 90 degrees. The two carriers are each amplitude modulated by an appropriate signal, and the two modulated carriers are then added together. This generates a single signal with amplitude and phase determined by the amplitudes of the two modulating signals.

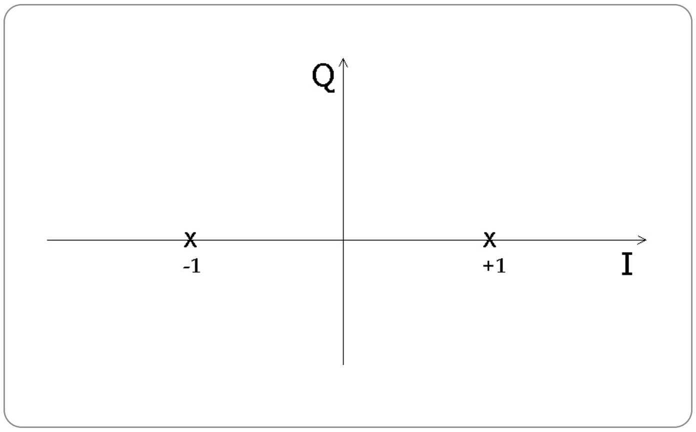


Figure 9-3.

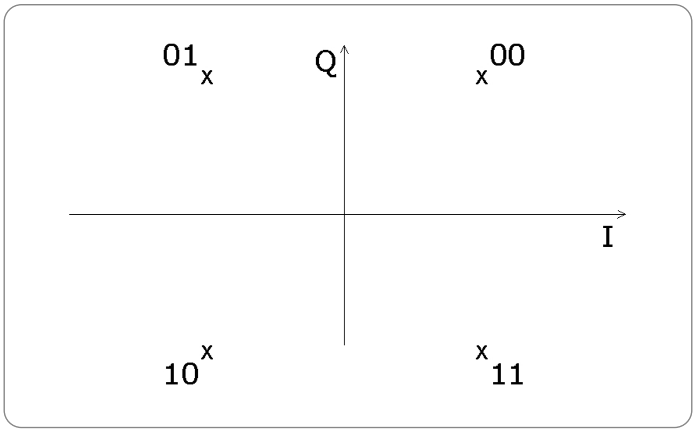


Figure 9-4.

Demodulation is achieved by an almost identical process; the received signal is sent to two demodulators. In each demodulator the signal is multiplied by a local oscillator signal, synchronized to the received signal. The two local oscillator signals are 90 degrees apart, and each demodulator recovers one axis of the quadrature modulation.

9.4 Simple Digital Modulation Systems

Most digital modulation systems use some form of quadrature modulation; the two axes are usually designated I and Q. Sometimes only one modulation axis is used.

All digital modulation schemes represent a compromise of bandwidth efficiency, robustness and complexity. *Symbol rate* is the principal factor in determining the bandwidth of the transmitted signal. The symbol rate is the rate at which the modulation is changed, so it is the same as the bandwidth of the modulating signal(s).

Some simple digital modulation systems carry only one bit of information per symbol. In other words, each symbol may represent one of two possible states, representing a binary zero or a binary one. In this case the *bit rate* of the system is the same as the symbol rate. However, other systems have many possible states for each symbol, so they can convey more than one bit of information per symbol. Generally the number of states is made to be a power of two, so the bit rate of the system is some integer multiple of the symbol rate.

Digital modulation systems are frequently labeled by the modulation type, preceded by a number representing the number of states for each symbol. For example, 4QAM describes quadrature amplitude modulation with four possible states for each symbol. Four states can convey two bits of information (00, 01, 10, 11), so the bit rate of a 4QAM system is twice the symbol rate.

The simplest digital modulation systems convey one bit of information per symbol. Each symbol has two possible states, representing binary zero and binary one. The states may be created by amplitude, frequency or phase modulation, but frequency and phase modulation are the most common.

Binary frequency shift keying (BFSK or 2FSK) uses one carrier frequency to represent a binary zero, and a different frequency to represent a binary one. Sometimes the frequency difference is very small, and is achieved by a phase modulator.

Binary phase shift keying (BPSK or 2PSK) uses one phase of the (constant amplitude) carrier to represent binary zero, and the inverse (180 degrees phase shift) to represent binary one. The different possible states of a symbol are usually shown in a *constellation diagram* showing the various combinations resulting from permitted values of the I and Q modulating signals. The constellation diagram for BPSK, shown in Figure 9.3, is very simple; only one axis is used, and there are only two permitted values.

These systems can be very robust; the receiver needs only enough signal (or signal-to-noise ratio) to determine which of two possible states has been transmitted for each symbol. However, they do not use spectrum efficiently; the bandwidth is nominally the same as the bit rate. These systems are used on very difficult transmission paths, such as deep-space telemetry.

9.5 Phase Shift Keying

BPSK or 2PSK was described in the previous section. Other forms of PSK modulation use both the I and Q axes. Quaternary phase shift keying (QPSK, also known as quadrature phase shift keying) is the most common, and uses two values on each axis. The constellation diagram is shown in Figure 9.4. QPSK has four possible states per symbol, so each symbol carries two bits of information; one possible mapping of states to binary values is shown in the figure. QPSK is used extensively in satellite communications.

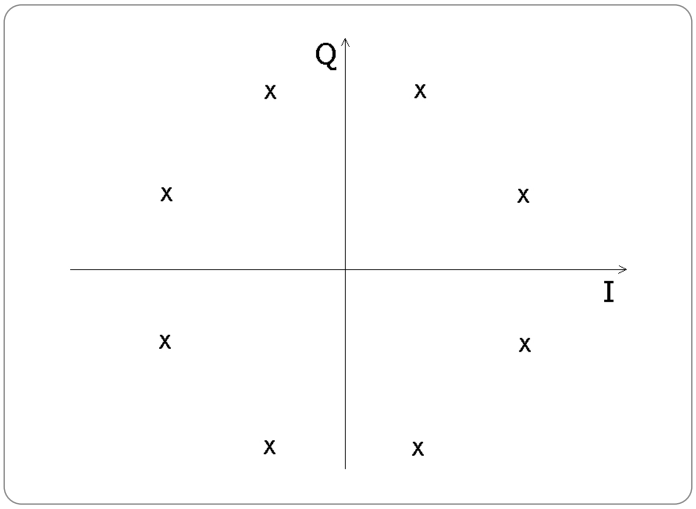


Figure 9-5.

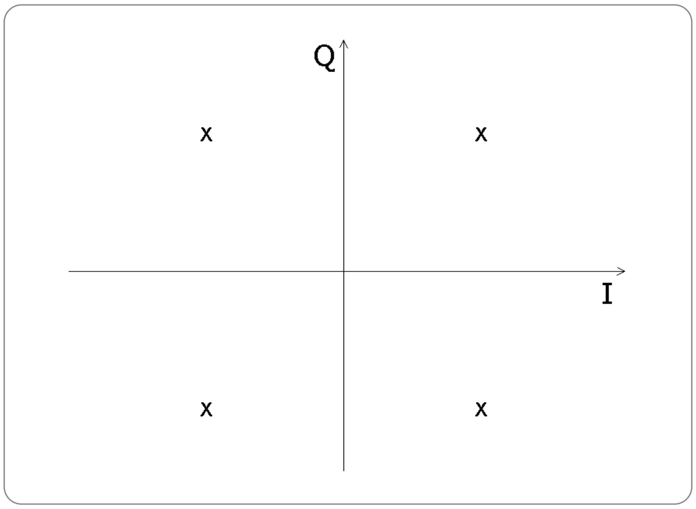


Figure 9-6.

8PSK is less common, but is also used in satellite systems, particularly in Japan. The constellation diagram is shown in Figure 9-5. 8PSK carries three bits of information in each symbol, so the bit rate is three times the symbol rate.

9.6 Quadrature Amplitude Modulation - QAM

Quadrature amplitude modulation (QAM) is the basis of many transmission systems. Both the I and Q axes are used for modulation and, depending on the particular variant, two or more amplitude levels are permitted for each axis.

The simplest variant is 4QAM where just two values are used for each axis, providing four possible states for each symbol. The constellation diagram is shown in Figure 9-6, where it will be seen that 4QAM is identical to QPSK, and carries two bits per symbol.

16QAM uses four values on each axis, providing 16 possible states. 16QAM systems carry four bits per symbol. If six values are permitted for each modulation axis, there are a total of 36 possible states. Five bits may be carried using only 32 states, so four of the possible combinations are not used in 32QAM. The constellation diagram for 16QAM is shown in Figure 9-7 and for 32QAM is shown in Figure 9-8. In 32QAM the four “corner” states are not used; these are the states that would represent the highest amplitude and, therefore, the highest transmitter power.

Figures 9-7 and 9-8 also help to show the trade off between bit rate and robustness. In the presence of noise or jitter, the closer spacing of the states in 32QAM (at equal transmitted power) will make decoding errors more likely. Put another way, the more possible states per symbol the better the signal-to-noise ratio required for a given error rate.

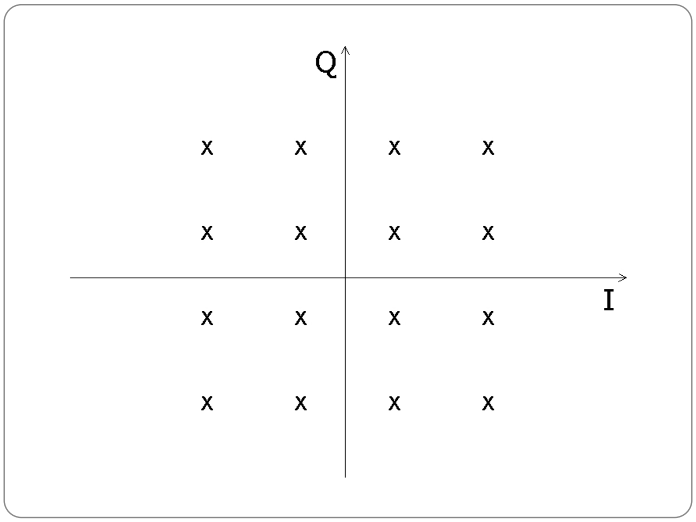


Figure 9-7.

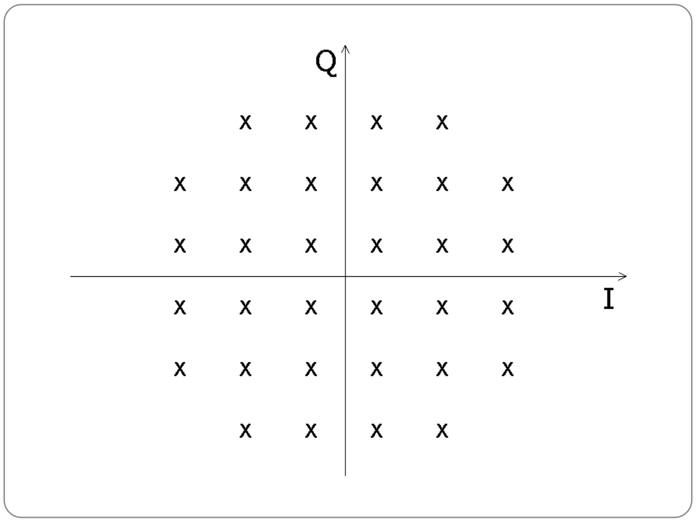


Figure 9-8.

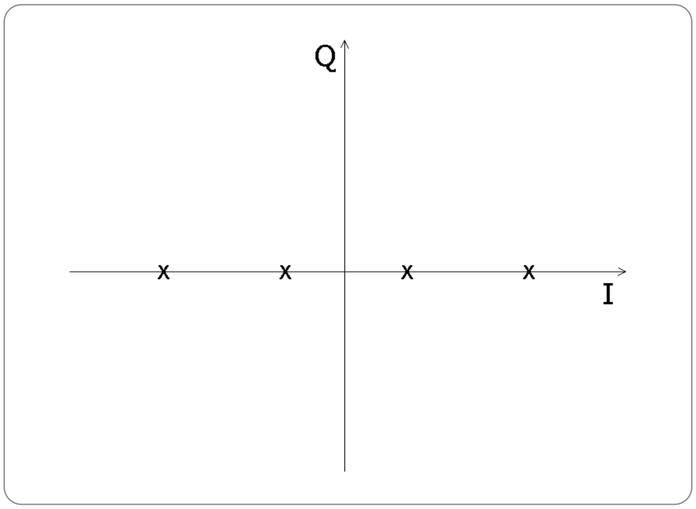


Figure 9-9.

When good signal-to-noise can be guaranteed, even greater constellation densities may be used. 64QAM uses eight values on each axis, and carries six bits per symbol. 64 QAM is the most extensively used modulation scheme in cable systems worldwide, as it provides a good trade-off between robustness and compatibility with legacy cable infrastructures. 256QAM, used in some of the latest cable television systems, has 16 permissible values for each modulation axis, and carries eight bits per symbol.

9.7 Vestigial Sideband Modulation – VSB

When a carrier is modulated, sidebands are generated above and below the carrier frequency. For example, a QAM system with a symbol rate of 3 megasymbols per second will have upper and lower sidebands each about 3 MHz wide, requiring a nominal channel bandwidth of 6 MHz.

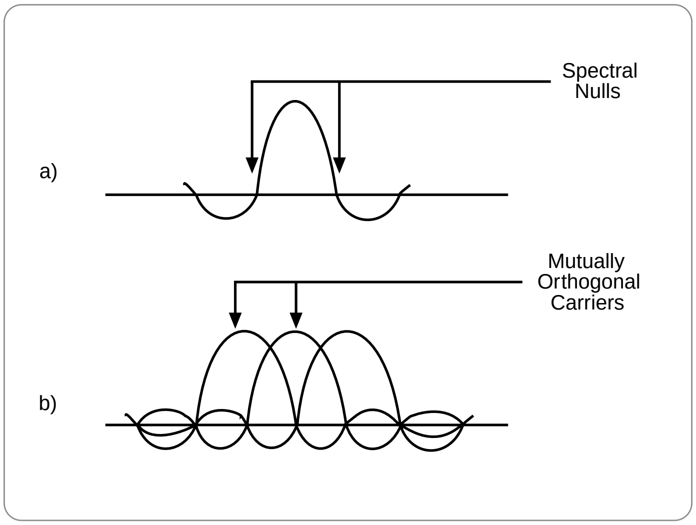


Figure 9-11.

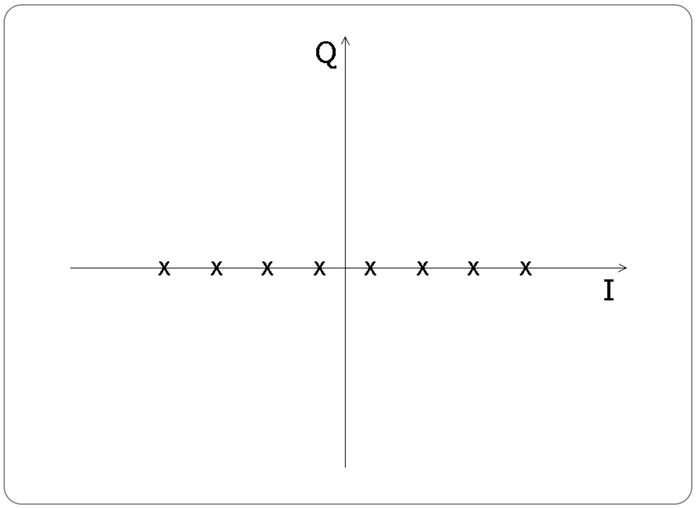


Figure 9-10.

To recover both amplitude and phase information (or both axes of the quadrature modulation), both sidebands must be recovered at the receiver.

Vestigial sideband systems eliminate most of one sideband prior to transmission, so only one modulation axis can be recovered. (An alternative interpretation is to say that the other modulation axis is used so as to suppress the unwanted sideband.) 2VSB has the same constellation as BPSK. Figure 9-9 shows the constellation diagrams for 4VSB and Figure 9-10 shows 8VSB, carrying respectively two and three bits per symbol.

8VSB modulation is used in the United States by the ATSC digital television standard. 4VSB was proposed originally, providing two bits per symbol. However, it was found that 8VSB, in conjunction with Trellis coding (see Section 10.6), and convolutional inner error correction provides the same date rate with improved signal-to-noise performance.

9.8 Coded Orthogonal Frequency Division Multiplex – COFDM

In the above systems, a baseband signal is supplied to modulators that operate on a single carrier to produce the transmitted sideband(s). An alternative to a wideband system is one that produces many narrowband carriers at carefully regulated spacing. Figure 9-11a shows that a digitally modulated carrier has a spectral null at each side. Another identical carrier can be placed here without interference because the two are mutually orthogonal as Figure 9-11b shows. This is the principle of OFDM (orthogonal frequency division multiplexing). In practice, a variant known as coded orthogonal frequency division multiplexing (COFDM) improves the performance dramatically in non-ideal channel conditions by the use of Viterbi convolutional coding, described in the next section. COFDM is used in the digital video broadcasting (DVB) terrestrial system, DVB-T.

Each carrier in an OFDM system may be modulated by any of the techniques described in this section. In practice QAM is generally used, 16QAM and 64QAM being most common. It is not necessary to use all of the carriers. For example, if one part of the channel is known to be subjected to a high degree of interference, the affected carriers may be omitted.

The number of carriers in an OFDM system can be very large. DVB-T has options for 1705 or 6817 carriers (known as 2k and 8k systems). Because the bandwidth allotted to each carrier is small, the symbol rate is correspondingly reduced, and the length of time to transmit each symbol is increased. This is the key to OFDM's tolerance to multi-path interference.

In a single-carrier system, such as 8VSB, a high data rate means that the symbol time is very short. In the case of the ATSC digital television system, some 11 million symbols are transmitted each second, giving a symbol duration of less than 100 ns. This means that even a very short multi-path delay will create inter-symbol interference because the delayed signal representing one symbol will arrive during reception of a subsequent symbol.

In contrast, an OFDM system with thousands of carriers will have a symbol time in the order of hundreds of microseconds (depending on the data rate, the number of carriers, and the modulation used). Inter-symbol interference may be virtually eliminated by adding a "guard band" to each symbol – deliberately making the symbol longer than needed. This reduces the symbol rate, but only to a relatively small degree. For example, if the nominal symbol duration is 200 µs, a guard band of 50 µs will reduce the symbol rate by only 20% – and the elimination of inter-symbol interference may allow a higher order constellation to be used, perhaps more than compensating for this loss. This technique is not practical on a wide-band single-carrier system. As an example, the same 50 µs guard band with a 100 ns symbol time would reduce the data rate to a fraction of one percent!

This tolerance to multi-path interference also makes COFDM systems well suited to *single frequency networks* where two or more synchronized transmitters emit the same signal. A receiver may, depending on its location and antenna system, receive signals from more than one transmitter at different times. If the path lengths are radically different, the main signal will probably be very much stronger than the secondary signal, and interference will be minimal. If path lengths and signal strengths are similar, the guard band will prevent inter-symbol interference.

COFDM systems are very flexible, and may be "tuned" to suit a wide variety of transmission requirements but, as always, increased robustness is at the expense of data rate. There are many arguments about the relative merits of single-carrier and multi-carrier systems, but it is generally believed that under simple channel conditions COFDM requires somewhat more power than VSB for the same coverage at the same data rate. COFDM also has a higher peak-to-average ratio at the transmitter that may result in more interference to other services.

However, many believe that in complex multi-path situations such as inner city "urban canyons," COFDM can provide more reliable reception.

9.9 Integrated Services Data Broadcasting (ISDB)

Integrated services data broadcasting (ISDB) is a development that uses many modulation schemes and has been developed for digital television services in Japan. It is designed to support hierarchical systems of many levels. It could be used, for example, to provide simultaneously low data rate reception under exceptionally difficult mobile conditions, intermediate data rate (standard definition) for fringe-area static reception, and high data rate (perhaps for HDTV) for good reception conditions.

There are three ISDB modulation systems currently in use in Japan:

9.9.1 ISDB-S Satellite System

Launched in December 2000, ISDB-S enabled two broadcasters to share a satellite transponder. It is also referred to as BS-digital or CS-digital when the space segment is a broadcast satellite or a communication satellite, respectively.

Up to eight transport streams can be used in all in a manner to be agreed between the broadcasters sharing the transponder. The aggregate bit rate will depend on the transponder bandwidth and the modulation mode used. For example, for a transponder of 34.5 MHz, the maximum rate including forward error correction is 56.610 Mbits/s.

Hierarchical modulation allows the modulation mode to be varied on a packet-by-packet basis within a 48-packet frame. Each packet is assigned a modulation slot. Four modulation modes are supported BSPK(1/2), QPSK (to 7/8) and TC8PSK.The number of slots vary according to the mode used.

9.9.2 ISDB-C Cable System

The main feature of this system is that it transmits multiple transport streams on a single 64 QAM carrier. The system was developed in order to be able to retransmit efficiently the information carried on ISDB-S signals. A maximum of 52.17 Mbits/s of information are transmitted typically on a BS-digital carrier. The information rate of a 64 QAM/6 MHz signal is 29.162 Mbits/s. Hence, at least two cable television channels must be used to retransmit information of a single BS carrier. The full BS digital service consists of four broadcasters and occupies approximately 174 MHz including guard bands. Using ISDB-C 8 cable channels would be required to carry this information whereas as many as 29 channels would be required using conventional cable transmission of one transport stream per carrier.

There are 52 modulation slots plus 1 slot for the synchronizing header, TSMF (transport stream multiplexing frame).

9.9.3 ISDB-T Terrestrial Modulation

The ISDB-T channel is divided into 13 segments (typically 400-500 kHz wide), and a separate COFDM transmission is used for each segment. All of the parameters affecting robustness (number of carriers, guard band length, modulation type, convolution coding) may be chosen separately for each layer of the hierarchy. For example, the most robust segment might use a long guard band, QPSK modulation, and 1/2 convolution coding. The highest level could use a shorter guard band, 64QAM, and 7/8 convolution coding – providing many times the data rate of the robust segment.

The center segment may be used for *partial reception*, designed to allow a narrow band receiver to receive this segment only.

In normal OFDM the complete channel bandwidth represents a single layer. The carriers used are spaced out across the bandwidth at set multiples of a certain frequency. In ISDB-T the channel bandwidth of 5.6 MHz is divided up into 13 segments each having a bandwidth of 429 kHz. Hierarchical transmission of ISDB-T is achieved by transmitting OFDM segment groups having different transmission parameters. These groups of layers constitute the layers. In non-hierarchical modulation the same modulation scheme is used for all 13 segments.

9.9.4 ISDB in Summary

ISDB-S provides a means of applying different modulation modes to multiple transport streams and transmitting them in a 34.5 MHz channel on a single carrier.

ISDB-C provides a means of transmitting multiple transport streams in a single 6 MHz channel on a single carrier but with a common modulation mode.

ISDB-T is concerned with up to three transmissions of the same transport stream in a single 6 MHz channel.

Section 10 – Introduction to DVB & ATSC

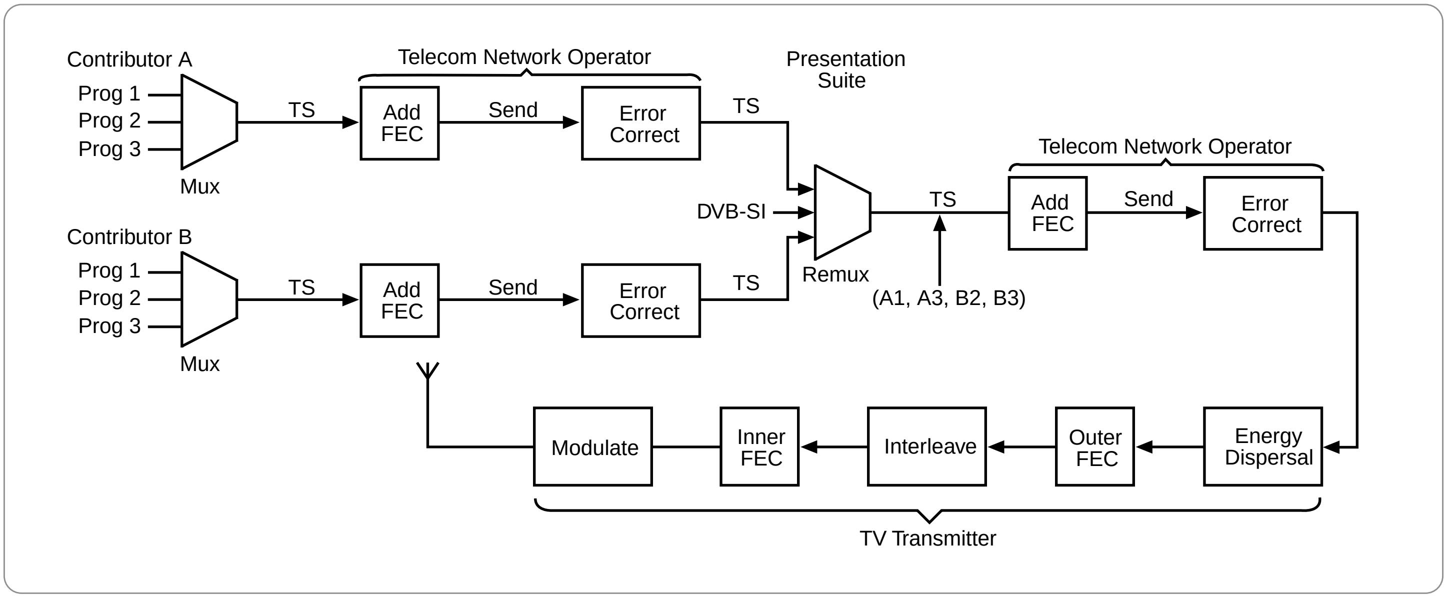


Figure 10-1.

MPEG compression is already being used in broadcasting and will become increasingly important in the future. This section discusses the additional requirements for digital television broadcasting, as implemented by the two principal DTV Standards.

10.1 An Overall View

ATSC (Advanced Television Systems Committee) is a U.S. organization that defines standards for terrestrial digital broadcasting. DVB refers to the Digital Video Broadcasting Project and to the standards and practices established by the DVB Project. This project was originally a European project, but produces standards and guides accepted in many areas of the world. These standards and guides encompass all transmission media, including satellite, cable and terrestrial broadcasting.

Digital broadcasting has different distribution and transmission requirements, as is shown in Figure 10.1. Broadcasters will produce transport streams that contain several television programs. Transport streams have no protection against errors, and in compressed data, the effect of errors is serious. Transport streams need to be delivered error-free to transmitters, satellite uplinks and cable head ends. In this context, error free means a bit error rate (BER) of 1 in 10^{-11} or better. This task is normally entrusted to telecommunications network operators, who will use an additional layer of error correction as needed (error correction strategies are selected depending on the transmission channel). This layer should be transparent to the destination.

A particular transmitter or cable operator may not want all of the programs in a transport stream. Several transport streams may be received and a selection of channels may be made and encoded into a single output transport stream using a remultiplexer. The configuration may change dynamically.

Broadcasting in the digital domain consists of conveying the entire transport stream to the viewer. Whether the channel is cable, satellite or terrestrial, the problems are much the same. Metadata describing the transmission must be encoded into the transport stream in a standardized way. In DVB, this metadata is called service information (DVB-SI) and includes services such as teletext as well as details of programs carried both within itself and within other multiplexes.

In broadcasting, there is much less control of the signal quality and noise or interference is a possibility. This requires some form of forward error correction (FEC) layer. Unlike the FEC used by the telecommunications network operators, which can be proprietary, (or standardized as per European Telecommunications Standard Institute (ETSI), which defines DVB transmission over SDH and PDH networks), the forward error connection (FEC) used in broadcasting must be standardized so that receivers will be able to handle it.

The addition of error correction obviously increases the bit rate as far as the transmitter or cable is concerned. Unfortunately, reliable, economical radio and cable-transmission of data requires more than serializing the data. Practical systems require channel coding.

10.2 Remultiplexing

This is a complex task because a remultiplexer has to output a compliant bit stream that is assembled from parts of others. The required data from a given input transport stream can be selected with reference to the program association table and the program map tables that will disclose the PIDs of the programs required. It is possible that the same PIDs have been used in two input transport streams; therefore, the PIDs of one or more elementary streams may have to be changed. The packet headers must pass on the program clock reference (PCR) that will allow the final decoder to recreate a 27 MHz clock. As the position of packets containing PCR may be different in the new multiplex, the remultiplexer may need to edit the PCR values to reflect their new position on the time axis.

The program map tables and program association tables will need to be edited to reflect the new transport stream structure, as will the conditional access tables (CAT).

If the sum of the selected program stream bit rates is less than the output bit rate, the remultiplexer will create stuffing packets with suitable PIDs. However, if the transport streams have come from statistical multiplexers, it is possible that the instantaneous bit rate of the new transport stream will exceed the channel capacity. This condition might occur if several selected programs in different transport streams simultaneously contain high entropy. In this case, the only solution is to recompress and create new, shorter coefficients in one or more bit streams to reduce the bit rate.

10.3 Service Information (SI)

In the future, digital delivery will mean that there will be a large number of programs, teletext and services available to the viewer and these may be spread across a number of different transport streams. Both the viewer and the integrated receiver decoder (IRD) will need help to display what is available and to output the selected service. This capability requires metadata beyond the capabilities of MPEG-PSI (program specific information) and is referred to as DVB-SI (service information). DVB-SI is considered to include the NIT, which is optional in MPEG transport streams.

DVB-SI is embedded in the transport stream as additional transport packets with unique PIDs and carries technical information for IRDs. DVB-SI also contains electronic program guide (EPG) information, such as the nature of a program, the timing and the channel on which it can be located, and the countries in which it is available. Programs can also be rated so that parental judgment can be exercised.

DVB-SI must include the following tables over and above MPEG-PSI:

- Network Information Table (NIT). Information in one transport stream that describes many transport streams. The NIT conveys information relating to the physical organization of the multiplex, transport streams carried via a given network and the characteristics of the network itself. Transport streams are identified by the combination of an original network ID and a Transport Stream ID in the NIT.
- Service Description Table (SDT). Each service in a DVB transport stream can have a service descriptor and these descriptors are assembled into the service description table. A service may be television, radio or teletext. The service descriptor includes the name of the service provider.
- Event Information Table (EIT). EIT is a table for DVB that contains program names, start times, durations and so on.
- Time and Date Table (TDT). The TDT is a table that embeds a UTC time and date stamp in the transport stream.

DVB-SI also defines other optional tables including: bouquet association table (BAT), running status table (RST), time offset table (TOT) and the stuffing table (ST).

The ATSC, like DVB, used the MPEG-2 private section table to define several new tables. This set of new mandatory tables defined by ATSC in A/65A is part of the program and system information protocol (PSIP). ATSC PSIP must include the following tables over and above the MPEG-PSI:

- Terrestrial Virtual Channel Table (TVCT) defining, at a minimum, MPEG-2 programs embedded in the transport stream in which the TVCT is carried.
- Master Guide Table (MGT) defining the type, packet identifiers and versions for all the other PSIP tables in the transport stream, except for the system time table (STT).
- Rating Region Table (RRT) defining the TV parental guideline system referenced by any content advisory descriptor carried within the transport stream.
- System Time Table (STT) defining the current date and time of day.
- Event Information Table (EIT-n) defining the first four Event Information Tables (EIT-0, EIT-1, EIT-2 and EIT-3) describing 12 hours of events (TV programs), each with a coverage of 3 hours, and including all of the virtual channels listed in the TVCT.

10.4 Error Correction

Error correction is necessary because conditions on long transmission paths cannot be controlled. In some systems, error detection is sufficient because it can be used to request a retransmission. Clearly, this approach will not work with real-time signals such as television. Instead, FEC is used in which sufficient extra bits, known as redundancy, are added to the data to allow the decoder to perform corrections in real time.

The FEC used in modern systems is usually based on the Reed-Solomon (R-S) codes. A full discussion of these is outside the scope of this book. Briefly, R-S codes add redundancy to the data to make a code word such that when each symbol is used as a term in a minimum of two simultaneous equations, the sum (or syndrome) is always zero if there is no error. This zero condition is obtained irrespective of the data and makes checking easy. In transport streams, the packets are always 188 bytes long prior to the addition of error-correction data. The addition of 16 bytes of R-S redundancy produces a packet length of 204 bytes. (In practice, transport streams may use 204-byte packets even when FEC is not present. The use of 16 stuffing bytes avoids reclocking the stream when FEC is added or deleted.)

In the event that the syndrome is non-zero, solving the simultaneous equations will result in two values needed for error correction: the location of the error and the nature of the error. However, if the size of the error exceeds half the amount of redundancy added, the error cannot be corrected. Unfortunately, in typical transmission channels, the signal quality is statistical. This means that while single bits may be in error due to noise, on occasion a large number of bits, known as a burst, can be corrupted together. This corruption might be due to lightning or interference from electrical equipment.

It is not economic to protect every code word against such bursts because they do not occur often enough. The solution is to use a technique known as interleaving. Figure 10.2 shows that, when interleaving is used, the source data are FEC coded, but prior to transmission, they are fed into a RAM buffer. Figures 10-3 shows one possible technique in which data enters the RAM in rows and is then read out in columns. The reordered data are now transmitted. On reception, the data are put back to their original order, or de-interleaved, by using a second RAM. The result of the interleaving process is that a burst of errors in the channel after de-interleaving becomes a large number of single-symbol errors, which are more readily correctable.

When a burst error reaches the maximum correctable size, the system is vulnerable to random bit errors that make code words uncorrectable. The use of an inner code applied after interleave and corrected before de-interleave can prevent random errors from entering the de-interleave memory.

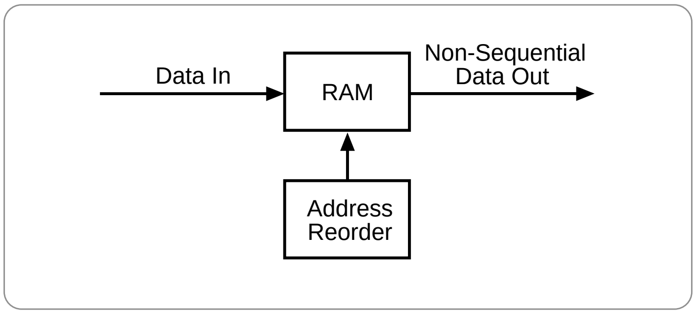


Figure 10-2.

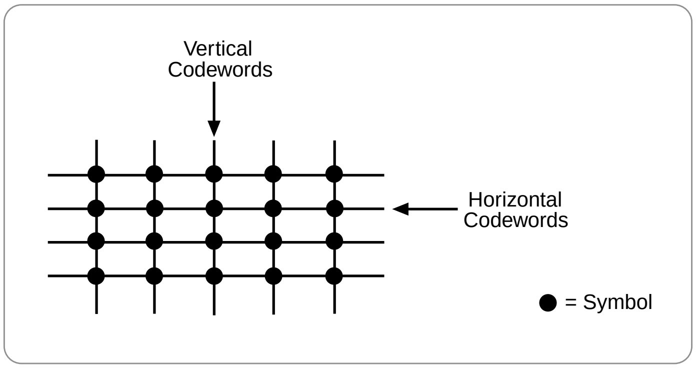


Figure 10-3.

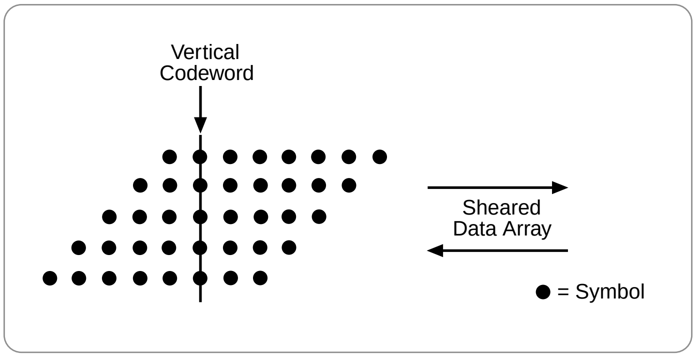


Figure 10-4.

As Figure 10-3 shows, when this approach is used with a block interleave structure, the result is a product code. Figure 10-4 shows that interleave can also be convolutional, in which the data array is sheared by applying a different delay to each row. Convolutional, or cross interleave, has the advantage that less memory is needed to interleave and de-interleave.

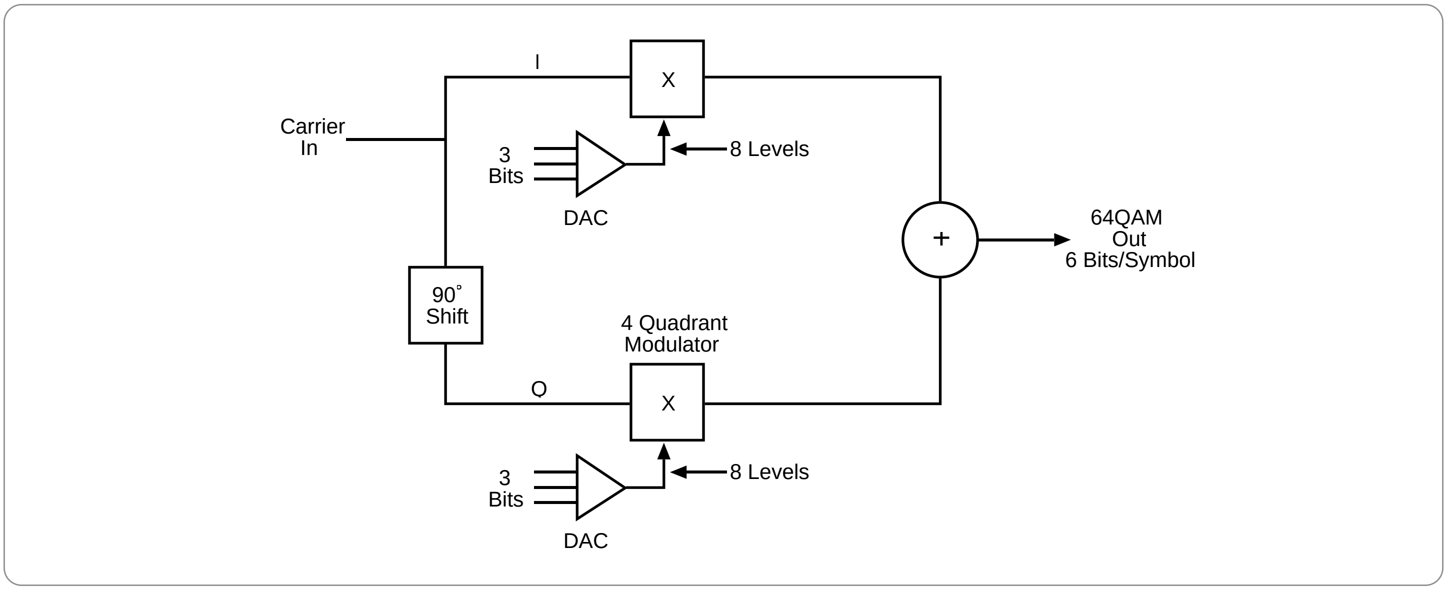


Figure 10-5.

10.5 Channel Coding

Raw serial binary data is unsuitable for transmission for several reasons. Runs of identical bits cause DC offsets and lack a bit clock. There is no control of the spectrum and the bandwidth required is too great. In practical radio and cable systems, a modulation scheme called a channel code is necessary. Digital modulations schemes are discussed in Section 9. Figure 10-5 shows the application of these principles to a 64QAM modulator.

In the schemes described above, the transmitted signal spectrum is signal dependent. Some parts of the spectrum may contain high energy and cause interference to other services, whereas other parts of the spectrum may contain little energy and be susceptible to interference. In practice, randomizing is necessary to decorrelate the transmitted spectrum from the data content. Figure 10-6 shows that when randomizing or energy dispersal is used, a pseudo-random sequence is added to the serial data before it is input to the modulator. The result is that the transmitted spectrum is noise-like with relatively stationary statistics. Clearly, an identical and synchronous sequence must be subtracted at the receiver as shown. Randomizing cannot be applied to sync patterns, or they could not be detected.

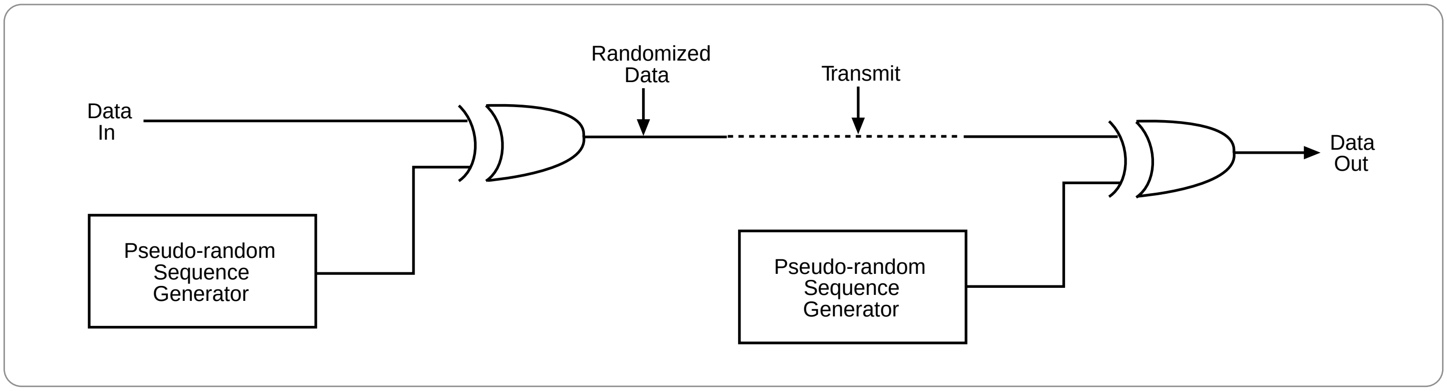

Data In
Randomized Data
Transmit
Data Out
Pseudo-random Sequence Generator
Pseudo-random Sequence Generator

Figure 10-6.

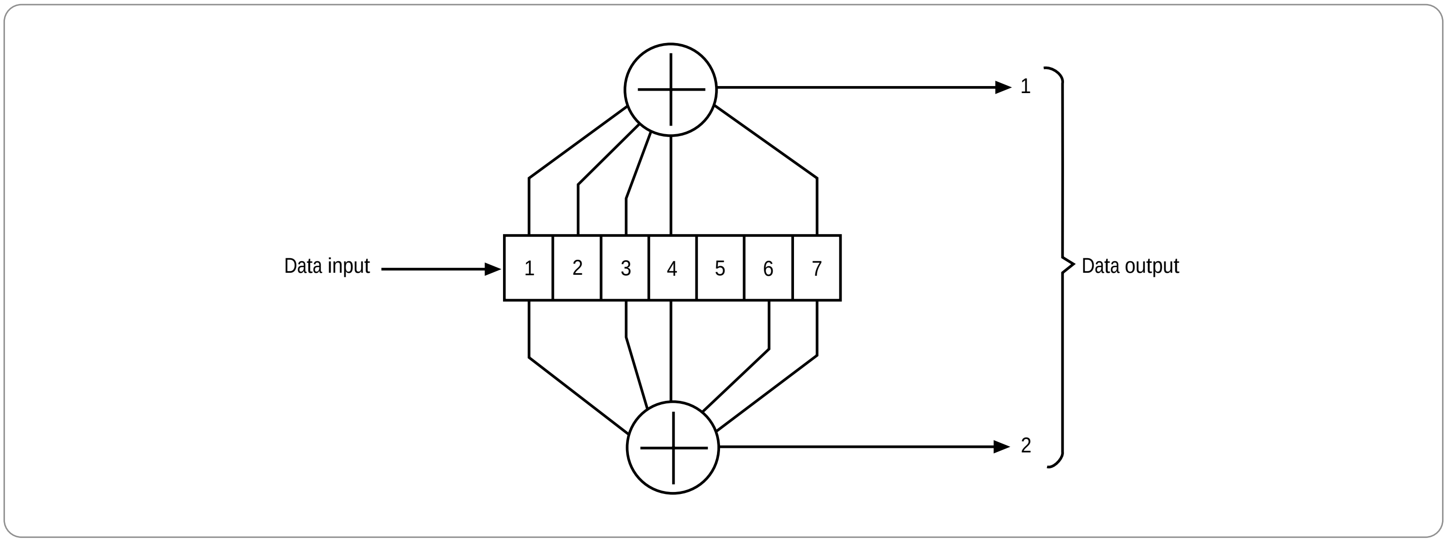


Figure 10-7.

10.6 Inner Coding

The inner code of a FEC system is designed to prevent random errors from reducing the power of the interleave scheme. A suitable inner code can prevent such errors by giving an apparent increase to the SNR of the transmission. In trellis coding, which can be used with multi-level signaling, several multi-level symbols are associated into a group. The waveform that results from a particular group of symbols is called a trellis. If each symbol can have eight levels, then in three symbols there can be 512 possible trellises.

In trellis coding, the data are coded such that only certain trellis waveforms represent valid data. If only 64 of the trellises represent error-free data, then two data bits per symbol can be sent instead of three. The remaining bit is a form of redundancy because trellises other than the correct 64 must be due to errors. If a trellis is received in which the level of one of the symbols is ambiguous due to noise, the ambiguity can be resolved because the correct level must be the one which gives a valid trellis. This technique is known as maximum-likelihood decoding.

The 64 valid trellises should be made as different as possible to make the system continue to work with a poorer signal-to-noise ratio. If the trellis coder makes an error, the outer code will correct it.

In DVB, Viterbi convolutional coding may be used. Figure 10-7 shows that following interleave, the data are fed to a shift register. The contents of the shift register produce two outputs that represent different parity checks on the input data so that bit errors can be corrected. Clearly, there will be two output bits for every input bit; therefore the coder shown is described as a 1/2 rate coder. Any rate between 1/1 and 1/2 would still allow the original data to be transmitted, but the amount of redundancy would vary. Failing to transmit the entire 1/2 output is called puncturing and it allows any required balance to be obtained between bit rate and correcting power.

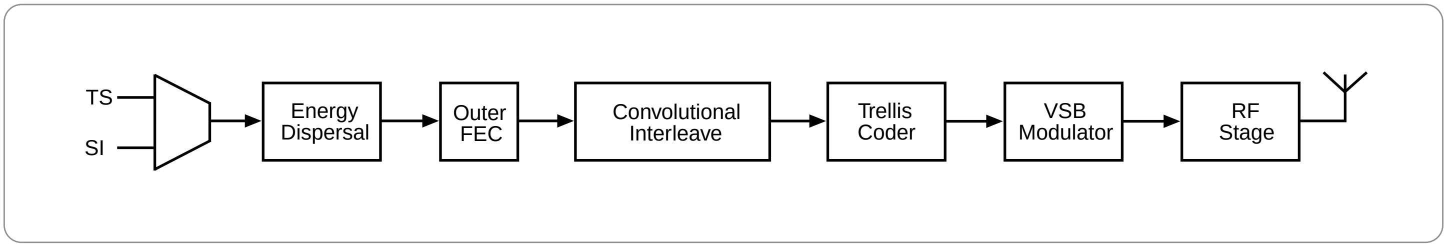


Figure 10-8.

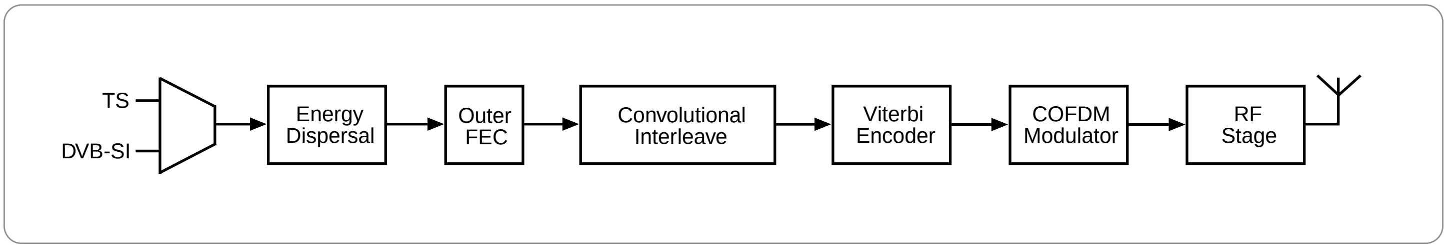


Figure 10-9.

10.7 Transmitting Digits

Figure 10-8 shows the elements of an ATSC digital transmitter. Service information describing the transmission is added to the Transport Stream. This stream is then randomized prior to routing to an outer R-S error correction coder that adds redundancy to the data. A convolutional interleave process then reorders the data so that adjacent data in the Transport Stream are not adjacent in the transmission. An inner trellis coder is then used to produce a multi-level signal for the vestigial sideband (VSB) modulator.

Figure 10-9 shows a DVB-T transmitter. Service information is added as before, followed by the randomizing stage for energy dispersal. Outer R-S check symbols are added prior to interleaving. After the interleaver, the inner coding process takes place, and the coded data is fed to a COFDM modulator. The modulator output is then upconverted to produce the RF output.

At the receiver, the bit clock is extracted and used to control the timing of the whole system. The channel coding is reversed to obtain the raw data plus the transmission errors. The inner code corrects random errors and may identify larger errors to help the outer coder after deinterleaving. The randomizing is removed and the result is the original transport stream. The receiver must identify the PAT, the service information (SI) and PMT that the PAT points to so the viewer can be told what is available in the transport stream and the selected program can be located in the multiplex.

Section 11 – Data Broadcast

The previous sections have looked at the basics of an MPEG-2 transport stream and their main application of carrying compressed video and audio streams, similar to conventional analog broadcasts. However one of the major advantages of an MPEG-2 transport stream is that it can carry data as well as video and audio. Although analog television systems can and do carry data, their maximum data bandwidth is severely limited compared with the bandwidths possible on a transport stream.

This section will provide an overview of the different methods provided by MPEG-2 and the regional variants (DVB, ATSC and ARIB (Association of Radio Industries and Businesses)) to encapsulate data within a transport stream. The next section will then deal with how the data is presented to the viewer by the set top box (which does not strictly fall within the scope of MPEG-2).

11.1 Applications

There are many different types of applications for data broadcast over a transport stream, and each application type may require different types of data with different timing requirements.

For example, the type of data involved in sending Internet traffic is very different from that needed to provide a firmware update for a set top box. A non-real time update of pricing information has very different timing requirements from a quiz application where answers and questions must be sent in close synchronization with video/audio.

MPEG-2 provides a large variety of different techniques to send data. The choice of technique is a trade-off between optimizing bandwidth (and hence the cost of providing the service) while meeting the timing requirements of the application.

The types of applications can be loosely grouped by their real-time requirements and level of interactivity as described in the next sections.

11.1.1 Program Related Data

The base MPEG-2 specification does not have any provision for an EPG that can give information about the TV channels or individual programs being carried on the transport stream. The regional variants ATSC, DVB and ARIB have used the MPEG-2 private table syntax such as the EIT to provide additional data about programs. These tables are required to be broadcast at regular intervals and give start times, synopsis and other information about services and programs.

However even this level of information may not be enough and so there are a number of proprietary EPGs that extend the amount of information available and also provide facilities like enhanced searching, favorite channels and other augmented services. These enhanced EPGs are usually permanently resident on the set top box and use a combination of the standard table information and proprietary data formats. Usually this sort of program information is not time critical and so cheap low bit rate techniques are used. In many cases the data is designed to be cached by the set top box so it appears to be immediately available to the user, even if the actual total transmission time is relatively long.

11.1.2 Opportunistic Data

It is rarely possible to utilize the complete bandwidth of a transport stream with video and audio streams, not least because of the need to handle the complex constraints of remultiplexing or table insertion. Opportunistic data systems make use of this spare capacity by replacing some null packets with useful data. However the bandwidth of this data cannot be guaranteed and may be very small. Hence it can only be used for applications with no real-time constraints.

Some applications using this sort of data could be file transfers such as price lists or stock level data distribution via satellite to all company locations. The only constraint is that the transfer must take place overnight and so the low bandwidth and hence cheap opportunistic data services can be used.

11.1.3 Network Data

A very common application is to simply use the broadcast transport stream as a carrier of network data. In the simplest case, a satellite link would provide a very high bandwidth interconnect between two geographically separated companies. This network link could be used to carry virtually any type of network protocol.

However the most common current use is for high speed Internet downloads using just the IP protocol. In this scenario a low-speed Internet connection is used to send commands and page requests to an Internet server. However the Internet server will then route any requested data via the satellite link at speeds much higher than that possible by an ISDN or average broadband connection.

A variant on this application is used when only a slow back channel is available. In this case, for example, data related to the current TV program can be delivered on demand over the transmitted signal without viewers being aware that they are accessing an Internet site.

11.1.4 Enhanced TV

In an enhanced TV application there is no back channel, and so all data required must be sent over the transmitted transport stream. Although this imposes obvious constraints on the design and range of an application, it is still possible to produce a very wide range of games, quizzes and infomercials where the viewer may be completely unaware that all interaction is with the TV only. This is especially the case with pre-recorded material, where it is relatively straightforward to synchronize data pop-ups, such as quiz questions and answers, with the audio/video.

11.1.5 Interactive TV

The key difference between enhanced and interactive TV is that interactive TV has a back channel to send or receive highly personalized data. This greatly increases the range of applications that can be supported, including the ability to provide real time interaction with other viewers, such as multi-player gaming or voting, as well as truly personalized feedback.

A further extension of this is when high-speed Internet connection is combined with a true back channel. This offers seamless integration of personalized data with broadcast television, while also permitting true VOD delivery of programming or other material.

11.2 Content Encapsulation

The first stage in a data broadcast is to encapsulate the data into a form suitable for transmission on a transport stream. There are a wide variety of different data types and so the MPEG-2 standards provide a number of different encapsulation methods. The various country specific standards such as the DVB and ATSC have further enhanced the basic MPEG-2 options to meet regional requirements, but these all build upon the core MPEG-2 standards.

11.2.1 MPEG Data Encapsulation

11.2.1.1 Data Piping

Data piping is used for simple asynchronous delivery of data to a target set top box on the network. Its main application is to send proprietary data in closed systems where the target set top box has been pre-programmed to look for specific data on certain PIDs. Data is carried directly in the payload of MPEG-2 TS packets without any timing information. A typical application might be a nightly update of product stock levels to all sales outlets throughout a region.

11.2.1.2 Data Streaming

Data streaming is used for the end-to-end delivery of data in asynchronous, synchronous or synchronized fashion. Data is carried as PES packets in a similar method to video and audio services. In fact, video and audio are really specific examples of a synchronous data streaming service. As PES packets are signaled in the service information (SI) tables, and can carry timing information, this approach is more flexible than data piping but in practice is used in a similar fashion in proprietary closed systems.

11.2.1.3 DSMCC – Digital Storage Medium Command and Control

The MPEG-2 DSM-CC specification (ISO/IEC 13818-6) provides further ways of broadcasting data in the sections of a standard MPEG-2 private table. It was originally devised as a way of supporting VOD delivery of program material across a network on a Transport Stream. The protocol has been extended to be able to cope with both on-demand delivery (using the MPE paradigm) as well as periodic delivery (using the carousel paradigm) of data across multiple network providers

11.2.1.4 MPE – Multi-protocol Encapsulation

Multi-protocol encapsulation (MPE) allows a datagram of any communication protocol to be transmitted in the section of a DSM-CC table via a transport stream. A datagram is a logical structure that contains all defining information about the data, i.e., its size and contents, where it should be going and how it should get there.

The most common application is Internet traffic where the TCP/IP datagram carries information about the logical (IP) addresses of the source and destination (target) as well as the Media Access Control (MAC) address (a unique network address) of the target. However MPE supports nearly any type of network protocol and is certainly not restricted to only TCP/IP data.

11.2.1.5 Carousels

Carousels are intended for the periodic transmission of information over a transport stream. Although the content of a carousel can be changed in response to a request from a target user it is more usual for the carousel to be regularly repeated regardless of whether any target is listening or needs that data at that moment. A target that needs a specific data item is expected to simply wait until it is retransmitted.

There are two different types of carousels, object carousels and data carousels, and the main differences between them are that:

- Data carousels contain only modules of data of *unspecified* content; it is up to the target to know what to do with the data it receives.
- Object carousels contain *identifiable* data objects such as pictures, text files, or executable application files and contains a directory listing of all objects in the carousel.

Data carousels are often used for downloading new system software to a set top box whereas an object carousel is used for shopping services, EPGs and to send applications and games.

In both data and object carousels, items are repeated at periodic intervals. However, object carousels make it easy to vary the repetition rate of individual objects. For example, the EPG for the next hours viewing may repeat far more often than that for next month. The repetition rates for objects may be a commercial decision made by the service provider to maximize bandwidth utilization.

Both object and data carousels are based upon the DSM-CC extensions to the MPEG-2 specification ISO13818-6, with specific extensions for the DVB, ARIB and ATSC systems.

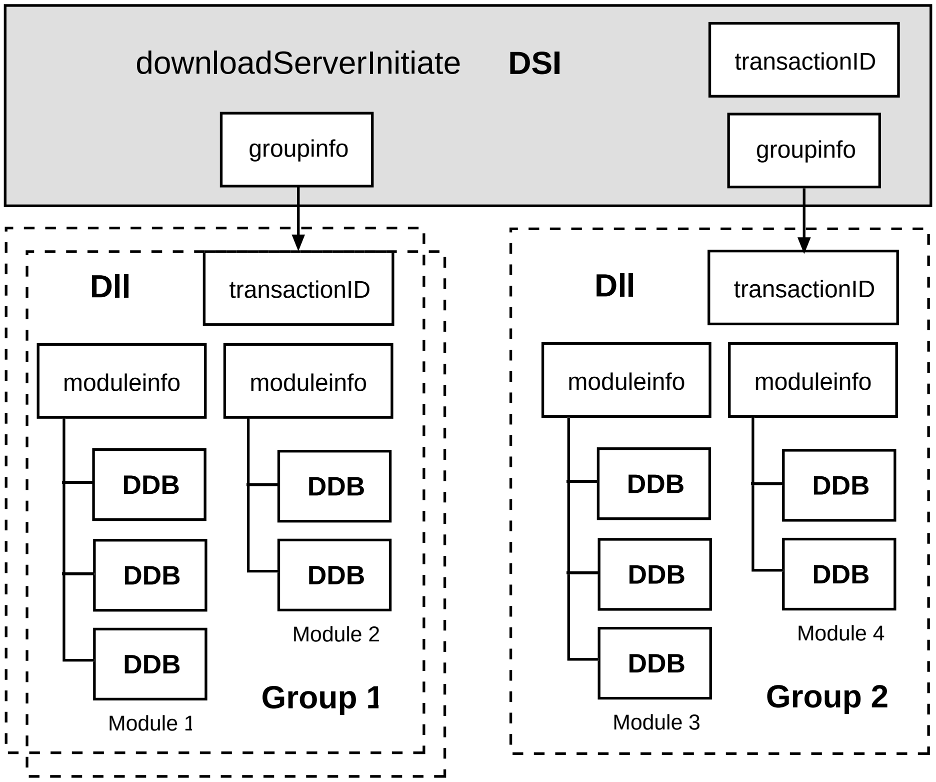


Figure 11-1.

11.2.1.6 Data Carousels

A data carousel does not contain any individual data items or directory structure but a single monolithic chunk of data. It is up to the target user to know what the data is and what to do with it.

The structure is shown in Figure 11-1. A complete single item of data is defined as a "module." Transmission modules are split up into one or more blocks. Each block is sent as a section in the payload of a DownloadDataBlock (DDB) message, which follows the MPEG-defined private table syntax. DDB messages can be sent in any order or at any periodicity; hence a mechanism is needed to identify which DDBs belong to what modules.

A DownloadInfoIndication (DII) message is used to link the DDBs for a module together. The information for more than one module can be in a single DII message; this forms a Group. Usually a group will contain logically related data modules.

If there are more related modules than can be grouped together into a single DII message then a Supergroup can be formed from multiple DII messages. These are referenced by a DownloadServerInitiate (DSI) message.

A one-layer data carousel contains a small number of modules referenced in a single DII.

A two-layer data carousel contains DSI messages referencing one or more DII messages. It may be smaller or larger than a single carousel.

A typical use for a 2-layer carousel would be for multi-language support. One group might convey the executable program in one module along with English text in a second module. The second group could then just carry a single module with just French text, saving the overhead of repeating the application module.

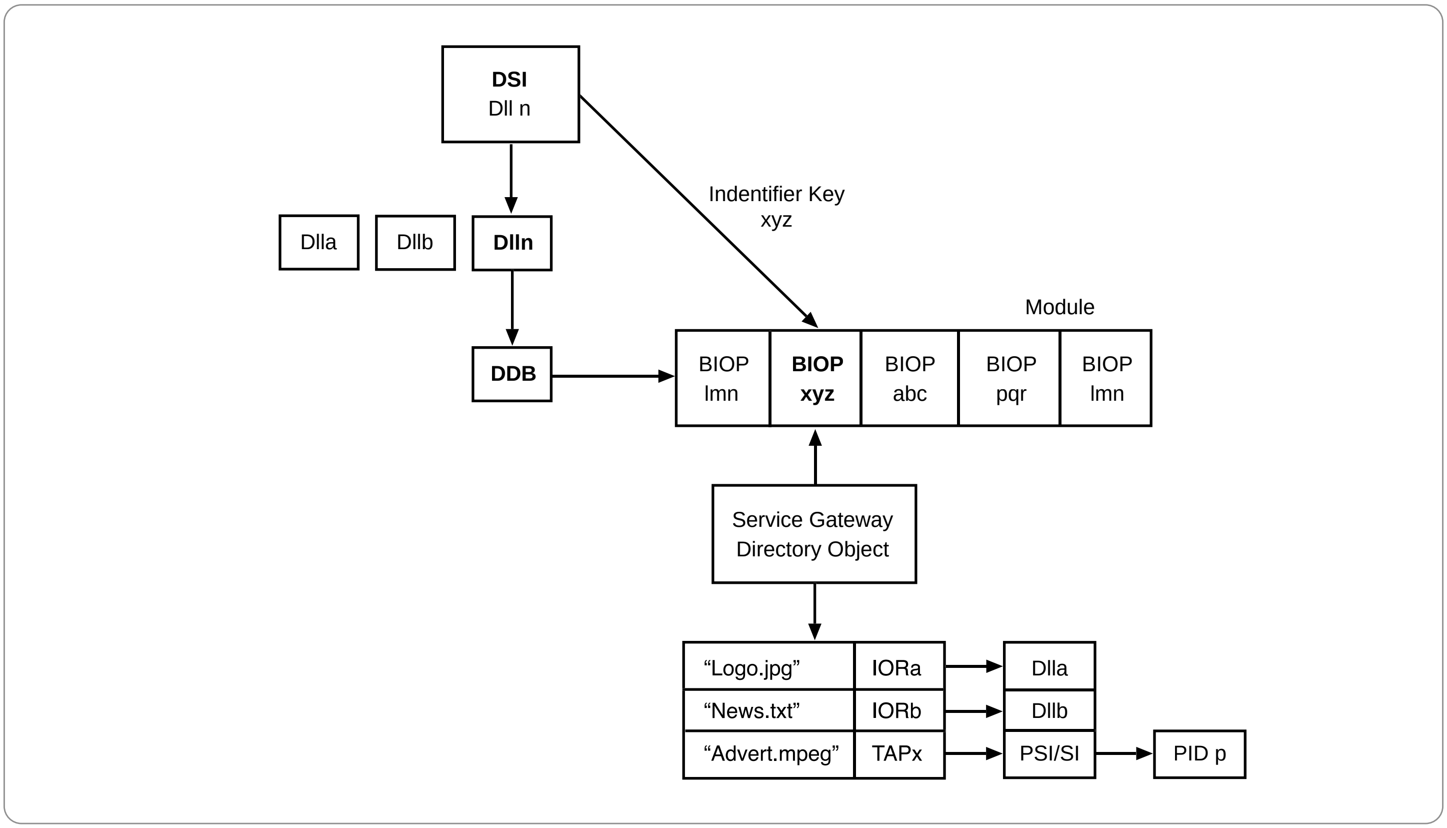


Figure 11-2.

11.2.1.7 Object Carousels

Object carousels are used to broadcast individually identifiable items of identified data from a server to a receiver. These items are called *objects* and may be pictures, text files, programs, a pointer to a video PID, a directory listing or *service gateway* of what is available in the carousel. Related objects grouped and sent together as a single carousel form a *service domain.* Objects can be sent as often as required and different objects may have very different repetition rates.

A key feature of object carousels is that all objects are sent using the BIOP (broadcast inter-ORB protocol) paradigm. Conventional software developers have been using ORB (object request brokerage) for many years. BIOP extends the basic system to support identifying and using objects in a broadcast environment across different networks from different service providers.

In essence, a BIOP is a method to exchange information about an object being broadcast in the carousel. The BIOP may contain the object or may simply provide a pointer to the object. The BIOP may also indicate how to use the object, including providing a link to where to download the application software needed to use the object.

Object carousels are similar to data carousels in that groups of objects are combined together to form modules. The basic data carousel methodology is then used to transmit that data using blocks, modules and DIIs. The key difference is that the DSI is used to point directly to the Service Gateway directory object, which can then be used to find all other objects in the carousel. This arrangement is shown in Figure 11-2.

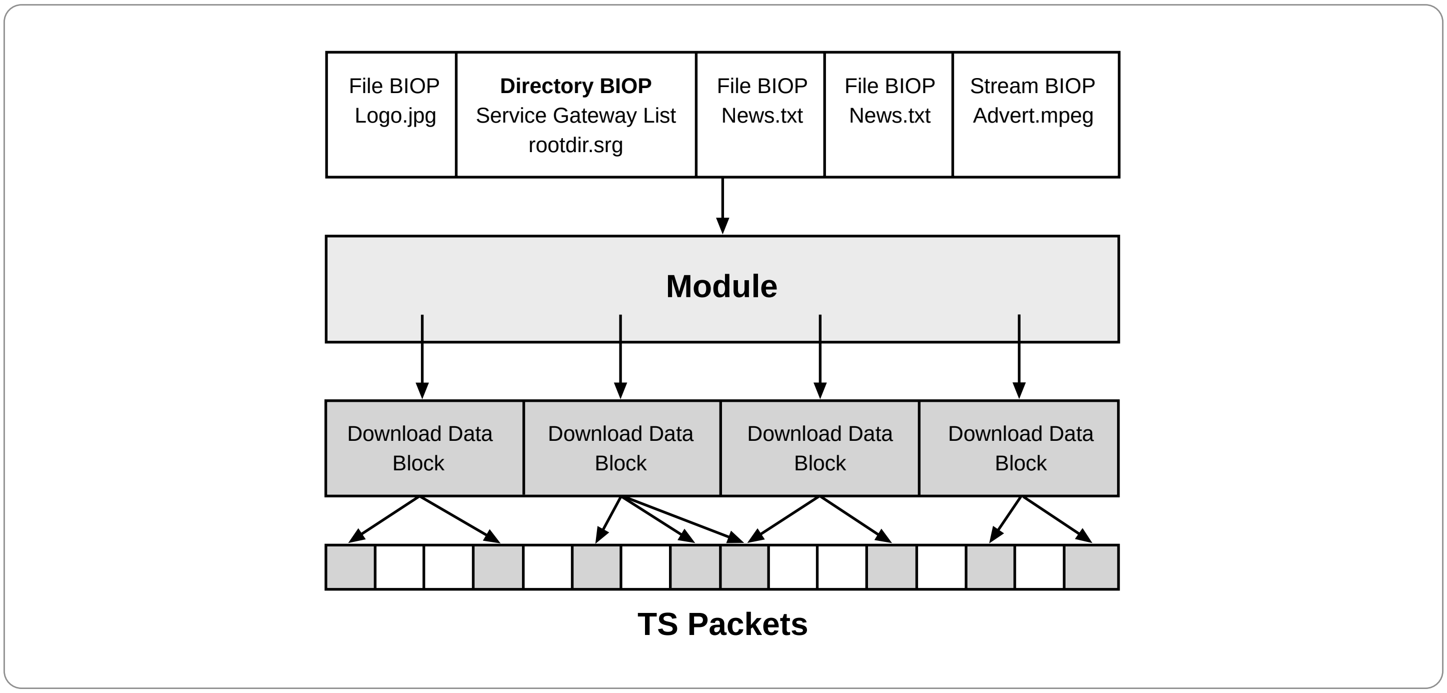


Figure 11-3.

11.2.1.8 How Object Carousels Are Broadcast

A full explanation is beyond the scope of this document; the following description is a brief and much-simplified overview. (Also see Figure 11-3.)

Directory, file and stream objects are sent in the same method as data carousels i.e., in modules split into blocks are sent as sections in the payload of a DownloadDataBlock (DDB).

A DownloadServerInitiate (DSI) message contains the location of a special directory object called the service gateway. DSI messages are referenced in the SI and so form the starting point to work out what is in a specific object carousel. The DSI references the DownloadInfoIndication (DII) that references the DDB that contain the module in which the service gateway object is sent.

Objects are referenced in a directory object using IORs (inter-operable object references). This contains all the information needed to access an object in the same service domain or on another object carousel (including those broadcast on other Transport Streams).

The name given to the structure in the IOR that describes the location of an object is called a profile body that comes in two flavors:

BIOP profile body – used only for objects within this service domain.

Lite Options Profile Body – used for objects on other servers or transport streams.

An IOR can contain more than one profile body if the object is available on more than one carousel and the set top box can choose the easiest/quickest one to access.

Taps are used to solve the problem that the actual PIDs used to broadcast DIIs, DDBs and video/audio streams are not known until immediately before transmission. Within the carousel therefore all references to PIDs are only made in terms of a tap; the association between a tap and a real PID is made in the SI. This vastly simplifies re-multiplexing streams across different networks.

11.2.1.9 MPEG-2 Data Synchronization

There is a need for data broadcasts to be synchronized in some way with programs being broadcast. It is not really practical to use the real-time delivery of a datum as the synchronization method, except in very non-critical real time applications such as updating sports scores where a few seconds or more error is of no practical significance. However even a second or two could have a very big impact on, for example, quiz shows where revealing the answer too early could have serious consequences.

MPEG-2 provides different timing mechanisms for the different types of data encapsulation. Data piping and MPE do not support any form of timing mechanism apart from near real-time delivery of the data.

Data streaming PES packets can contain presentation time stamp (PTS) and possibly decoding time stamp (DTS) timestamps for synchronization with the system clock derived from the PCR values in the stream. The mechanism is exactly the same as for video or audio synchronization and decode.

MPEG-2 data carousels have no timing information. However, object carousels can contain a special object called a “stream event” which contains timing information relative to the normal play time (NPT) of an individual television program. The NPT is not the same as the PCR as the NPT clock can be paused during, for example, a commercial break. In other words the NPT of a program can remain in full synchronization with the program timeline, regardless of when it is transmitted or how it is split into commercial breaks.

11.2.2 DVB Data Encapsulation

DVB has adopted the standard MPEG-2 encapsulation methods with only very minor variations, mainly to remove possible ambiguities that emerged from the original specifications (EN 301 192). These include specifying a slightly different MPE format and imposing restrictions on the maximum data PES packet size. DVB has also taken responsibility for ensuring that there can be no ambiguity between data services that use IORs by providing unique allocation of network IDs and server IDs.

DVB has also defined two specific data streaming PES formats for DVB subtitling and for DVB teletext and have a standardized format for using data carousels to transmit set top box (STB) firmware updates. The DVB have also defined a number of data descriptors and additional tables that are discussed in the signaling and announcement sections below. Most useful of these additions is the ability to use a descriptor to provide a simple mask for the MAC addresses in an MPE datagram. This enables simple support for narrow, multi or broadcast MPE services.

11.2.3 ATSC A/90 Data Encapsulation

The ATSC A/90 specification was written several years later than the DVB system and includes some significant differences from both the DVB and the MPEG-2 standards. First, A/90 does not use object carousels and the CORBA/IOR system at all but instead uses a separate table, the Network Resources Table (NRT), to provide the binding information that links a referenced datum with its real location. Apart from arguably simplifying the system operation it also enables direct support for Internet URLs that is not supported by DVB or MPEG. The mechanism is shown in Figure 11-4.

Secondly, the A/90 uses only data carousels that can effectively contain only a single object. The data carousel format has been extended to optionally include a DSM-CC adaptation header that may contain a PTS to enable synchronous data delivery. The MPEG concept of NPT within a program is not supported.

A/90 also defines additional data types including its own version of DSM-CC MPE as well allowing IP data to be streamed using the PES format. It also supports a wider variety of timing models for the different types of data than DVB.

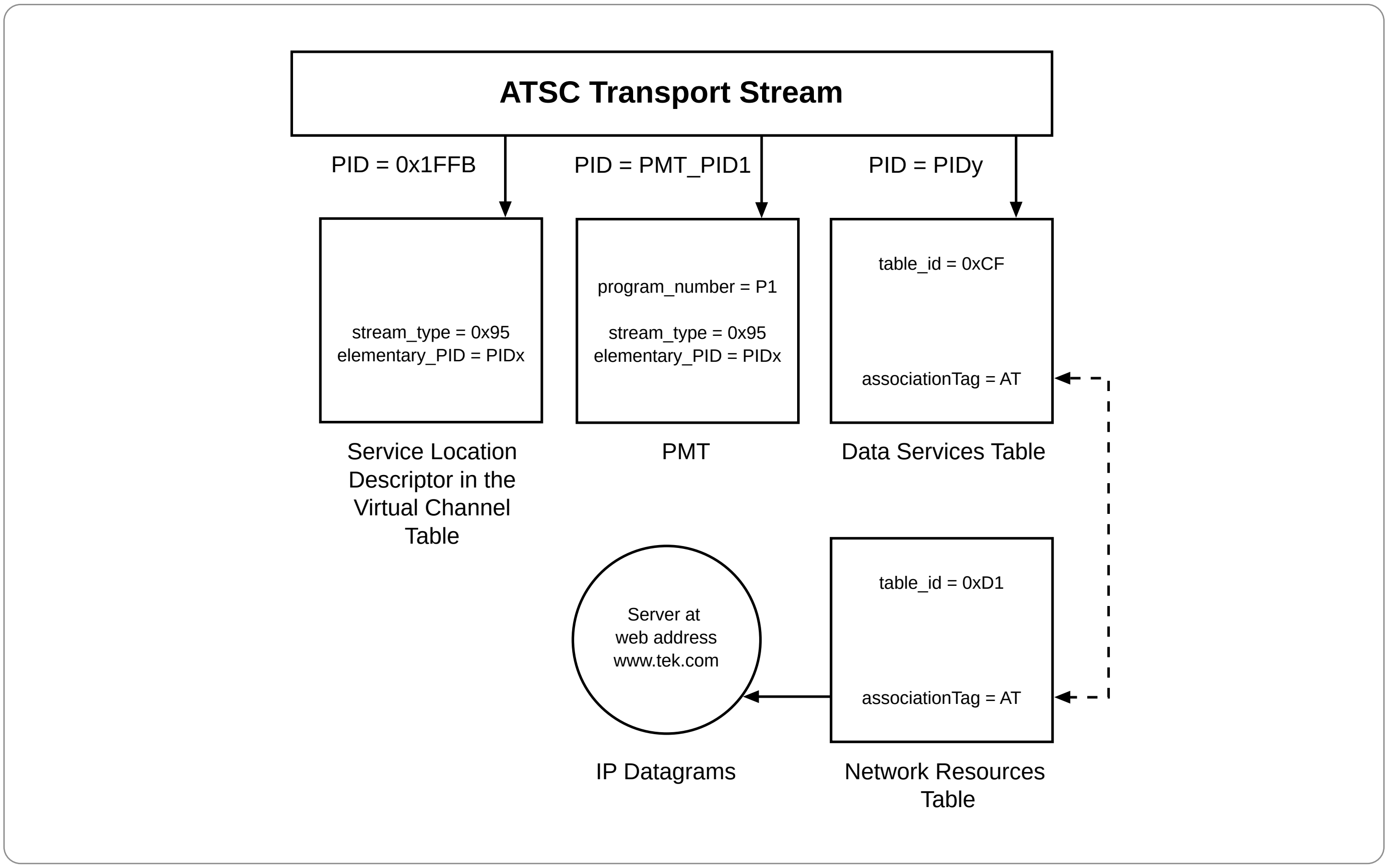


Figure 11-4.

11.2.4 ARIB Data Encapsulation

The Japanese ARIB standard was defined after the A/90 standard and is arguably the simplest of all systems. It does not support data piping or object carousels. Instead it uses the data carousels format to send one or more entities by imposing a specific directory structure upon the data carousel content. ARIB does not allow references to any entities that are not directly referenced in the PMT and so has no need for either CORBA (common object request broker architecture) or NRT type linkages.

11.3 Data Content Transmission

Data services are first encapsulated according to the relevant national or international standard and then transmitted via a transport stream to an STB.

However in order to make use of a data service the STB must first know that a data service exists and when it will be available (“announcement”). Secondly it must then be possible to find and identify the various components of the data service (“signaling”) when it is actually being broadcast. MPEG-2 only defines the PAT and PMT so in effect all announcement and signaling can only be made by inspection of the elementary stream type and some very basic descriptors.

Although usable there are a large number of limitations to just using only the PAT and PMT, not least that scheduling of data services or events is not possible. Both the DVB and ATSC have therefore defined additional tables that significantly increase the amount of information provided about any data services present. Note that in all cases the data-carrying PIDs must be signaled as present with an entry in at least one PMT.

11.3.1 DVB Announcement

The DVB consider data services to be either associated with an individual event (for example a single quiz program) or to be part of a service such as a sports channel. In effect it conforms to the MPEG-2 paradigm where a data service is logically indistinguishable from a conventional audio or video stream.

It therefore has simply defined a number of additional descriptors that are placed into either the EIT or the SDT table which announce that a data service or event will be available. The actual location of data services and components is then signaled using the techniques described above.

The only new table is the application information table, which is defined for use with MHP services only. This contains a list of all applications within a carousel, a pointer to their boot class and optionally allows applications to be auto-started or killed upon a channel change.

11.3.2 ATSC Announcement

In contrast to the DVB the A/90 specification treats data events and services totally separately from video and audio services and has added a number of tables to support this paradigm. These are the data services table (DST), which list the data services being sent and the data event table (DET). The DET is exactly analogous to the EIT for conventional program scheduling information and uses an identical syntax. A third table, the long term services table, is similar to the DET but provides long term schedule information beyond the maximum 16 days provided by the DET.

11.4 Content Presentation

The final stage of data broadcasting is running the application on the STB and presenting information to the viewer. It should be noted that a transport stream is entirely agnostic to the actual content it is carrying, provided of course that it is encapsulated in the correct protocol for video, audio or data. In other words the following section is almost completely divorced from MPEG-2, DVB or ATSC data broadcasting standards as applications and data are just delivered by the transport stream to be used by the STB.

11.4.1 Set Top Box Middleware

In some cases, such as Internet data, the environment in which the data is to be used is self-defined or relatively unimportant. However in most cases, such as games and applications it is critical that the data application is constructed to run properly on the target STB. In most cases the STB will have a software application program interface (API) to run that connects the high level function calls from the application to the software drivers that control the real hardware in the box and link to the data in the transport stream. This software layer, which sits between the application and the drivers, is known as the “middleware.”

There are however a very large variety of STB in use in the world. Their performance and feature sets run from very basic boxes that can only just decode and display a basic standard definition picture through to the latest units with powerful processing capabilities. Some modern units have high performance graphics chip sets and run full operating systems such as Linux or Windows. They are equipped with hard disk drives for recording programs, have DVD ROMs for DVD replay and can run conventional programs such as Microsoft or Star Office packages.

A simple graphic designed for display on a new high performance box might not even be displayable on an older unit. Application developers would have to write programs completely differently using different programming languages for each type of set top box, middleware and operating system, clearly inefficient and increasing the development cost.

There have been a number of attempts to standardize boxes or at least their minimum functionality, where the functionality of the box is clearly defined and controlled by the broadcasters themselves in a very closed fashion.

Another approach has been taken by companies that provide a complete solution by offering a middleware along with the application development tools to develop compliant programs. These systems are more open in that the middleware can be licensed to run on a variety of different set top boxes with different performance or features as long as they meet the basic required functionality.

A proprietary data transmission scheme and a “walled garden” interactive service has been widely used as an interactive system, providing true interactivity via a back channel. Other companies offerings are also widely used throughout Europe and applications can be delivered using standard object carousels. There are many other proprietary systems.

A more open standard is an application environment defined by the Multimedia Hypermedia Experts Group (MHEG). A particular variant of this, MHEG-5, is optimized for low performance low memory applications and set top boxes in particular. MHEG-5 has had some success, notably in the UK where it is used for terrestrial broadcasts to STBs.

The current trend is therefore towards standardizing upon an open middleware with a clearly defined programming API. This has the benefit that the consumer can choose from a wide variety of set top boxes whilst enabling application developers to reuse code. Recently a number of front-runners for a global standard have emerged, including the Multimedia Home Platform (MHP) from the DVB. In the USA the Advanced TeleVision Enhancement Forum (ATVEF) and the Digital TV Application Software Environment (DASE) are the leading open standards.

11.4.2 The DVB Multimedia Home Platform (MHP)

The multimedia home platform (MHP) defines a very comprehensive API with several different profiles and levels to allow for different performance and cost STB and services. Enhanced TV and Interactive TV (with back channels) are supported, and it also provides support for Internet connections, which is otherwise not directly supported by the other DVB standards.

MHP applications are delivered using standard DVB object carousels and announced using the AIT table described earlier. This allows applications to be auto-started, paused or stopped and for resources to be explicitly cached in order to speed application boot times. The MHP standard is based on Java technology and encompasses a set of APIs derived from existing standards such as JavaTV, HAVI (home audio video interoperability) and DAVIC (Digital Audio Visual Council).

Applications can be either Java- or HTML-based and there are a number of MHP software development kits (SDK) available from several different suppliers. Applications will have to undergo a compliance test and be electronically signed to ensure both the interoperability and security of the system.

The MHP logo is also protected and STBs must undergo a compliance process before they can be sold with the logo. Regular interoperability tests have taken place over the last two years to ensure that all MHP applications can run on all MHP set top boxes from a variety of leading manufacturers.

MHP is the de facto standard in Europe, and services officially started in Finland in August 2001. There are a large number of test services currently being broadcast through out Europe. Germany will also use MHP from the middle of 2002, with other countries following over the next few years.

However, MHP is not just a European standard as it has been adopted throughout the world. For example it is mandated in South Korea for satellite data broadcasting and in Australia for terrestrial broadcasting. In February 2002 the open cable applications platform (OCAP) organization in the USA adopted MHP as the American digital cable broadcasting standard. Many other countries and organizations are expected to adopt MHP over the next few years.

The global impact of MHP must not be underestimated. The current market leaders for interactive TV services have committed to providing an MHP plug-in for use on their own proprietary middleware during 2002, effectively opening up all their proprietary closed set top boxes to a global market. With countries already committed to MHP throughout Europe there is already a potential market of nearly 1 billion viewers, which augurs well for the future of this open standard.

11.4.3 ATVEF DASE

It is sometimes said that the difference between ATVEF (Advanced Television Enhancement Forum) and DASE (DigitalTV application software environment) is that the former recommends the use of HTML but allows the use of Java, while the latter recommends Java and allows HTML. However there are some other fundamental differences.

The ATVEF specification was written by a cross-industry group of manufacturers in order to define standardized functionality that must be provided in order to enable interactive content to run on set top boxes, PC-based receivers and interactive TVs. The enhanced content specification (ECS) was the result.

The ATVEF ECS defines HTML as the foundation for creating interactive TV content, although JavaScript is also allowed. A number of other basic functionalities are also required such as using the portable network graphics (.png) format as the standard picture interchange format. ATVEF therefore has substantial commonality with Internet web page design, with obvious benefits for application developers.

One of the ATVEF's strengths is that the transport mechanism is not completely defined or assumed by the standard. It is perfectly capable of running across analog and digital transmissions schemes, with or without video content. It achieves this by defining a transport-independent content format and by the use of IP as the reference binding. Any transmission scheme that uses IP can therefore be used. This has enabled ATVEF to run across NTSC schemes as well as MPEG Transport Streams, and provides a simple mechanism for a return path. ATVEF is in use in the USA on both analog and digital television systems.

11.4.4 DASE

By contrast, DASE is a much more complex middleware specification that was developed by the ATSC in the USA. As the ATSC also developed the A/90 data broadcasting standard, the DASE specification provides a binding to the A/90 transmission scheme as well as defining the API that a DASE application will run on. The DASE-1 standard was released as a candidate standard in early 2002.

DASE incorporates a *signaling scheme* and an *announcement scheme* as well as the actual application data content itself, which is called the *data essence.* Two types of data essence are allowed, along with the environment and functionality that they can use:

- Declarative data essence based upon .xdml and .xml markup text and scripts.
- Procedural data essence based upon javatv xlets.

A DASE receiver provides an engine to handle both types of data essence along with common core functionality such .jpeg or .png codecs that are shared between the two systems. DASE also provides direct links to audio or video content being transmitted and defines a number of graphical screen formats to display interactive content on, including HDTV resolutions.

Applications can be initialized, activated, suspended or uninitialized, similar to the MHP application lifecycle paradigm. Resources can also be cached as required.

Although DASE has the weight of the ATSC behind it, it is a relatively late entrant into the field and so has not yet been widely adopted.

Section 12 – MPEG Testing

The ability to analyze existing transport streams for compliance is essential, but this ability must be complemented by an ability to create transport streams that are known to be compliant.

12.1 Testing Requirements

Although the technology of MPEG differs dramatically from the technology that preceded it, the testing requirements are basically the same. On an operational basis, the user wants to have a simple, regular confidence check that ensures all is well. In the event of a failure, the location of the fault needs to be established rapidly. For the purpose of equipment design, the nature of problems needs to be explored in some detail. As with all signal testing, the approach is to combine the generation of known valid signals for insertion into a system with the ability to measure signals at various points.

One of the characteristics of MPEG that distances it most from traditional broadcast video equipment is the existence of multiple information layers, in which each layer is hoped to be transparent to the one below. It is very important to be able to establish in which layer any fault resides to avoid a fruitless search.

For example, if the picture monitor on an MPEG decoder is showing visible defects, these defects could be due to a number of possibilities. Perhaps the encoder is faulty, and the transport stream is faithfully delivering the faulty information. On the other hand, the encoder might be fine, but the transport layer is corrupting the data. In DVB, there are even more layers such as energy dispersal, error correction, and interleaving. Such complexity requires a structured approach to fault finding, using the right tools. The discussion of protocol analysis of the compressed data in this primer may help the user derive such an approach. Reading the discussion of another important aspect of testing for compressed television, picture-quality assessment, may also be helpful. This later discussion is found in the Tektronix publication, “A Guide to Video Measurements for Compressed Television Systems.”

12.2 Analyzing a Transport Stream

An MPEG transport stream has an extremely complex structure, but an analyzer such as the AD953 can break down the structure in a logical fashion such that the user can observe any required details. Many general types of analysis can take place in real time on a live transport stream. These include displays of the hierarchy of programs in the transport stream and of the proportion of the stream bit rate allocated to each stream.

More detailed analysis is only possible if part of a transport stream is recorded so that it can be picked apart later. This technique is known as deferred-time testing and could be used, for example, to examine the contents of a time stamp.

When used for deferred-time testing, the MPEG transport-stream analyzer is acting like a logic analyzer that provides data-interpretation tools specific to MPEG. As with all logic analyzers, a real-time triggering mechanism is required to determine the time or conditions under which capture will take place. Figure 12-1 shows that an analyzer contains a real-time section, a storage section, and a deferred section. In real-time analysis, only the real-time section operates, and a signal source needs to be connected. For capture, the real-time section is used to determine when to trigger the capture. The analyzer includes tools known as filters that allow selective analysis to be applied before or after capture.

Once the capture is completed, the deferred section can operate on the captured data and the input signal is no longer necessary. There is a good parallel in the storage oscilloscope which can display the real-time input directly or save it for later study.

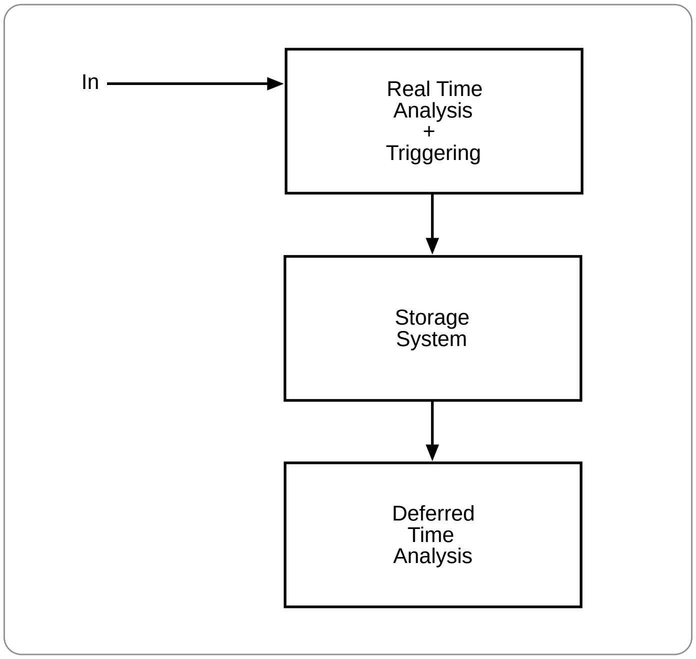
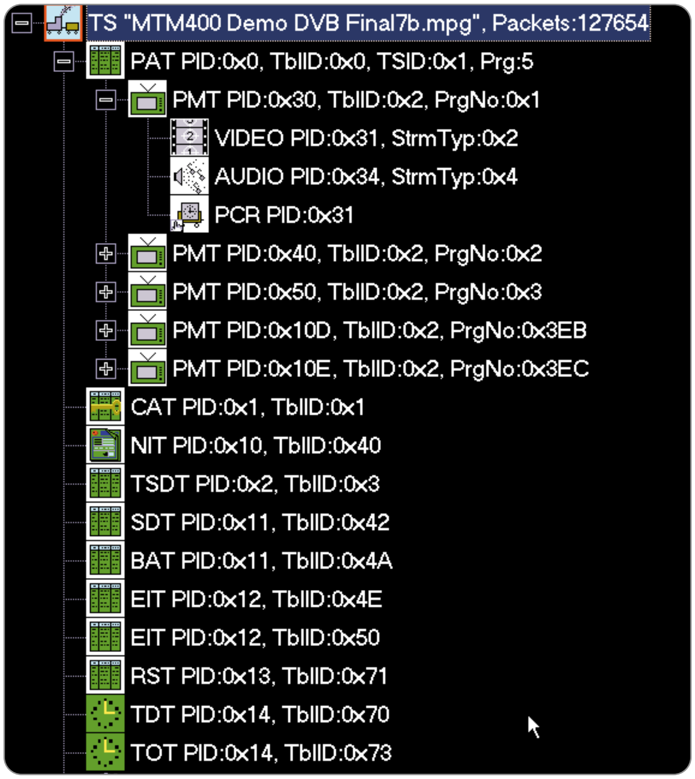


Figure 12-1.



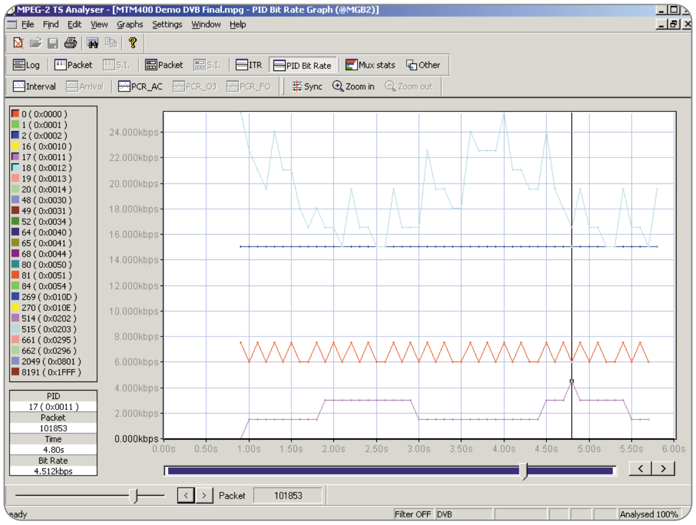
```
TS "MTM400 Demo DVB Final7b.mpg", Packets:127654
    PAT PID:0x0, TblID:0x0, TSID:0x1, Prg:5
        PMT PID:0x30, TblID:0x2, PrgNo:0x1
            VIDEO PID:0x31, StrmTyp:0x2
            AUDIO PID:0x34, StrmTyp:0x4
            PCR PID:0x31
        PMT PID:0x40, TblID:0x2, PrgNo:0x2
        PMT PID:0x50, TblID:0x2, PrgNo:0x3
        PMT PID:0x10D, TblID:0x2, PrgNo:0x3EB
        PMT PID:0x10E, TblID:0x2, PrgNo:0x3EC
    CAT PID:0x1, TblID:0x1
    NIT PID:0x10, TblID:0x40
    TSDT PID:0x2, TblID:0x3
    SDT PID:0x11, TblID:0x42
    BAT PID:0x11, TblID:0x4A
    EIT PID:0x12, TblID:0x4E
    EIT PID:0x12, TblID:0x50
    RST PID:0x13, TblID:0x71
    TDT PID:0x14, TblID:0x70
    TOT PID:0x14, TblID:0x73
```

Figure 12-2.

12.3 Hierarchic View

When analyzing an unfamiliar transport stream, the hierarchic view is an excellent starting point because it enables a graphic view of every component in the stream. Figure 12-2 shows an example of a hierarchic display such as that provided by the Tektronix MTX100. Beginning at top left of the entire transport stream, the stream splits and an icon is presented for every stream component. Table 12-1 shows the different icons that the hierarchical view uses and their meaning. The user can very easily see how many program streams are present and the video and audio content of each. Each icon represents the top layer of a number of lower analysis and information layers.

The analyzer creates the hierarchic view by using the PAT and PMT in the PSI data in the transport stream. The PIDs from these tables are displayed beneath each icon. PAT and PMT data are fundamental to the operation of any demultiplexer or decoder; if the analyzer cannot display a hierarchic view or displays a view which is obviously wrong, the transport stream under test has a PAT/PMT error. It is unlikely that equipment further up the line will be able to interpret the stream at all.

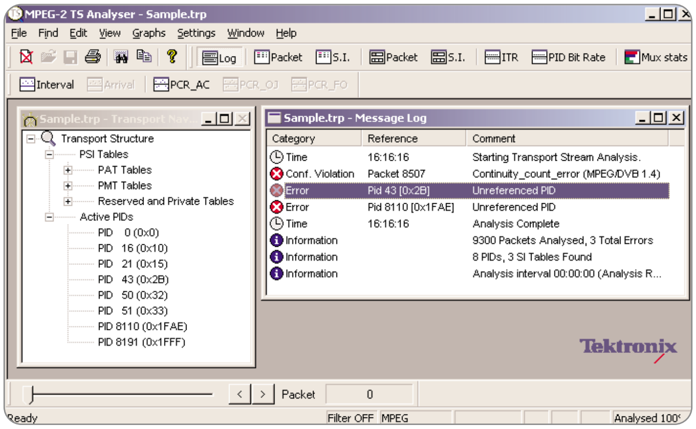


MPEG-2 TS Analyser - [MTM400 Demo DVB Final.mpg - PID Bit Rate Graph (@MGB2)]

PID
17 (0x0011)
Packet
101853
Time
4.80s
Bit Rate
4.512kbps

Packet 101853

Figure 12-3.



MPEG-2 TS Analyser - Sample.trp

Sample.trp - Transport Nav...

- Transport Structure
  - PSI Tables
    - PAT Tables
    - PMT Tables
    - Reserved and Private Tables
  - Active PIDs
    - PID 0 (0x0)
    - PID 16 (0x10)
    - PID 21 (0x15)
    - PID 43 (0x2B)
    - PID 50 (0x32)
    - PID 51 (0x33)
    - PID 8110 (0x1FAE)
    - PID 8191 (0x1FFF)

Sample.trp - Message Log

Category	Reference	Comment
Time	16:16:16	Starting Transport Stream Analysis.
Conf. Violation	Packet 8507	Continuity_count_error (MPEG/DVB 1.4)
Error	Pid 43 [0x2B]	Unreferenced PID
Error	Pid 8110 [0x1FAE]	Unreferenced PID
Time	16:16:16	Analysis Complete
Information		9300 Packets Analysed, 3 Total Errors
Information		8 PIDs, 3 SI Tables Found
Information		Analysis interval 00:00:00 (Analysis R...

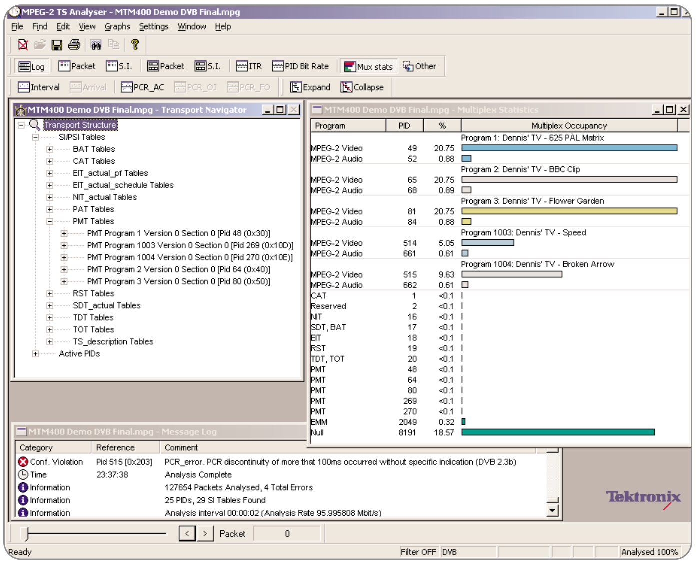
Packet 0

Figure 12-4.

The ability of a demux or decoder to lock to a transport stream depends on the frequency with which the PSI data are sent. The PSI/SI rate option shown in Figure 12-3 displays the frequency of insertion of system information. PSI/SI information should also be consistent with the actual content in the bit stream. For example, if a given PID is referenced in a PMT, it should be possible to find PIDs of this value in the bit stream. The consistency-check function makes such a comparison. Figure 12-4 shows a consistency-error from a stream including two unreferenced packets.

Table 12-1. Hierarchical View Icons

Icon	Element Type
TS	Multiplex transport packets. This icon represents all (188- and 204-byte transport packets) that make up the stream. If you visualize the transport stream as a train, this icon represents every car in the train, regardless of its configuration (for example, flat car, boxcar or hopper) and what it contains.
	Transport packets of a particular PID (Program ID). Other elements (tables, clocks, PES packets) are the "payload" contained within transport packets or are constructed from the payload of several transport packets that have the same PID. The PID number appears under the icon. In the hierarchic view, the icon to the right of this icon represents the payload of packets with this PID.
	Transport packets that contain independent PCR clocks. The PID appears under the icon.
PAT	PAT (program association table) sections. Always contained in PID 0 transport packets.
PMT	PMT (program map table) sections.
	NIT (Network Information Table). Provides access to SI Tables through the PSI/SI command from the Selection menu. Also used for Private sections. When the DVB option (in the Options menu) is selected, this icon can also represent SDT, BAT, EIT and TDT sections.
PES	PES (packetized elementary stream). This icon represents all packets that, together, contain a given elementary stream. Individual PES packets are assembled from the payloads of several transport packets.
VIDEO	Video elementary stream.
AUDIO	Audio elementary stream.
010010 1001110 1110001 DATA	Data elementary stream.
ECM	ECM (entitlement control message) sections.
	EMM (entitlement management message) sections.


MPEG-2 TS Analyser - MTM400 Demo DVB Final.mpg
File Find Edit View Graphs Settings Window Help
Log Packet S.I. Packet S.I. ITR PID Bit Rate Mux stats Other
Interval Arrival PCR_AC PCR_OJ PCR_FO Expand Collapse
MTM400 Demo DVB Final.mpg - Transport Navigator
MTM400 Demo DVB Final.mpg - Multiplex Statistics
MTM400 Demo DVB Final.mpg - Message Log
Packet 0
Ready Filter OFF DVB Analysed 100%
Tektronix

```
Transport Structure
  SI/PSI Tables
    BAT Tables
    CAT Tables
    EIT_actual_pf Tables
    EIT_actual_schedule Tables
    NIT_actual Tables
    PAT Tables
    PMT Tables
      PMT Program 1 Version 0 Section 0 [Pid 48 (0x30)]
      PMT Program 1003 Version 0 Section 0 [Pid 269 (0x10D)]
      PMT Program 1004 Version 0 Section 0 [Pid 270 (0x10E)]
      PMT Program 2 Version 0 Section 0 [Pid 64 (0x40)]
      PMT Program 3 Version 0 Section 0 [Pid 80 (0x50)]
    RST Tables
    SDT_actual Tables
    TDT Tables
    TOT Tables
    TS_description Tables
  Active PIDs
```

Program	PID	%	Multiplex Occupancy
			Program 1: Dennis' TV - 625 PAL Matrix
MPEG-2 Video	49	20.75	
MPEG-2 Audio	52	0.88	
			Program 2: Dennis' TV - BBC Clip
MPEG-2 Video	65	20.75	
MPEG-2 Audio	68	0.89	
			Program 3: Dennis' TV - Flower Garden
MPEG-2 Video	81	20.75	
MPEG-2 Audio	84	0.88	
			Program 1003: Dennis' TV - Speed
MPEG-2 Video	514	5.05	
MPEG-2 Audio	661	0.61	
			Program 1004: Dennis' TV - Broken Arrow
MPEG-2 Video	515	9.63	
MPEG-2 Audio	662	0.61	
CAT	1	<0.1	
Reserved	2	<0.1	
NIT	16	<0.1	
SDT, BAT	17	<0.1	
EIT	18	<0.1	
RST	19	<0.1	
TDT, TOT	20	<0.1	
PMT	48	<0.1	
PMT	64	<0.1	
PMT	80	<0.1	
PMT	269	<0.1	
PMT	270	<0.1	
EMM	2049	0.32	
Null	8191	18.57	

Category	Reference	Comment
Conf. Violation	Pid 515 [0x203]	PCR_error. PCR discontinuity of more that 100ms occurred without specific indication (DVB 2.3b)
Time	23:37:38	Analysis Complete
Information		127654 Packets Analysed, 4 Total Errors
Information		25 PIDs, 29 SI Tables Found
Information		Analysis interval 00:00:02 (Analysis Rate 95.995808 Mbit/s)

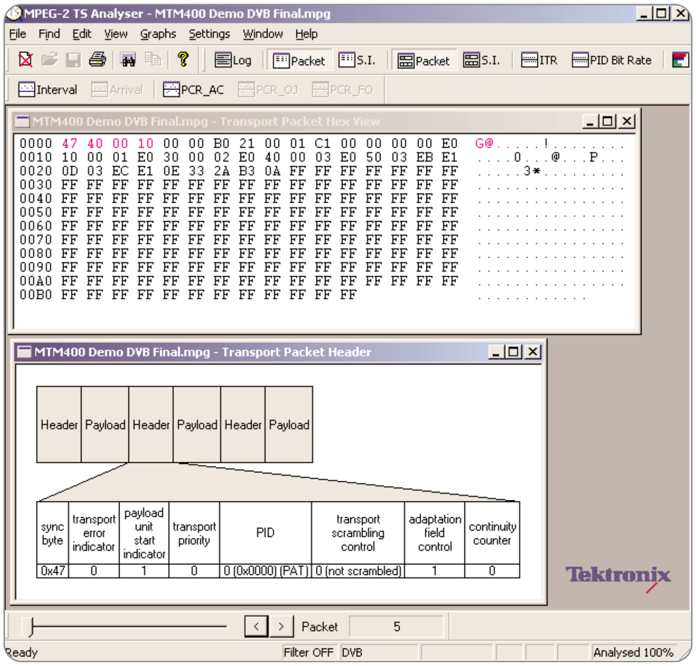
Figure 12-5.

A MUX allocation chart may graphically display the proportions of the transport stream allocated to each PID or program. Figure 12-5 shows an example of a MUX allocation chart display. The hierarchical view and the MUX allocation chart show the number of elements in the transport stream and the proportion of bandwidth allocated.

12.4 Interpreted View

As an alternative to checking for specific data in unspecified places, it is possible to analyze unspecified data in specific places, including in the individual transport stream packets, the tables, or the PES packets. This analysis is known as the interpreted view because the analyzer automatically parses and decodes the data and then displays its meaning. Figure 12-6 shows an example of an MPEG transport packet in hex view as well as interpreted view. As the selected item is changed, the packet number relative to the start of the stream can be displayed.

Figure 12-7 shows an example of a PAT in the interpreted view.


MPEG-2 TS Analyser - MTM400 Demo DVB Final.mpg
File Find Edit View Graphs Settings Window Help
Log Packet S.I. Packet S.I. ITR PID Bit Rate
Interval Arrival PCR_AC PCR_OJ PCR_FO
MTM400 Demo DVB Final.mpg - Transport Packet Hex View
MTM400 Demo DVB Final.mpg - Transport Packet Header
Header Payload Header Payload Header Payload
Packet 5
Ready Filter OFF DVB Analysed 100%
Tektronix

```
0000 47 40 00 10 00 00 B0 21 00 01 C1 00 00 00 00 E0   G@.....!........
0010 10 00 01 E0 30 00 02 E0 40 00 03 E0 50 03 EB E1   ....0...@...P...
0020 0D 03 EC E1 0E 33 2A B3 0A FF FF FF FF FF FF FF   ......3*........
0030 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF   ................
0040 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF   ................
0050 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF   ................
0060 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF   ................
0070 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF   ................
0080 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF   ................
0090 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF   ................
00A0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF   ................
00B0 FF FF FF FF FF FF FF FF FF FF FF FF               ............
```

sync byte	transport error indicator	payload unit start indicator	transport priority	PID	transport scrambling control	adaptation field control	continuity counter
0x47	0	1	0	0 [0x0000] [PAT]	0 (not scrambled)	1	0

Figure 12-6.

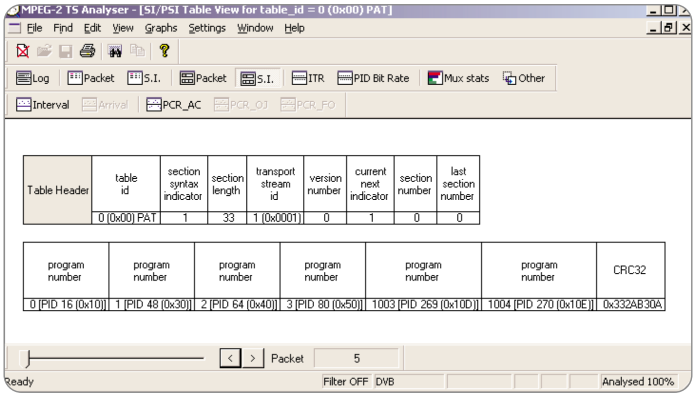

MPEG-2 TS Analyser - [SI/PSI Table View for table_id = 0 (0x00) PAT]
File Find Edit View Graphs Settings Window Help
Log Packet S.I. Packet S.I. ITR PID Bit Rate Mux stats Other
Interval Arrival PCR_AC PCR_OJ PCR_FO
Packet 5
Ready Filter OFF DVB Analysed 100%

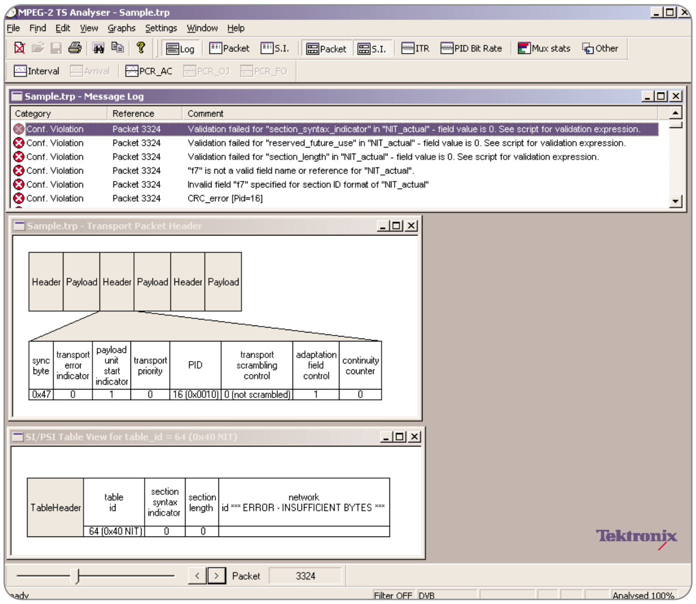
Table Header	table id	section syntax indicator	section length	transport stream id	version number	current next indicator	section number	last section number
	0 (0x00) PAT	1	33	1 (0x0001)	0	1	0	0

program number	program number	program number	program number	program number	program number	CRC32
0 [PID 16 (0x10)]	1 [PID 48 (0x30)]	2 [PID 64 (0x40)]	3 [PID 80 (0x50)]	1003 [PID 269 (0x10D)]	1004 [PID 270 (0x10E)]	0x332AB30A

Figure 12-7.

12.5 Syntax and CRC Analysis

To ship program material, the transport stream relies completely on the accurate use of syntax by encoders. Without correct settings of fixed flag bits, sync patterns, packet-start codes, and packet counts, a decoder may misinterpret the bit stream. The syntax check function considers all bits that are not program material and displays any discrepancies. Spurious discrepancies could be due to transmission errors; consistent discrepancies point to a faulty encoder or multiplexer. Figure 12-8 shows a syntax-error as well as a missing cyclic redundancy check (CRC).


MPEG-2 TS Analyser - Sample.trp
File Find Edit View Graphs Settings Window Help
Log Packet S.I. Packet S.I. ITR PID Bit Rate Mux stats Other
Interval Arrival PCR_AC PCR_OJ PCR_FO
Sample.trp - Message Log

Category	Reference	Comment
Conf. Violation	Packet 3324	Validation failed for "section_syntax_indicator" in "NIT_actual" - field value is 0. See script for validation expression.
Conf. Violation	Packet 3324	Validation failed for "reserved_future_use" in "NIT_actual" - field value is 0. See script for validation expression.
Conf. Violation	Packet 3324	Validation failed for "section_length" in "NIT_actual" - field value is 0. See script for validation expression.
Conf. Violation	Packet 3324	"f7" is not a valid field name or reference for "NIT_actual".
Conf. Violation	Packet 3324	Invalid field "f7" specified for section ID format of "NIT_actual"
Conf. Violation	Packet 3324	CRC_error [Pid=16]

sync byte	transport error indicator	payload unit start indicator	transport priority	PID	transport scrambling control	adaptation field control	continuity counter
0x47	0	1	0	16 (0x0010)	0 (not scrambled)	1	0

TableHeader	table id	section syntax indicator	section length	network id *** ERROR - INSUFFICIENT BYTES ***
	64 (0x40 NIT)	0	0	

Figure 12-8.

Many MPEG tables have checksums or CRCs attached for error detection. The analyzer can recalculate the checksums and compare them with the actual checksum. Again, spurious CRC mismatches could be due to stream-bit errors, but consistent CRC errors point to a hardware fault.

12.6 Filtering

A transport stream contains a great amount of data, and in real fault conditions, it is probable that, unless a serious problem exists, much of the data is valid and that perhaps only one elementary stream or one program is affected. In this case, it is more effective to test selectively, which is the function of filtering.

Essentially, filtering allows the user of an analyzer to be more selective when examining a transport stream. Instead of accepting every bit, the user can analyze only those parts of the data that meet certain conditions.

One condition results from filtering packet headers so that only packets with a given PID are analyzed. This approach makes it very easy to check the PAT by selecting PID 0, and, from there, all other PIDs can be read out. If the PIDs of a suspect stream are known, perhaps from viewing a hierarchical display, it is easy to select a single PID for analysis.

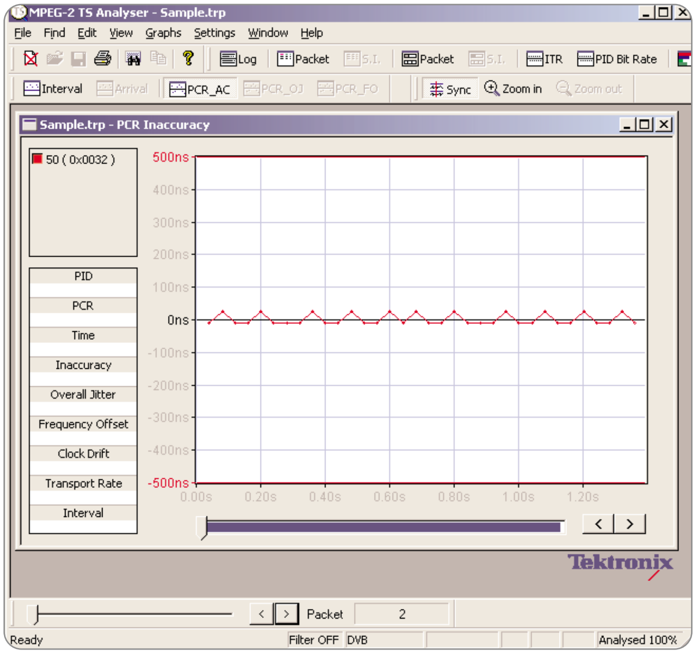


Figure 12-9.

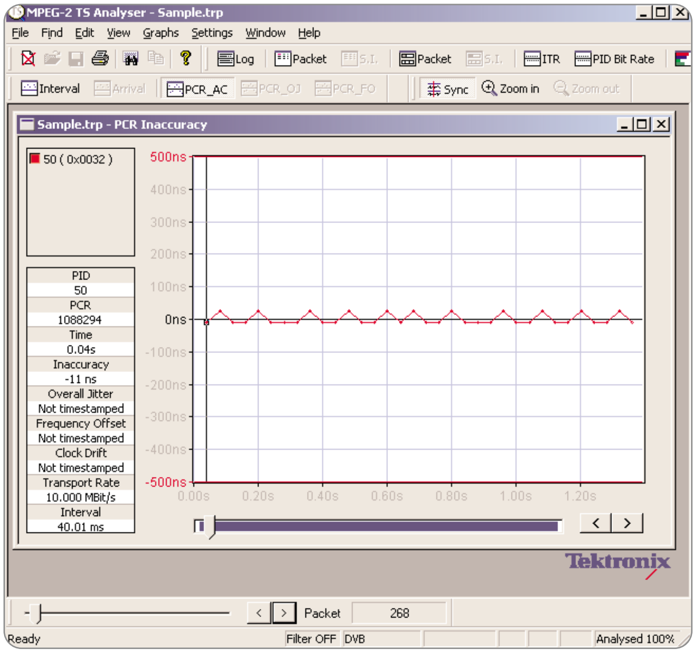
12.7 Timing Analysis

The tests described above check for the presence of the correct elements and syntax in the transport stream. However, to display real-time audio and video correctly, the transport stream must also deliver accurate timing to the decoders. This task can be confirmed by analyzing the PCR and time-stamp data.

The correct transfer of program-clock data is vital because this data controls the entire timing of the decoding process. PCR analysis can show that, in each program, PCR data is sent at a sufficient rate and with sufficient accuracy to be compliant.

The PCR data from a multiplexer may be precise, but remultiplexing may put the packets of a given program at a different place on the time axis, requiring that the PCR data be edited by the remultiplexer. Consequently, it is important to test for PCR inaccuracies after the data is remultiplexed.

Figure 12-9 shows a PCR display that indicates the positions at which PCRs were received with respect to an average clock. At the next display level, each PCR can be opened to display the PCR data, as is shown in Figure 12-10. To measure inaccuracies, the analyzer predicts the PCR value by using the previous PCR and the bit rate to produce what is called the interpolated PCR. The actual PCR value is subtracted from the estimated PCR to give an estimate of the inaccuracy.



MPEG-2 TS Analyser - Sample.trp

Sample.trp - PCR Inaccuracy

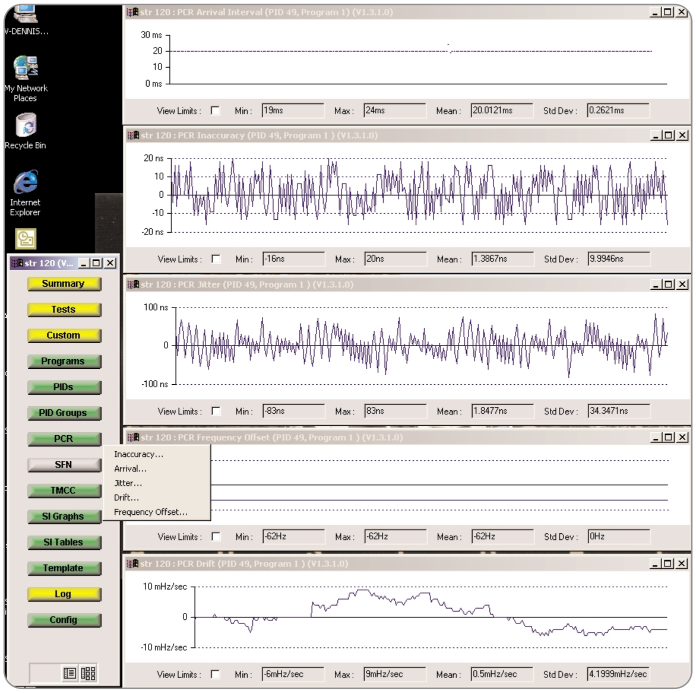
50 (0x0032)

PID
50
PCR
1088294
Time
0.04s
Inaccuracy
-11 ns
Overall Jitter
Not timestamped
Frequency Offset
Not timestamped
Clock Drift
Not timestamped
Transport Rate
10.000 MBit/s
Interval
40.01 ms

Packet 268

Ready | Filter OFF | DVB | Analysed 100%

Figure 12-10.



str 120 : PCR Arrival Interval (PID 49, Program 1) (V1.3.1.0)
View Limits : ☐ Min : 19ms Max : 24ms Mean : 20.0121ms Std Dev : 0.2621ms

str 120 : PCR Inaccuracy (PID 49, Program 1) (V1.3.1.0)
View Limits : ☐ Min : -16ns Max : 20ns Mean : 1.3867ns Std Dev : 9.9946ns

str 120 : PCR Jitter (PID 49, Program 1) (V1.3.1.0)
View Limits : ☐ Min : -83ns Max : 83ns Mean : 1.8477ns Std Dev : 34.3471ns

str 120 : PCR Frequency Offset (PID 49, Program 1) (V1.3.1.0)
View Limits : ☐ Min : -62Hz Max : -62Hz Mean : -62Hz Std Dev : 0Hz

str 120 : PCR Drift (PID 49, Program 1) (V1.3.1.0)
View Limits : ☐ Min : -6mHz/sec Max : 9mHz/sec Mean : 0.5mHz/sec Std Dev : 4.1999mHz/sec

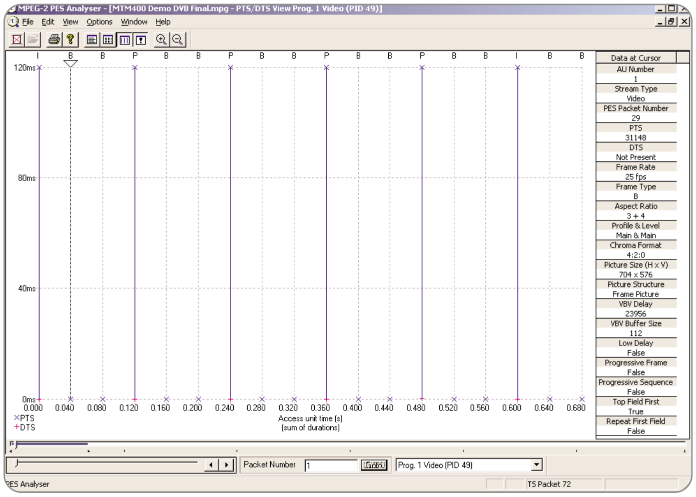
Figure 12-11.

An alternate approach shown in Figure 12-11 provides a graphical display of PCR interval, inaccuracy, jitter, frequency offset and drift, which is updated in real-time.

Figure 12-12 shows a time-stamp display for a selected elementary stream. The access unit, the presentation time, and, where appropriate, the decode times are all shown.

In MPEG, the reordering and use of different picture types causes delay and requires buffering at both encoder and decoder. A given elementary stream must be encoded within the constraints of the availability of buffering at the decoder. MPEG defines a model decoder called the T-STD (transport stream system target decoder); an encoder or multiplexer must not distort the data flow beyond the buffering ability of the T-STD. The transport stream contains parameters called VBV (video buffer verify) specifying the amount of buffering needed by a given elementary stream.

The T-STD analysis displays the buffer occupancy graphically so that overflows or underflows can be easily seen. Figure 12-13 shows a buffering display.



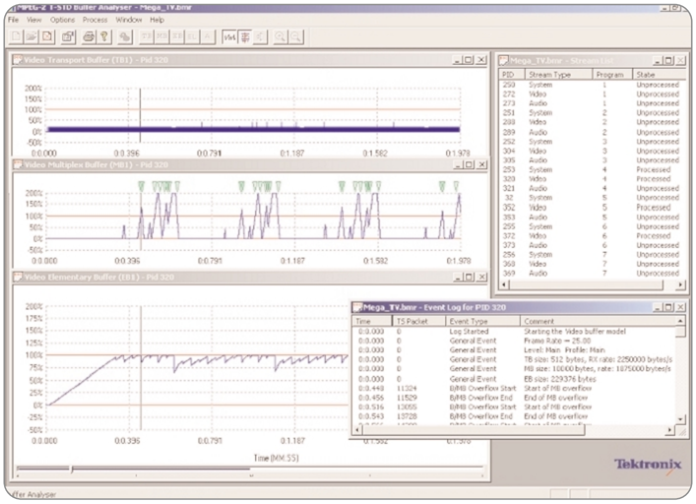
MPEG-2 PES Analyser - [MTM400 Demo DVB Final.mpg - PTS/DTS View Prog. 1 Video (PID 49)]

Data at Cursor
AU Number
1
Stream Type
Video
PES Packet Number
29
PTS
31148
DTS
Not Present
Frame Rate
25 fps
Frame Type
B
Aspect Ratio
3 + 4
Profile & Level
Main & Main
Chroma Format
4:2:0
Picture Size (H x V)
704 x 576
Picture Structure
Frame Picture
VBV Delay
23956
VBV Buffer Size
112
Low Delay
False
Progressive Frame
False
Progressive Sequence
False
Top Field First
True
Repeat First Field
False

Packet Number 1 | Prog. 1 Video (PID 49)

PES Analyser | TS Packet 72

Figure 12-12.



MPEG-2 T-STD Buffer Analyser - Mega_TV.bmr

Mega_TV.bmr - Stream List

PID	Stream Type	Program	State
250	System	1	Unprocessed
272	Video	1	Unprocessed
273	Audio	1	Unprocessed
251	System	2	Unprocessed
288	Video	2	Unprocessed
289	Audio	2	Unprocessed
252	System	3	Unprocessed
304	Video	3	Unprocessed
305	Audio	3	Unprocessed
253	System	4	Processed
320	Video	4	Processed
321	Audio	4	Unprocessed
32	System	5	Unprocessed
352	Video	5	Processed
353	Audio	5	Unprocessed
255	System	6	Unprocessed
372	Video	6	Processed
373	Audio	6	Unprocessed
256	System	7	Unprocessed
368	Video	7	Unprocessed
369	Audio	7	Unprocessed

Mega_TV.bmr - Event Log for PID 320

Time	TS Packet	Event Type	Comment
0:0.000	0	Log Started	Starting the Video buffer model
0:0.000	0	General Event	Frame Rate = 25.00
0:0.000	0	General Event	Level: Main Profile: Main
0:0.000	0	General Event	TB size: 512 bytes, RX rate: 2250000 bytes/s
0:0.000	0	General Event	MB size: 10000 bytes, rate: 1875000 bytes/s
0:0.000	0	General Event	EB size: 229376 bytes
0:0.448	11324	B/MB Overflow Start	Start of MB overflow
0:0.456	11529	B/MB Overflow End	End of MB overflow
0:0.516	13055	B/MB Overflow Start	Start of MB overflow
0:0.543	13728	B/MB Overflow End	End of MB overflow

Figure 12-13.

The output of a normal compressor/multiplexer is of limited use because it is not deterministic. If a decoder defect is seen, there is no guarantee that the same defect will be seen on a repeat of the test because the same video signal will not result in the same transport stream. In this case, an absolutely repeatable transport stream is essential so that the defect can be made to occur at will for study or rectification.

Transport stream jitter should be within certain limits, but a well-designed decoder should be able to recover programs beyond this limit in order to guarantee reliable operation. There is no way to test for this capability using existing transport streams because, if they are compliant, the decoder is not being tested. If there is a failure, it will not be reproducible and it may not be clear whether the failure was due to jitter or some other noncompliance. The solution is to generate a transport stream that is compliant in every respect and then add a controlled amount of inaccuracy to it so that the inaccuracy is then known to be the only source of noncompliance. The editor feature of the AD953 is designed to create such signals.

12.8 Elementary Stream Testing

Because of the flexible nature of the MPEG bit stream, the number of possibilities and combinations it can contain is almost incalculable. As the encoder is not defined, encoder manufacturers are not compelled to use every possibility; indeed, for economic reasons, this is unlikely. This fact makes testing quite difficult because the fact that a decoder works with a particular encoder does not prove compliance. That decoder may simply not be using the modes that cause the decoder to fail.

A further complication occurs because encoders are not deterministic and will not produce the same bit stream if the video or audio input is repeated. There is little chance that the same alignment will exist between I-, P- and B-pictures and the video frames. If a decoder fails a given test, it may not fail the next time the test is run, making fault-finding difficult. A failure with a given encoder does not determine whether the fault lies with the encoder or the decoder. The coding difficulty depends heavily on the nature of the program material, and any given program material will not necessarily exercise every parameter over the whole coding range.

To make tests that have meaningful results, two tools are required:

- A known source of compliant test signals that deliberately explore the whole coding range. These signals must be deterministic so that a decoder failure will give repeatable symptoms. The Sarnoff compliant bit streams are designed to perform this task.
- An elementary stream analyzer that allows the entire syntax from an encoder to be checked for compliance.

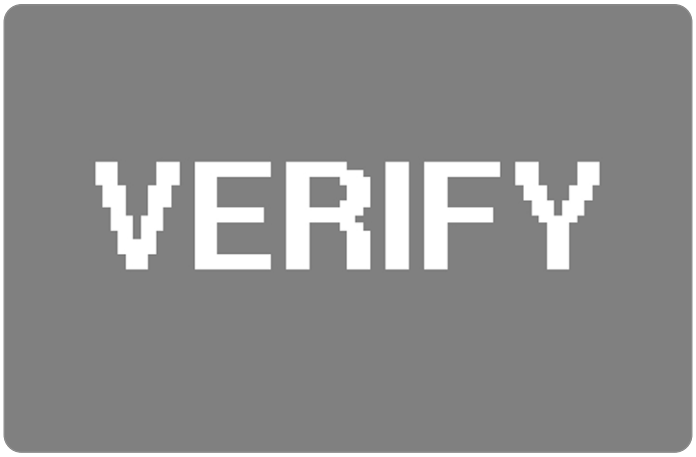

VERIFY

Figure 12-14.

12.9 Sarnoff® Compliant Bit Streams

These bit streams have been specifically designed by The Sarnoff® Corporation for decoder compliance testing. They can be multiplexed into a transport stream feeding a decoder.

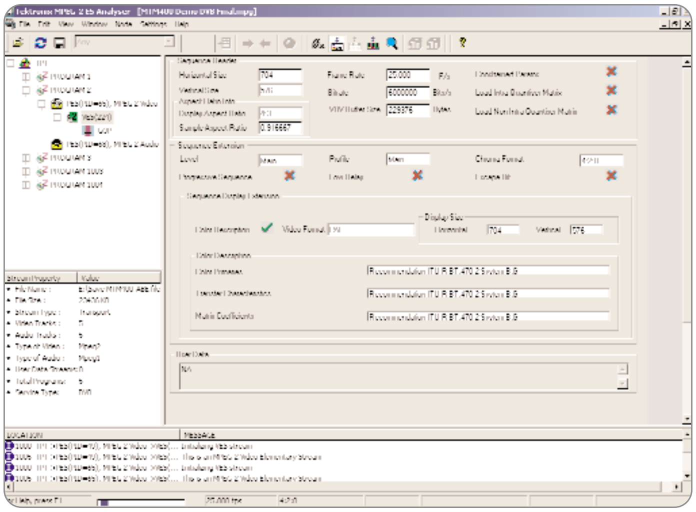
No access to the internal working of the decoder is required. To avoid the need for lengthy analysis of the decoder output, the bit streams have been designed to create a plain picture when they complete so that it is only necessary to connect a picture monitor to the decoder output to view them.

There are a number of these simple pictures. Figure 12-14 shows the gray verify screen. The user should examine the verify screen to look for discrepancies that will display well against the gray field. There are also some verify pictures which are not gray.

Some tests will result in no picture at all if there is a failure. These tests display the word “VERIFY” on screen when they complete.

Further tests require the viewer to check for smooth motion of a moving element across the picture. Timing or ordering problems will cause visible jitter.

The suite of Sarnoff tests may be used to check all of the MPEG syntax elements in turn. In one test, the bit stream begins with I-pictures only, adds P-pictures, and then adds B-pictures to test whether all MPEG picture types can be handled and correctly reordered. Backward compatibility with MPEG-1 can be proven. Another bit-stream tests using a range of different GOP structures. There are tests that check the operation of motion vectors over the whole range of values, and there are tests that vary the size of slices or the amount of stuffing.



Sequence Header

Horizontal Size	704	Frame Rate	25.000	F/s	Constrained Params	☒
Vertical Size	576	Bitrate	6000000	Bits/s	Load Intra Quantiser Matrix	☒
Display Aspect Ratio	4:3	VBV Buffer Size	229376	Bytes	Load Non Intra Quantiser Matrix	☒
Sample Aspect Ratio	0.916667					

Sequence Extension

Level	Main	Profile	Main	Chroma Format	4:2:0
Progressive Sequence	☒	Low Delay	☒	Escape Bit	☒

Sequence Display Extension

Color Description ✓ Video Format PAL

Display Size: Horizontal 704 Vertical 576

Color Description

Color Primaries	Recommendation ITU R BT.470 2 System B,G
Transfer Characteristics	Recommendation ITU R BT.470 2 System B,G
Matrix Coefficients	Recommendation ITU R BT.470 2 System B,G

User Data

NA

Figure 12-15.

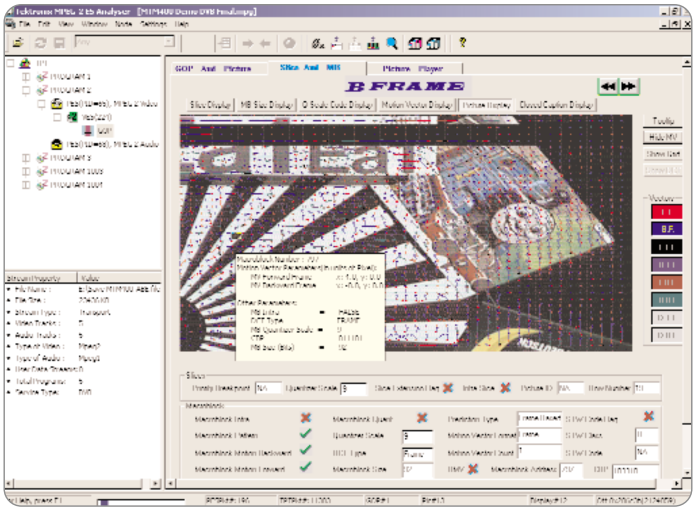


Figure 12-16.

In addition to providing decoder tests, the Sarnoff streams also include sequences that cause a good decoder to produce standard video test signals to check DACs (digital-to-analog converter), signal levels, and composite or Y/C encoders. These sequences turn the decoder into a video test-pattern generator capable of producing conventional video signals such as zone plates, ramps and color bars.

12.10 Elementary Stream Analysis

An elementary stream is a payload that the transport stream must deliver transparently. The transport stream will do so whether or not the elementary stream is compliant. In other words, testing a transport stream for compliance simply means checking that it is delivering elementary streams unchanged. It does not mean that the elementary streams were properly assembled in the first place.

The elementary stream structure or syntax is the responsibility of the compressor. Therefore an elementary stream test is essentially a form of compressor test. It should be noted that a compressor can produce compliant syntax, and yet still have poor audio or video quality. However, if the syntax is incorrect, a decoder may not be able to interpret the elementary stream. Since compressors are algorithmic rather than deterministic, an elementary stream may be intermittently noncompliant if some less common mode of operation is not properly implemented.

As transport streams often contain several programs that come from different coders, elementary stream problems tend to be restricted to one program, whereas transport stream problems tend to affect all programs. If problems are noted with the output of a particular decoder, then the Sarnoff compliance tests should be run on that decoder. If these are satisfactory, the fault may lie in the input signal. If the transport stream syntax has been tested, or if other programs are working without fault, then an elementary stream analysis is justified.

Elementary stream analysis can begin at the top level of the syntax and continue downwards. Sequence headers are very important as they tell the decoder all of the relevant modes and parameters used in the compression. The elementary stream syntax described in Sections 5.1 and 5.2 should be used as a guide. Figure 12-15 shows a sequence header and extension displayed on an AD953. At a lower level of testing, Figure 12-16 shows a decoded B frame along with the motion vectors overlayed to the picture.

12.11 Creating a Transport Stream

Whenever the decoder is suspect, it is useful to be able to generate a test signal of known quality. Figure 12-17 shows that an MPEG transport stream must include Program Specific Information (PSI), such as PAT, PMT, and NIT describing one or more program streams. Each program stream must contain its own PCR and elementary streams having periodic time stamps.

A DVB transport stream will contain additional service information, such as BAT, SDT and EIT tables. A PSI/SI editor enables insertion of any desired compliant combination of PSI/SI into a custom test stream.

Clearly, each item requires a share of the available transport-stream rate. The multiplexer provides a rate gauge to display the total bit rate used. The remainder of the bit rate is used up by inserting stuffing packets with PIDs that contain all 1s, which a decoder will reject.

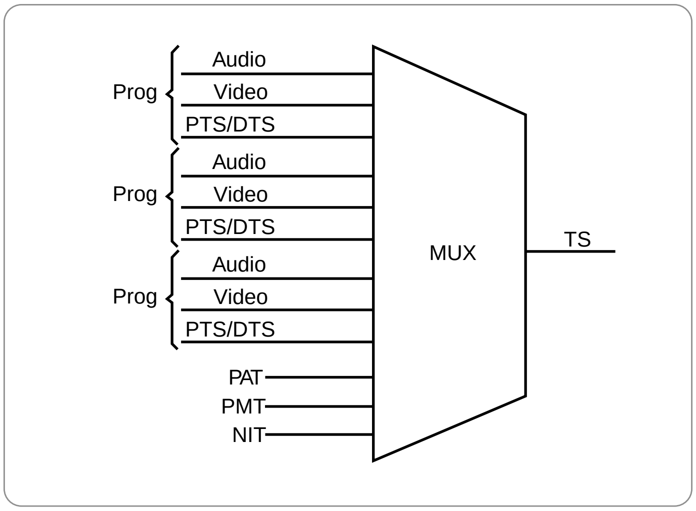


Figure 12-17.

12.12 PCR Inaccuracy Generation

The MPEG decoder has to recreate a continuous clock by using the clock samples in PCR data to drive a phase-locked loop. The loop needs filtering and damping so that jitter in the time of arrival of PCR data does not cause instability in the clock.

To test the phase-locked loop performance, a signal with known inaccuracy is required; otherwise, the test is meaningless. The AD953 can generate simulated inaccuracies for this purpose. Because it is a reference generator, the AD953 has highly stable clock circuits and the actual output jitter is very small. To create the effect of jitter, the timing of the PCR data is not changed at all. Instead, the PCR values are modified so that the PCR count they contain is slightly different from the ideal. The modified value results in phase errors at the decoder that are indistinguishable from real jitter.

The advantage of this approach is that jitter of any required magnitude can easily be added to any program stream simply by modifying the PCR data and leaving all other data intact. Other program streams in the transport stream need not have jitter added. In fact, it may be best to have a stable program stream to use as a reference.

For different test purposes, the time base may be modulated in a number of ways that determine the spectrum of the loop phase error in order to test the loop filtering. Square-wave jitter alternates between values which are equally early or late. Sinusoidal jitter values cause the phase error to be a sampled sine wave. Random jitter causes the phase error to be similar to noise.

Glossary

AAC – Advanced Audio Coding.

AAU – Audio Access Unit. *See Access unit.*

AC-3 – The audio compression scheme invented by Dolby Laboratories and specified for the ATSC Digital Television Standard. In the world of consumer equipment it is called Dolby Digital.

Access Unit – The coded data for a picture or block of sound and any stuffing (null values) that follows it.

A/D – Analog to digital converter.

AES – Audio Engineering Society.

Anchor Frame – A video frame that is used for prediction. I frames and P frames are generally used as anchor frames, but B frames are never anchor frames.

ANSI – American National Standards Institute.

API – Application Program Interface.

ARIB – Association of Radio Industries and Businesses.

Asynchronous Transfer Mode (ATM) – A digital signal protocol for efficient transport of both constant-rate and bursty information in broadband digital networks. The ATM digital stream consists of fixed-length packets called "cells," each containing 53 8-bit bytes – a 5-byte header and a 48-byte information payload.

ATM – *See asynchronous transfer mode.*

ATSC – Advanced Television Systems Committee.

ATVEF – Advanced Television Enhancement Forum.

AU – Access Unit.

BAT – Bouquet Association Table.

BER – Bit Error Rate.

BFSK – Binary Frequency Shift Keying.

BIOP – Broadcast Inter-ORB Protocol.

Bit rate – The rate at which the compressed bit stream is delivered from the channel to the input of a decoder.

Block – A block is an array of pixel values or DCT coefficients, usually 8-by-8 (8x8), representing luminance or chrominance information.

Bouquet – A group of transport streams in which programs are identified by combination of network ID and PID (part of DVB-SI).

BPSK – Binary Phase Shift Keying.

CA – Conditional Access. Information indicating whether a program is scrambled.

CAT – Conditional Access Table. Packets having PID *(see Section 8 – Transport Streams)* codes of 1 and that contain information about the scrambling system. *See ECM and EMM.*

CD – Compact disc.

CELP – Code Excited Linear Predictive.

Channel Code – A modulation technique that converts raw data into a signal that can be recorded or transmitted by radio or cable.

CIF – Common Interchange Format. A 352x240 pixel format for 30 fps video conferencing.

Closed GOP – A Group of Pictures in which the last pictures do not need data from the next GOP for bidirectional coding. Closed GOP is used to make a splice point in a bit stream.

Coefficient – A number specifying the amplitude of a particular frequency or basis function in a transform.

CORBA – Common Object Request Broker Architecture.

COFDM – Coded Orthogonal Frequency Division Multiplex, a modified form of OFDM. A digital modulation scheme using a very large number of carriers, each carrying a very low data rate. Used by DVB-T.

Compression – Reduction in the number of bits used to represent an item of data.

CRC – Cyclic Redundancy Check.

DAC – Digital-to-Analog Converter.

DASE – DigitalTV Application Software Environment.

DAVIC – Digital Audio Visual Council.

DCT – Discrete Cosine Transform.

DDB – DownloadDataBlock.

DET – Data Event Table.

DFT – Discrete Fourier Transform.

DII – DownloadInfoIndication.

Dolby Digital – *See AC-3.*

DSI – DownloadServerInitiate.

DSMCC – Digital Storage Media Command and Control.

DST – Data Services Table.

DTS – Decoding Time Stamp. Part of PES header indicating when an access unit is to be decoded.

DVB – Digital Video Broadcasting. Generally refers to the European-initiated consortium of broadcasters, manufacturers, regulatory bodies and others that created standards for the delivery of digital television and data services. Includes DVB-C (cable), DVB-S (satellite) and DVB-T (terrestrial) versions.

DVB-SI – DVB Service Information. Information carried in a DVB multiplex describing the contents of different multiplexes. Includes NIT, SDT, EIT, TDT, BAT, RST and ST *(see Section 10 – Introduction to DVB & ATSC).*

DVC – Digital Video Cassette.

DVD – Digital Versatile Disk or Digital Video Disk.

Elementary Stream – The raw output of a compressor carrying a single video or audio signal.

ECM – Entitlement Control Message. Conditional access information specifying control words or other stream-specific scrambling parameters.

ECS – Enhanced Content Specification.

EIT – Event Information Table. Part of DVB-SI.

EMM – Entitlement Management Message. Conditional access information specifying authorization level or services of specific decoders. An individual decoder or a group of decoders may be addressed.

ENG – Electronic News Gathering. Term used to describe use of video-recording instead of film in news coverage.

Entropy Coding – Variable length lossless coding of the digital representation of a signal to reduce redundancy.

EOB – End of Block.

EPG – Electronic Program Guide. A program guide delivered by data transfer rather than printed paper.

ETSI – European Telecommunication Standard Institute.

FEC – Forward Error Correction. System in which redundancy is added to the message so that errors can be corrected dynamically at the receiver.

FGS – Fine Grain Scalability.

GOP – Group of Pictures. In transmission order a GOP starts with an I-picture and ends with the last picture before the next I-picture.

HAVI – Home Audio Video Interoperability.

Huffman coding – A type of source coding that uses codes of different lengths to represent symbols which have unequal likelihood of occurrence.

IEC – International Electrotechnical Commission.

Inter-coding – Compression that uses redundancy between successive pictures; also known as temporal coding.

Interleaving – A technique used with error correction that breaks up burst errors into many smaller errors.

Intra-coding – Compression that works entirely within one picture; also known as spatial coding.

IOR – Inter-operable Object Reference.

IP – Internet Protocol.

I-pictures – Intra-coded Pictures.

IRD – Integrated Receiver Decoder. A combined RF receiver and MPEG decoder that is used to adapt a TV set to digital transmissions.

ISDB – Integrated Services Data Broadcasting, the digital broadcasting system developed in Japan.

ISO – International Organization for Standardization.

ITU – International Telecommunication Union.

JPEG – Joint Photographic Experts Group.

JTC1 – Joint Technical Committee of the IEC.

JVT – Joint Video Team.

Level – The size of the input picture in use with a given profile *(see Section 2 – Compression in Video).*

MAC – Media Access Control.

Macroblock – The screen area represented by several luminance and color-difference DCT blocks that are all steered by one motion vector.

Masking – A psycho-acoustic phenomenon whereby certain sounds cannot be heard in the presence of others.

MDCT – Modified Discreet Cosine Transform.

MGT – Master Guide Table.

MHP – Multimedia Home Platform.

Motion Vector – A pair of numbers which represent the vertical and horizontal displacement of a region of a reference picture for prediction.

MP@HL – Main Profile at High Level.

MP@LL – Main Profile at Low Level.

MP@ML – Main Profile at Main Level.

MPE – Multi-protocol Encapsulation.

MPEG – Moving Picture Experts Group ISO/IEC JTC1/SC29/WG11, and the Standards developed by this Group.

MPEG-LA – MPEG License Agreements.

NIT – Network Information Table. Information in one transport stream that describes many transport streams.

NPT – Normal Play Time.

NRT – Network Resources Table.

Null Packets – Packets of “stuffing” that carry no data but are necessary to maintain a constant bit rate with a variable payload. Null packets always have a PID of 8191 (all ones). *(See Section 8 – Transport Streams.)*

OCAP – Open Cable Applications Platform.

OFDM – Orthogonal Frequency Division Multiplexing.

ORB – Object Request Brokerage.

PAL – Phase Alternate Line.

PAT – Program Association Table. Data appearing in packets having PID *(see Section 8 – Transport Streams)* code of zero that the MPEG decoder uses to determine which programs exist in a Transport Stream. PAT points to PMT, which, in turn, points to the video, audio and data content of each program.

PCM – Pulse Code Modulation. A technical term for an analog source waveform, for example, audio or video signals, expressed as periodic, numerical samples. PCM is an uncompressed digital signal.

PCR – Program Clock Reference. The sample of the encoder clock count that is sent in the program header to synchronize the decoder clock.

PES – Packetized Elementary Stream.

PID – Program Identifier. A 13-bit code in the transport packet header. PID 0 indicates that the packet contains a PAT PID. *(See Section 8 – Transport Streams.)* PID 1 indicates a packet that contains CAT. The PID 8191 (all ones) indicates null (stuffing) packets. All packets belonging to the same elementary stream have the same PID.

PMT – Program Map Tables. The tables in PAT that point to video, audio and data content of a transport stream.

Packets – A term used in two contexts: in program streams, a packet is a unit that contains one or more presentation units; in transport streams, a packet is a small, fixed-size data quantum.

Pixel – Picture element (sometimes pel). The smallest unit of an image, represented by one sample, or a set of samples such as GBR or YC_rC_b.

Preprocessing – The video signal processing that occurs before MPEG Encoding. Noise reduction, downsampling, cut-edit identification and 3:2 pulldown identification are examples of preprocessing.

Profile – Specifies the coding syntax used.

Program Stream – A bit stream containing compressed video, audio and timing information.

PS – Program Stream.

PSI – Program Specific Information. Information that keeps track of the different programs in an MPEG transport stream and in the elementary streams in each program. PSI includes PAT, PMT, NIT, CAT, ECM and EMM.

PSI/SI – A general term for combined MPEG PSI and DVB-SI.

PSIP – Program and System Information Protocol.

PTS – Presentation Time Stamp. The time at which a presentation unit is to be available to the viewer.

PU – Presentation Unit. One compressed picture or block of audio.

QAM – Quadrature Amplitude Modulation, a digital modulation system.

QCIF – One-quarter-resolution (176x144 pixels) Common Interchange Format. *See CIF.*

QMF – Quadrature Mirror Filter.

QPSK – Quaternary Phase Shift Keying (also known as Quadrature Phase Shift Keying), a digital modulation system particularly suitable for satellite links.

QSIF – One-quarter-resolution Source Input Format. *See SIF.*

Quantization – a processing step that approximates signal values by allocating one of a number of pre-defined values.

RLC – Run Length Coding. A coding scheme that counts number of similar bits instead of sending them individually.

RRT – Rating Region Table.

RST – Running Status Table.

R-S – Reed-Solomon is a polynomial function used by DVB for protecting up to 8 bytes within each transport packet.

SAOL – Structured Audio Orchestra Language.

Scalability – A characteristic of MPEG-2 that provides for multiple quality levels by providing layers of video data. Multiple layers of data allow a complex decoder to produce a better picture by using more layers of data, while a more simple decoder can still produce a picture using only the first layer of data.

SDI – Serial Digital Interface. Serial coaxial cable interface standard intended for production digital video signals.

SDK – Software Development Kit.

SDT – Service Description Table. A table listing the providers of each service in a transport stream.

SDTV – Standard Definition Television.

SI – *See DVB-SI.*

SIF – Source Input Format. A half-resolution input signal used by MPEG-1.

Slice – A sequence of consecutive macroblocks.

SMPTE – Society of Motion Picture and Television Engineers.

SNR – Signal-to-Noise Ratio.

SP@ML – Simple Profile at Main Level.

SPTS – Single Program Transport Stream.

ST – Stuffing Table.

STB – Set Top Box.

STC – System Time Clock. The common clock used to encode video and audio in the same program.

STT – System Time Table.

Stuffing – Meaningless data added to maintain constant bit rate.

Syndrome – Initial result of an error checking calculation. Generally, if the syndrome is zero, there is assumed to be no error.

TCP/IP – Transmission Control Protocol/Internet Protocol.

TDAC – Time Domain Aliasing Cancellation. A coding technique used in AC-3 audio compression.

TDT – Time and Date Table. Used in DVB-SI.

TOT – Time Offset Table.

Transport Stream (TS) – A multiplex of several program streams that are carried in packets. Demultiplexing is achieved by different packet IDs (PIDs). *See PSI, PAT, PMT and PCR.*

Truncation – Shortening the wordlength of a sample or coefficient by removing low-order bits.

T-STD – Transport Stream System Target Decoder. A decoder having a certain amount of buffer memory assumed to be present by an encoder.

TVCT – Terrestrial Virtual Channel Table.

VAU – Video Access Unit. One compressed picture in program stream.

VBV – Video Buffer Verify.

VCO – Voltage Controlled Oscillator.

VLC – Variable Length Coding. A compressed technique that allocates short codes to frequency values and long codes to infrequent values.

VOD – Video On Demand. A system in which television programs or movies are transmitted to a single consumer only when requested.

VSB – Vestigial Sideband Modulation. A digital modulation system used by ATSC.

Wavelet – A transform using a basis function that is not of fixed length but that grows longer as frequency reduces.

Weighting – A method of changing the distribution of the noise that is due to truncation by pre-multiplying values.

Y/C – Luminance and chrominance.

tektronix.com/video_audio

For Further Information
Tektronix maintains a comprehensive, constantly expanding collection of application notes, technical briefs and other resources to help engineers working on the cutting edge of technology. Please visit **www.tektronix.com**

Contact Tektronix:

ASEAN / Australasia / Pakistan	(65) 6356 3900
Austria	+43 2236 8092 262
Belgium	+32 (2) 715 89 70
Brazil & South America	55 (11) 3741-8360
Canada	1 (800) 661-5625
Central Europe & Greece	+43 2236 8092 301
Denmark	+45 44 850 700
Finland	+358 (9) 4783 400
France & North Africa	+33 (0) 1 69 86 80 34
Germany	+49 (221) 94 77 400
Hong Kong	(852) 2585-6688
India	(91) 80-2275577
Italy	+39 (02) 25086 1
Japan	81 (3) 3448-3010
Mexico, Central America & Caribbean	52 (55) 56666-333
The Netherlands	+31 (0) 23 569 5555
Norway	+47 22 07 07 00
People's Republic of China	86 (10) 6235 1230
Poland	+48 (0) 22 521 53 40
Republic of Korea	82 (2) 528-5299
Russia, CIS & The Baltics	+358 (9) 4783 400
South Africa	+27 11 254 8360
Spain	+34 (91) 372 6055
Sweden	+46 8 477 6503/4
Taiwan	886 (2) 2722-9622
United Kingdom & Eire	+44 (0) 1344 392400
USA	1 (800) 426-2200
USA (Export Sales)	1 (503) 627-1916

For other areas contact Tektronix, Inc. at: 1 (503) 627-7111

Updated 20 September 2002

11/02 HB/PG 25W-11418-4

Tektronix
Enabling Innovation