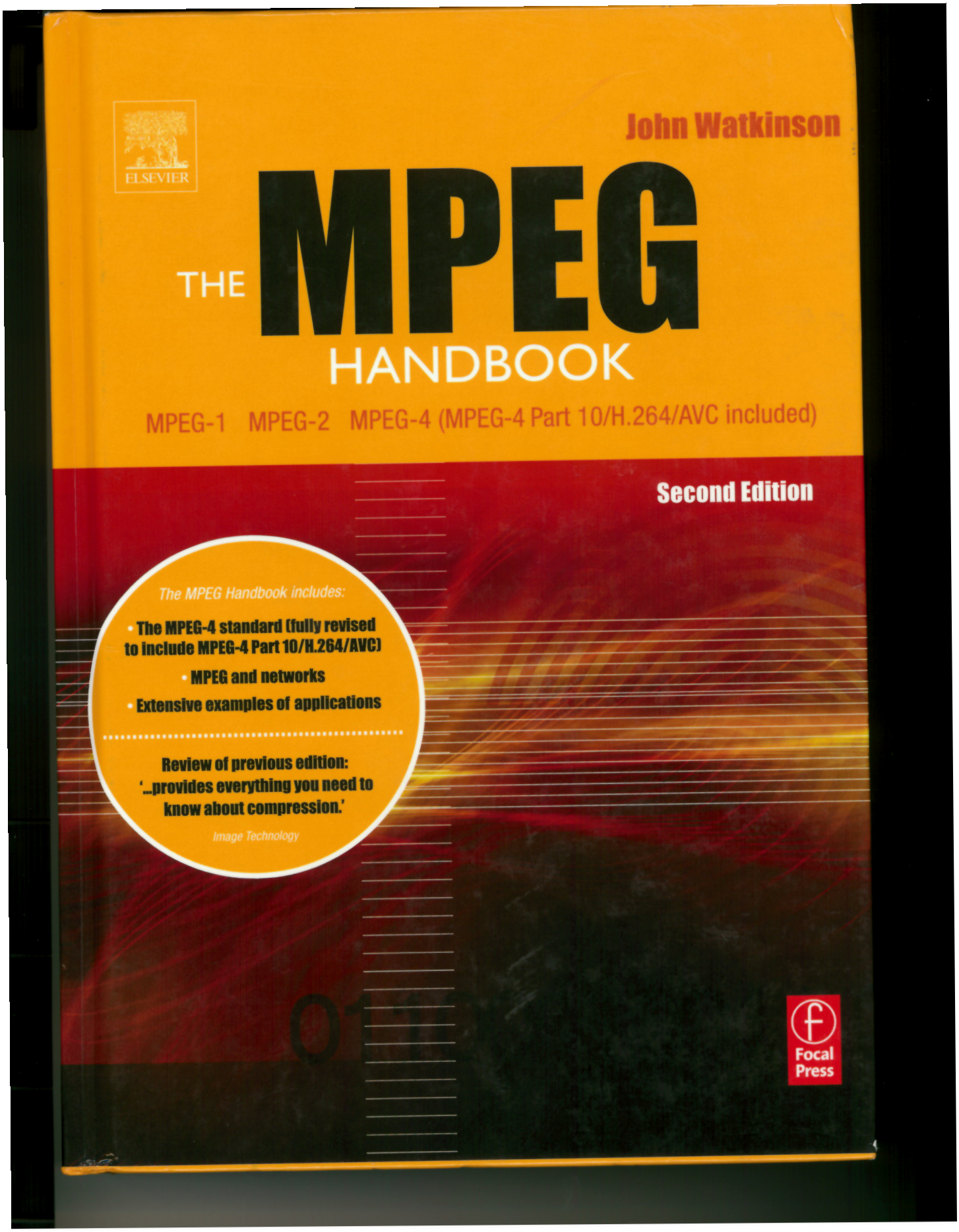

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 1 of 14

The MPEG Handbook

MPEG-1, MPEG-2, MPEG-4

Second edition

John Watkinson

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON • NEW YORK • OXFORD
PARIS • SAN DIEGO • SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Focal Press is an imprint of Elsevier

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338

Focal Press is an imprint of Elsevier
Linacre House, Jordan Hill, Oxford OX2 8DP, UK
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA

First edition 2001
Second edition 2004
Reprinted 2005, 2006

Copyright © 2001, 2004, John Watkinson. Published by Elsevier Ltd.
All rights reserved

The right of John Watkinson to be identified as the author of this work has been asserted in accordance with the Copyright, Designs and Patents Act 1988

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means electronic, mechanical, photocopying, recording or otherwise without the prior written permission of the publisher

Permissions may be sought directly from Elsevier's Science & Technology Rights Department in Oxford, UK: phone (+44) (0) 1865 843830; fax (+44) (0) 1865 853333; email: permissions@elsevier.com. Alternatively you can submit your request online by visiting the Elsevier web site at http://elsevier.com/locate/permissions, and selecting *Obtaining permission to use Elsevier material*

Notice
No responsibility is assumed by the publisher for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions or ideas contained in the material herein. Because of rapid advances in the medical sciences, in particular, independent verification of diagnoses and drug dosages should be made

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the Library of Congress

ISBN–13: 978-0-240-80578-8
ISBN–10: 0-240-80578-X

For information on all Focal Press publications visit our website at www.focalpress.com

Transferred to Digital Printing in 2010

Working together to grow libraries in developing countries
www.elsevier.com | www.bookaid.org | www.sabre.org
ELSEVIER BOOK AID International Sabre Foundation

Contents

Preface xi

Acknowledgements xii

Chapter 1 Introduction to compression 1

1.1 What is MPEG? 1
1.2 Why compression is necessary 3
1.3 MPEG-1, 2, 4 and H.264 contrasted 4
1.4 Some applications of compression 5
1.5 Lossless and perceptive coding 7
1.6 Compression principles 9
1.7 Video compression 14
1.7.1 Intra-coded compression 16
1.7.2 Inter-coded compression 16
1.7.3 Introduction to motion compensation 18
1.7.4 Film-originated video compression 19
1.8 Introduction to MPEG-1 21
1.9 MPEG-2: Profiles and Levels 21
1.10 Introduction to MPEG-4 24
1.11 Introduction to H.264 (AVC) 26
1.12 Audio compression 28
1.12.1 Sub-band coding 28
1.12.2 Transform coding 28
1.12.3 Predictive coding 29
1.13 MPEG bitstreams 29
1.14 Drawbacks of compression 30
1.15 Compression pre-processing 32
1.16 Some guidelines 32
References 33

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 4 of 14

Chapter 2 Fundamentals 35

2.1 What is an audio signal? 35
2.2 What is a video signal? 35
2.3 Types of video 36
2.4 What is a digital signal? 38
2.5 Sampling 41
2.6 Reconstruction 46
2.7 Aperture effect 50
2.8 Choice of audio sampling rate 52
2.9 Video sampling structures 56
2.10 The phase-locked loop 58
2.11 Quantizing 60
2.12 Quantizing error 61
2.13 Dither 65
2.14 Introduction to digital processing 67
2.15 Logic elements 69
2.16 Storage elements 71
2.17 Binary coding 73
2.18 Gain control 84
2.19 Floating-point coding 85
2.20 Multiplexing principles 87
2.21 Packets 88
2.22 Statistical multiplexing 88
2.23 Timebase correction 89
References 91

Chapter 3 Processing for compression 92

3.1 Introduction 92
3.2 Transforms 95
3.3 Convolution 97
3.4 FIR and IIR filters 97
3.5 FIR filters 100
3.6 Interpolation 107
3.7 Downsampling filters 116
3.8 The quadrature mirror filter 117
3.9 Filtering for video noise reduction 121
3.10 Warping 123
3.11 Transforms and duality 129
3.12 The Fourier transform 132
3.13 The discrete cosine transform (DCT) 138
3.14 The wavelet transform 140
3.15 The importance of motion compensation 147

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 5 of 14

3.16 Motion-estimation techniques 150
3.16.1 Block matching 150
3.16.2 Gradient matching 152
3.16.3 Phase correlation 153
3.17 Motion-compensated displays 158
3.18 Camera-shake compensation 159
3.19 Motion-compensated de-interlacing 161
3.20 Compression and requantizing 162
References 167

Chapter 4 Audio compression 169

4.1 Introduction 169
4.2 The deciBel 170
4.3 Audio level metering 175
4.4 The ear 177
4.5 The cochlea 180
4.6 Level and loudness 182
4.7 Frequency discrimination 183
4.8 Critical bands 184
4.9 Beats 187
4.10 Codec level calibration 188
4.11 Quality measurement 189
4.12 The limits 191
4.13 Compression applications 192
4.14 Audio compression tools 192
4.15 Sub-band coding 197
4.16 Audio compression formats 198
4.17 MPEG audio compression 199
4.18 MPEG Layer I audio coding 201
4.19 MPEG Layer II audio coding 206
4.20 MPEG Layer III audio coding 208
4.21 MPEG-2 AAC – advanced audio coding 212
4.22 Dolby AC-3 218
4.23 MPEG-4 audio 219
4.24 MPEG-4 AAC 220
4.25 Compression in stereo and surround sound 221
References 227

Chapter 5 MPEG video compression 230

5.1 The eye 230
5.2 Dynamic resolution 234
5.3 Contrast 238
5.4 Colour vision 240
5.5 Colour difference signals 240
5.6 Progressive or interlaced scan? 245

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 6 of 14

5.7	Spatial and temporal redundancy in MPEG	250
5.8	*I* and *P* coding	255
5.9	Bidirectional coding	256
5.10	Coding applications	259
5.11	Intra-coding	260
5.12	Intra-coding in MPEG-1 and MPEG-2	265
5.13	A bidirectional coder	270
5.14	Slices	274
5.15	Handling interlaced pictures	275
5.16	MPEG-1 and MPEG-2 coders	281
5.17	The elementary stream	282
5.18	An MPEG-2 decoder	283
5.19	MPEG-4 and AVC	286
5.20	Video objects	288
5.21	Texture coding	291
5.22	Shape coding	297
5.23	Padding	299
5.24	Video object coding	301
5.25	Two-dimensional mesh coding	303
5.26	Sprites	308
5.27	Wavelet-based compression	310
5.28	Three-dimensional mesh coding	315
5.29	Animation	324
5.30	Scaleability	326
5.31	Advanced Video Coding (AVC)	328
5.32	Motion compensation in AVC	335
5.33	An AVC codec	337
5.34	AVC profiles and levels	340
5.35	Coding artifacts	340
5.36	MPEG and concatenation	346
References		352

Chapter 6 MPEG bitstreams **353**

6.1	Introduction	353
6.2	Packets and time stamps	355
6.3	Transport streams	357
6.4	Clock references	359
6.5	Program Specific Information (PSI)	361
6.6	Multiplexing	362
6.7	Remultiplexing	364
Reference		365

Chapter 7 MPEG applications **366**

7.1	Introduction	366
7.2	Telephones	367

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 7 of 14

7.3 Digital television broadcasting 368
7.4 The DVB receiver 381
7.5 ATSC 383
7.6 CD-Video and DVD 389
7.7 Personal video recorders 394
7.8 Networks 397
7.9 FireWire 405
7.10 Broadband networks and ATM 407
7.11 ATM AALs 411
References 414

Index 415

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 8 of 14

5

MPEG video compression

In this chapter the principles of video compression are explored, leading to descriptions of MPEG-1, MPEG-2 MPEG-4. MPEG-4 Part 10, also known as H.264 or AVC, is also covered, and for simplicity it will be referred to throughout as AVC.

MPEG-1 supports only progressively scanned images, whereas MPEG-2 and MPEG-4 support both progressive and interlaced scan. MPEG uses the term 'picture' to mean a full-screen image of any kind at one point on the time axis. This could be a field or a frame in interlaced systems but only a frame in non-interlaced systems. The terms field and frame will be used only when the distinction is important. MPEG-4 introduces object coding that can handle entities that may not fill the screen. In MPEG-4 the picture becomes a plane in which one or more video objects can be displayed, hence the term video object plane (VOP).

5.1 The eye

All imaging signals ultimately excite some response in the eye and the viewer can only describe the result subjectively. Familiarity with the functioning and limitations of the eye is essential to an understanding of image compression. The simple representation of Figure 5.1 shows that the eyeball is nearly spherical and is swivelled by muscles. The space between the cornea and the lens is filled with transparent fluid known as *aqueous humour*. The remainder of the eyeball is filled with a transparent jelly known as *vitreous humour*. Light enters the cornea, and the amount of light admitted is controlled by the pupil in the iris. Light entering is involuntarily focused on the retina by the lens in a process called *visual accommodation*. The lens is the only part of the eye which is not nourished by the bloodstream and its centre is technically dead. In a young person

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 9 of 14

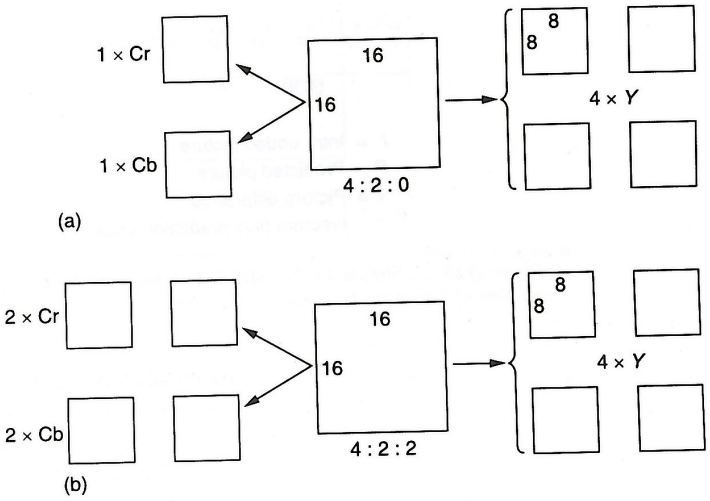

Figure 5.22 The structure of a macroblock. (A macroblock is the screen area steered by *one* vector.) (a) In 4:2:0, there are two chroma DCT blocks per macroblock whereas in 4:2:2 (b) there are four, 4:2:2 needs 33 per cent more data than 4:2:0.

8 × 8 DCT block rather than with a 16 × 16 macroblock. This allows much more precise prediction of complex motion, especially near the edge of objects, and will result in a reduced residual bit rate. On a macroblock-by-macroblock basis an MPEG-4 coder can decide whether to code four vectors per macroblock or only one. When the compressed vector data and residual are combined, one of these approaches will result in the least data. In AVC the vector density may be even higher still.

5.8 *I* and *P* coding

Predictive (*P*) coding cannot be used indefinitely, as it is prone to error propagation. A further problem is that it becomes impossible to decode the transmission if reception begins part-way through. In real video signals, cuts or edits can be present across which there is little redundancy and which make motion estimators throw up their hands.

In the absence of redundancy over a cut, there is nothing to be done but to send the new picture information in absolute form. This is called *I* coding where *I* is an abbreviation of *intra*-coding. As *I* coding needs no previous picture for decoding, then decoding and/or error recovery can begin at *I* coded information.

MPEG is effectively a toolkit. The bigger the MPEG-number, the more tools are available. However, there is no compulsion to use all the tools available. Thus an encoder may choose whether to use *I* or *P* coding,

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 10 of 14

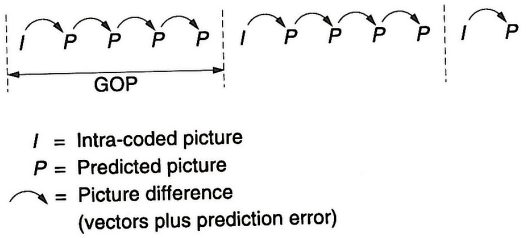

Figure 5.23 A Simple Profile MPEG-2 signal may contain periodic *I* pictures with a number of *P* pictures between.

either once and for all or dynamically on a macroblock-by-macroblock basis. For practical reasons, an entire frame should be encoded as *I* macroblocks periodically. This creates a place where the bitstream might be edited or where decoding could begin.

Figure 5.23 shows a typical application of the Simple Profile of MPEG-2. Periodically an *I* picture is created. Between *I* pictures are *P* pictures which are based on the picture before. These *P* pictures predominantly contain macroblocks having vectors and prediction errors. However, it is perfectly legal for *P* pictures to contain *I* macroblocks. This might be useful where, for example, a camera pan introduces new material at the edge of the screen which cannot be created from an earlier picture.

Note that although what is sent is called a *P* picture, it is not a picture at all. It is a set of instructions to convert the previous picture into the current picture. If the previous picture is lost, decoding is impossible. An *I* picture together with all of the pictures before the next *I* picture form a *group of pictures* (GOP). There is no requirement for GOPs to be of constant size and indeed the size may vary dynamically.

5.9 Bidirectional coding

Motion-compensated predictive coding is a useful compression technique, but it does have the drawback that it can only take data from a previous picture.

Where moving objects reveal a background this is completely unknown in previous pictures and forward prediction fails. However, more of the background is visible in later pictures. Figure 5.24 shows the concept. In the centre of the diagram, a moving object has revealed some background. The previous picture can contribute nothing, whereas the next picture contains all that is required.

Bidirectional coding is shown in Figure 5.25. A bidirectional or *B* macroblock can be created using a combination of motion compensation

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 11 of 14

the more bit rate is lost to synchronizing. Thus for recording it is advantageous to have a different approach in which larger packets are used to cut down synchronizing overhead.

6.2 Packets and time stamps

The video elementary stream is an endless bitstream representing pictures which are not necessarily in the correct order and which take a variable length of time to transmit. Storage and transmission systems prefer discrete blocks of data. In MPEG-2 elementary streams are packetized to form a PES (packetized elementary stream). Audio elementary streams are also packetized. A packet is shown in Figure 6.2. It begins with a header containing an unique packet start code and a code which identifies the type of data stream. Optionally the packet header also may contain one or more time stamps which are used for synchronizing the video decoder to real time and for obtaining lip-sync.

Figure 6.3 shows that a time stamp is a sample of the state of a counter which is driven by a 90 kHz clock. This is obtained by dividing down the master 27 MHz clock of MPEG-2. There are two types of time stamp: PTS and DTS. These are abbreviations for presentation time stamp and decode time stamp. A presentation time stamp determines when the associated picture should be displayed on the screen, whereas a decode

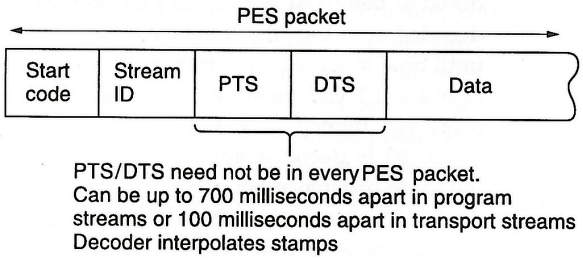

Figure 6.2 A PES packet structure is used to break up the continuous elementary stream.

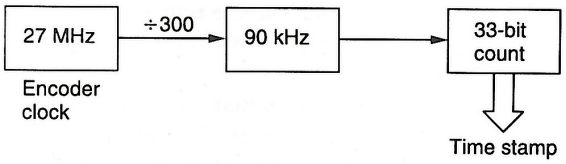

Figure 6.3 Time stamps are the result of sampling a counter driven by the encoder clock.

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 12 of 14

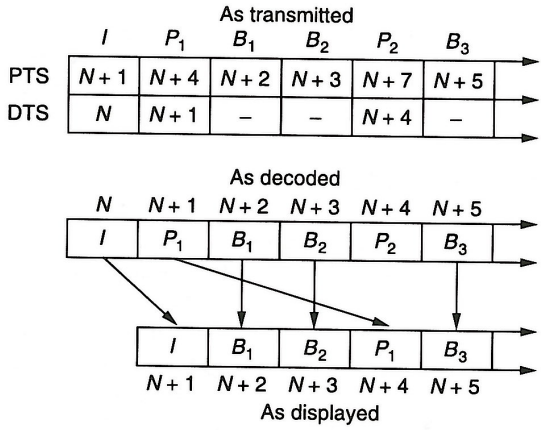

Figure 6.4 An example of using PTS/DTS to synchronize bidirectional decoding.

time stamp determines when it should be decoded. In bidirectional coding these times can be quite different.

Audio packets only have presentation time stamps. Clearly if lip-sync is to be obtained, the audio and the video streams of a given program must have been locked to the same master 27 MHz clock and the time stamps must have come from the same counter driven by that clock.

With reference to Figure 6.4, the GOP begins with an *I* picture, and then *P1* is sent out of sequence prior to the *B* pictures. *P1* has to be decoded before *B1* and *B2* can be decoded. As only one picture can be decoded at a time, the *I* picture is decoded at time *N*, but not displayed until time $N+1$. As the *I* picture is being displayed, *P1* is being decoded at $N+1$. *P1* will be stored in RAM. At time $N+2$, *B1* is decoded and displayed immediately. For this reason *B* pictures need only PTS. At $N+3$, *B2* is decoded and displayed. At $N+4$, *P1* is displayed, hence the large difference between PTS and DTS in *P1*. Simultaneously *P2* is decoded and stored ready for the decoding of *B3* and so on.

In practice the time between input pictures is constant and so there is a certain amount of redundancy in the time stamps. Consequently PTS/DTS need not appear in every PES packet. Time stamps can be up to 700 ms apart in program streams and up to 100 ms apart in transport streams. As each picture type (*I*, *P* or *B*) is flagged in the bitstream, the decoder can infer the PTS/DTS for every picture from the ones actually transmitted.

Figure 6.5 shows that one or more PES packets can be assembled in a pack whose header contains a sync pattern and a system clock reference code which allows the decoder to re-create the encoder clock. Clock reference operation is described in section 6.4.

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 13 of 14

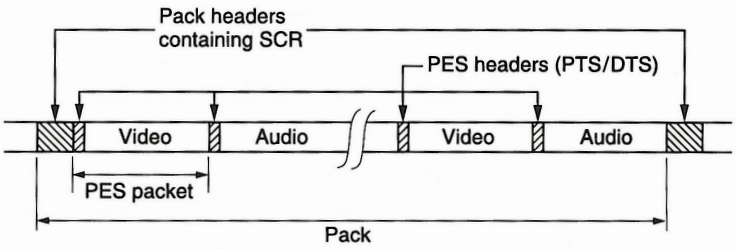

Figure 6.5 A pack is a set of PES packets. The pack header contains a clock reference code.

In AVC the rigid structure of MPEG-2 is relaxed somewhat. Temporal coding relies on identification of reference pictures and MPEG-2 does this with the combination of PTS/DTS and the known group structure. AVC allows the reference order of coded pictures to be decoupled from the display order. Thus encoded pictures can be sent in any order within buffering constraints. The encoder can select the order that minimizes overall bit rate. This requires a different picture identification mechanism. AVC uses frame number and Picture Order Count (POC). The frame number is a count which increments in the order of decoding whereas POC increments in the order of display. Note that these are picture counts, not time counts. These counts can be converted to time stamps by mutiplication by the period of each frame which is a constant for a given video format.

POC may be sent explicitly for each picture in slice headers, or it may be compressed. Where a repeating group structure is used, this will be sent in the sequence parameters and the decoder can compute the POC from the picture type and its transmitted position in the group. AVC encoders may change the group structure temporarily by sending delta POCs to change the decoder POCs from the predicted value to the actual value. In the event that the display order is the same as the decoding order the POC can be obtained from the frame number.

6.3 Transport streams

The MPEG-2 transport stream is intended to be a multiplex of many TV programs with their associated sound and data channels, although a single program transport stream (SPTS) is possible. The transport stream is based upon packets of constant size so that adding error-correction codes and interleaving[1] in a higher layer is eased. Figure 6.6 shows that these are always 188 bytes long. Transport stream packets should not be confused with PES packets which are larger and vary in size.

Snap - Exhibit 1023
Snap v. Nokia - IPR2025-01338
Page 14 of 14