DECODING OF DVB DIGITAL TELEVISION SUBTITLES

CHENGYUAN PENG AND PETRI VUORIMAA

Telecommunications software and Multimedia Laboratory,
Department of Computer Science and Engineering,
Helsinki University of Technology
P.O. Box 5400 FIN-02015 HUT
Tel. +358 9 4512916 Fax. +358 9 4515014
Email: pcy@tml.hut.fi and petri.vuorimaa@hut.fi

ABSTRACT

The new subtitling system will be established along with the digital television broadcasting. The system uses MPEG-2 multiplexing and region-based bitmap graphics (with indexed pixel colors) technologies to transmit subtitle data to set-top box. It provides more interactivity than the existing analogue television although interactivity is limited. This paper gives a description of Digital Video Broadcasting (DVB) subtitle encoding and transmission technologies. A subtitle decoding process model in set-top box is designed. It describes demultiplexing and decoding procedures and memory organization and allocation of a software-based decoder. The means of user controlling via digital television Navigator (i.e., enable/disable subtitles in a program and language setting/selection) is presented. The two approaches for implementation based on the client-server model are proposed to ensure better performance and the interoperability. Finally, technical requirements are discussed in the paper.

KEYWORDS:

Digital television subtitles, decoding, transport stream, distributed multimedia computing

1. INTRODUCTION

The primary aim of television subtitles (or captions) is to assist the viewer's communication with the TV screen more effectively. There are two main reasons for subtitling [1]. Firstly, subtitles are often provided for the purpose of language translation, where speech is translated so that the viewer can understand the program easier. Secondly, for those that are unable to adequately hear the audio track a dialogue can be fed to the viewer in order that they are informed of sound effects and spoken words.

Subtitles can be categorized as pre-prepared and live subtitles [2]. Live subtitling is created in real-time and used for live broadcast television, e.g., news programs or sports events. Pre-prepared subtitles are usually used in movies, videos, TV sitcoms, TV soap operas, etc. There are two main subtitle formats in analogue broadcast system, i.e., Line 21 closed captions and Teletext subtitles [2]. The third format is called open captions format, which does not belong to the broadcasting format. Line 21 closed captions can be found, e.g., in US, Canadian, and Japanese markets. UK uses Teletext to deliver subtitles.

The Line 21 closed captions and Teletext use the Vertical Blanking Interval (VBI) (i.e., the Line 21 captions uses the 21st line) of a video signal to carry subtitle information [1]. The subtitles data encoded in the video signal is inserted by data inserters into the VBI data stream and decoded by the receiver. It can only appear on the screens of television sets equipped with a built-in decoder chip. It has only basic letters, numbers, symbols, and limited colors. Only the foreground color can be changed. The decoding device (usually present in the television set) has complete control over the style of fonts, character sets, and size of the characters on the screen.

Open subtitles are always visible to everyone with or without a decoder. They are graphical elements (i.e., RGB graphical signal) and mixed with the analogue video signal through a computer graphical device. Subtitler can decide the look & feel of the subtitle (e.g., it's screen position, font style and character size, change the colors of both the fore- and backgrounds, transparency of background, and adding some special effects). Although it is not a broadcast format, the character generation products can be manufactured to broadcast standards.

Digital television will provide viewers a new subtitle service with more interactivity than the analogue television system. Viewers can use their remote controls to switch the subtitles on or off and select languages [5]. The subtitles are encoded and multiplexed to form subtitle streams, which are carried in MPEG-2 transport streams and transmitted via broadcast network to set-top boxes where the streams are demultiplexed and decoded. The decoded subtitles (i.e., images) are then rendered into the on-screen display (OSD) memory of set-top box. The OSD is in fact an overlay hardware module. The OSD is inserted down streams of any video processing designed to match the video to the display. Subtitles are therefore

ISBN 0-88986-321-0 © IASTED 2002

painted onto the “glass” and are invariant to the set-top box display modes.

The DVB subtitling system uses region-based, bit-mapped, and indexed color graphics [3]. The system applies Color Look-Up Tables (CLUTs) to define the colors of the graphical elements. The CLUT is independent of set-top box. The bitmapped pixel images are run-length coded and transmitted along with CLUTs to the set-top box. Such system uses small amounts of memory and performs a number of fast and special graphical effects.

The use of bit-mapped images allows the broadcaster to determine the look & feel of subtitles. The Tiresias is a candidate font [4]. It is a new easy-to-read font and developed in the Royal National Institute for the Blind (RNIB). Text can be in italics and in more colors.

The transport of the coded graphical elements is based on the MPEG-2 system [3]. The packetization process of Packetized Elementary Stream (PES) provides the means by which subtitles can be presented co-timed with the video and audio via the standard MPEG time-stamp mechanism in set-top box. The DVB subtitling system also provides a number of techniques that allows for efficient transmission of graphical data [3].

2. DISPLAY DATA STRUCTURE

Figure 1 shows the logical structure of DVB subtitles. Each TV program may have one or more epochs. The subtitle data for one epoch should be resident in the subtitle decoder memory until next epoch is signaled to be decoded. The shadow boxes between epochs indicate the boundaries of epochs. They also means the mode changes of the decoder (cf. Section 4). An epoch includes a sequence of one or more displays. A display (may stay on the screen for several seconds) is a completed screen of graphics. Each display is composed of a number of regions with specified positions.

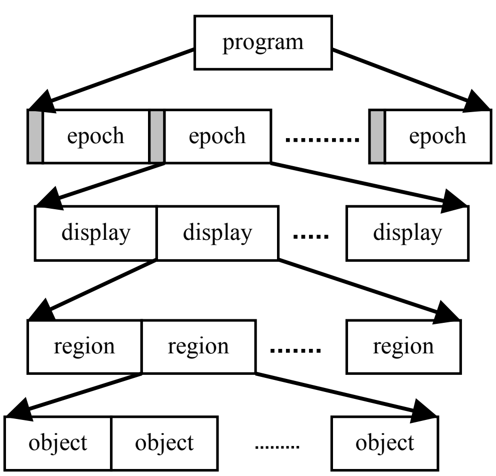

Figure 1. Subtitle hierarchical structure.

A region is a rectangular area of screen with a horizontal and vertical size and pixel depth. Each region is associated with a CLUT, which defines the color and transparency for each of the pixel codes. Pixel depths of 2, 4, and 8-bits are supported allowing up to 4, 16, or 256 different pixel codes to be used in each region. A CLUT in each region is used for translating the objects’ pseudo colors into the correct colors on the screen. A region can have a defined background color.

Each region consists of a number of graphical objects, which can be positioned within the region. The graphical objects can be anything, e.g., a subtitle, a channel logo, a map, etc. An object can be regarded as a graphical unit.

Figure 2. Example of a display screen.

Figure 2 shows an example of one display. This display consists of two regions in the upright and bottom. The upright region has one object (i.e., channel logo). The bottom region, which has transparency background, consists of three objects (i.e., three lines of text).

3. INTERACTIVITY VIA NAVIGATOR

The DVB-MHP (Multimedia Home Platform) specifies that the viewer can only control language and presentation display of subtitles (enable or disable). Other settings, e.g., the position of text and graphics are fully controlled by the broadcaster. The Navigator is the main interface to control subtitle settings [5].

3.1 Subtitle Settings

In order to control subtitle settings, the system of set-top box needs to maintain a configuration table in a cache, which is used by subtitle decoder. Table 1 shows an example of the data table. The value of each field is set by the viewer via the Navigator. The subtitle decoder will read these values of the data before decoding of transport streams.

The principle of subtitle setting is to set the field values the data record. For example, when the viewer has selected a TV program, the *event_id* and *service_id* field are set. Then the *transport_ stream_id* and *original network_id* are automatically set by the Navigator

ISBN 0-88986-321-0 © IASTED 2002

(Channel Guide or number on remote control) during tuning the channels by decoding/demultiplexing DVB-SI tables. The display state and language fields can also be set via the Navigator user interface. All the fields in data record have default values.

Field	Value
display_state	on / off
service_id	206
event_id	6666
transport_stream_id	10
original_network_id	22
language_code	Fre

Table 1. Configuration table of subtitles.

If the *display_state* is set "off", the subtitle decoder does not do anything. Otherwise, if the *display_state* is set "on", the *language_code* will be used as a search key to identify the PES-packets, which carries the subtitle streams with viewer's selected language, via PMT of the selected program. After that, the subtitle decoder will decode the selected PES-packets for presentation.

3.2 Demultiplexing Subtitle Streams

A single transport stream can carry several different subtitle streams. The different subtitle streams can be subtitles in different languages for a common program. Alternatively, they can be for different programs provided that the programs share a common Program Clock Reference (PCR) [3]. Different subtitle streams can also be supplied to address different display characteristics, e.g., different subtitle streams can be provided for 4:3 and 16:9 aspect ratio displays.

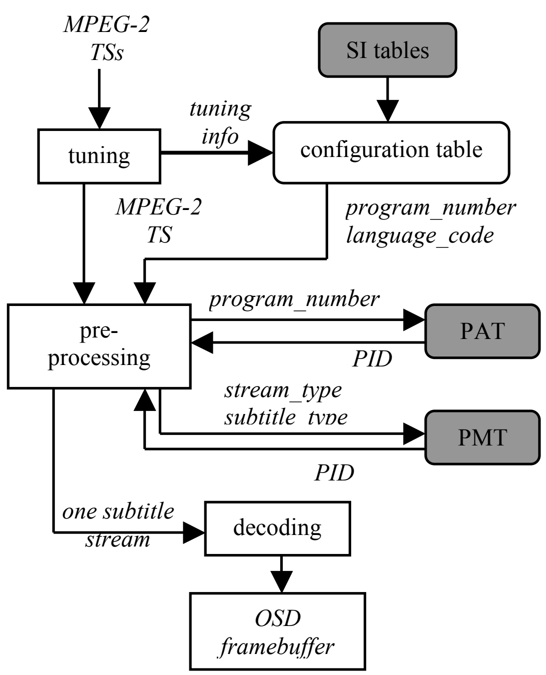

Figure 3. Subtitle setting procedure.

Figure 3 shows a filtering procedure in the decoder to extract one language program subtitles. A broadcasting network will carry one or more MPEG-2 transport streams. Through tuning process (write into cache), one transport stream is extracted and the tuning info is cached in the configuration table.

The pre-processing process completes the filtering function including demultiplexing subtitle streams from other elementary program streams, extracting one subtitle stream for the current TV program. It gets setting parameters from cache, from the Program Association Table (PAT) by providing program_number, and from Program Map Table (PMT) by providing stream_type (0x06) and subtitle_type. The decoding process performs the actual decoding task (cf. Section 5 for details).

3.3 Parameters for Subtitles Demultiplexing

A complete list of all the programs available in a transport stream is maintained in the PAT. The payload of a PID 0 packet in a transport stream is a PAT [7]. Each program has a program_number associated with it and is listed along with the PID value of the transport packets that contain its PMT. The more detailed referencing information comes in subsequent packets with the corresponding PMT.

Every program carried in a transport stream has a Program Map Table (PMT) associated with it. The PMT has the PIDs of subsequent packets containing each program's compressed video, audio, or data (e.g. subtitles). Table 2 shows the main fields of the PMT table and corresponding values, when subtitle streams are demultiplexing.

Field	Value
stream_type	0x06
PID	222
subtitle_descriptor 1	one subtitle stream
subtitle_descriptor 2	one subtitle stream
subtitle_descriptor N	one subtitle stream

Table. 2 Program Map Table.

The packets with stream_type 0x06 carry the subtitle elementary streams. One or more subtitling_descriptors in the PMT for a program describe the available subtitling streams (cf. Table 2).

The subtitle descriptor (cf. Table 3) is used by the Navigator. The ISO_639_language_code field contained in the subtitle descriptor contains three-character language code of the language of the subtitle [6]. The subtitle_type field specifies subtitle display format, e.g., 0x11 means DVB subtitles (normal) for display on 4:3 aspect ratio monitor. The composition_page_id and ancillary_page_id are used to identify the subtitle composition page and subtitle ancillary page (optional), respectively (cf. Section 4 for details).

Field	Value
ISO_639_laguage_code	Fre
subtitle_type	0x10
composition_page_id	2
ancillary_page_id	3

Table 3. Subtitle descriptor.

4. SUBTITLE STREAM

The transport streams of digital television program will comprises one or more packetized elementary streams, i.e., coded MPEG-2 compressed video, audio, and data (including subtitles, digital Teletext, and other private data) (cf. Figure 4). The data segments form one DVB subtitle stream. The subtitles segments are encapsulated in PES packets which are in-turn carried by transport packets. The number of segments carried by each PES packet is only limited by the maximum length of a PES packet defined by MPEG.

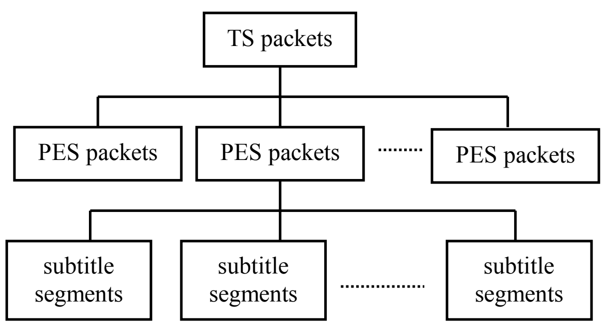

Figure 4. encapsulation of subtitle segments.

4.1 Subtitle Page Composition

The segments comprising a display in subtitling stream contain data from the composition page and the ancillary page (optional) (cf. Figure 5). The pages can be identified by composition_page_id and ancillary_page_id of the subtitle descriptor (cf. Table 3).

The composition page carries the segments unique to a single subtitle stream. It includes the page composition, region composition, CLUT definition, and object data segments (cf. Figure 5). The ancillary page is an optional page, which can be used to carry CLUT definition and object data segments that could be shared by more than one subtitle streams [3]. For example, it could be used to carry logos or character glyphs, or to carry objects that are common to two or more subtitle streams.

The page composition in the composition page is the top-level definition of a screen layout [3]. At any one time, only one page composition can be active for displaying, one or more other composition pages, which are carried simultaneously in the bit-stream, can be constructed in off screen memory.

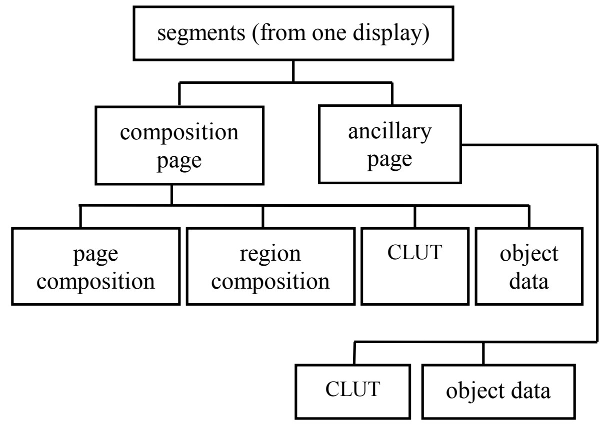

Figure 5. Organization of a single subtitle stream.

4.2 PES Structure

A PES-packet consists of a header and a payload. The sequences of segmented subtitle data are inserted into PES-packet payload. PES-packets may be of variable length. In a transport stream, the PES-packets from the various elementary streams are each divided among the payload parts of a number of transport packets and inserted into the TS packet payload.

Table 4 shows some fields of a PES-packet header, which are related to subtitle data. The *stream_id* field indicates the type of elementary streams. The Presentation Time Stamp (PTS) field indicates the beginning of the presentation time of the subtitle display carried by the PES packets [8]. The PTS in the PES packet provides presentation timing information for the subtitling data.

Fields	Size
start code prefix	3
stream_id	1
PES-packet length	2
Flags	2
PTS	2
other optional fields	xxx

Table 4. Sample PES packet header format.

Unlike video data, subtitle displays have no natural refresh rate. So, each display should be associated with a PTS to control, when it is displayed. Time stamps are the mechanism provided by the MPEG-2 systems layer to ensure correct synchronism between related elementary streams at a decoder [8]. All of the data for a display should be transmitted to the decoder in sufficient time to allow a decoder to decode all of the data by the time indicated by the PTS. A detailed description of the sequences of subtitle segments inserted into the payload PES packets can be found in [3].

ISBN 0-88986-321-0 © IASTED 2002

5. SUBTITLE DECODING MODEL

Figure 6 shows a data flow diagram of the subtitle decoding process (cf. Figure 3). It performs a series of activities, e.g., validating transport stream packets, extracting subtitle segments, composing each display of the epochs, etc. The input to the subtitle decoding process is a subtitle stream (i.e., filtered MPEG-2 transport stream packets). The MPEG-2 transport stream packets enter into a transport buffer with a size of 512 bytes. The rate of removing from this buffer should be 192 kbps [3].

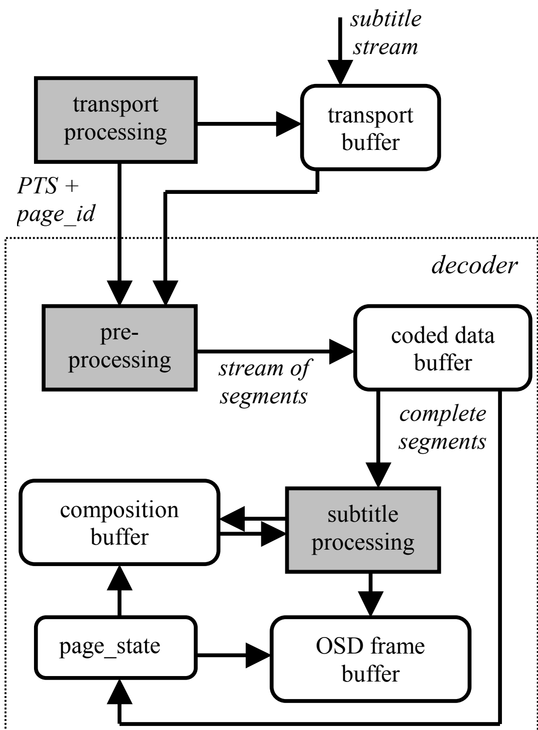

Figure 6. Data flow diagram of subtitle decoding.

The MPEG-2 transport stream packets are then processed in transport processing process by stripping off the packet headers of transport stream packets and of PES packets and then transfer the data stream to the pre-processing process. The PTS value of PES header and the page ids in *subtitle_descriptor* are also passed on to the next processes of the subtitle decoding. The output of the pre-processing process is a stream of subtitle segments, which are filtered based on their page ids (composition page or ancillary page).

The selected segments enter into the *coded data buffer*, which has a size of 24 KB. Only complete segments are removed from this buffer by the subtitle decoder. The removal and decoding of the segments should be instantaneous. The subtitle processing process is responsible for transmitting the complete segments to and from *composition buffer* and a complete display pixel data to the *On Screen Display (OSD) frame buffer* for final display. The rate for the transport of pixel data into the OSD frame buffer should be 512 kbps [3].

The *composition buffer* holds all the display data structures other than the displayed graphical objects. It has a capacity of 4 KB. The *composition buffer* holds the structural information of page composition, region composition, and CLUT definition. All regions' structure to be used in an epoch introduced by the RCSs are stored in this buffer. Also, all of the CLUT entries to be used during the epoch are stored in this buffer.

The capacity of the *OSD frame buffer* is the sum of the region_bits of all the defined regions. The region_bits can be calculated as follows: region_bits = region:width x region_height x region_depth. The capacity of this buffer is about 60KB [3].

The buffers are limited memory resources in set-top box. It is impossible to store all epochs for a TV program. Hence, this decoding model was designed so that the content of the buffers only hold subtitle data for one epoch. These allocations persist until the next page composition segment signals with page state of "mode change".

The *OSD frame buffer* and the *composition buffer* access the page state of the subtitle decoder (cf. Figure 6). The epoch for which this state is defined is between PCSs with page state of "mode change" (cf. page_state field of the PCS). A PCS with the page state of "mode change" destroys all current OSD frame and composition buffer allocations and leaves the contents of the memory undefined.

6. IMPLEMENTATION

6.1 An Approach

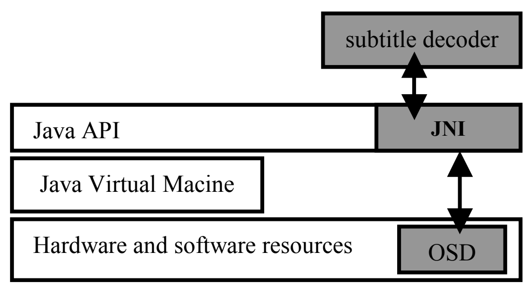

Figure 7. Software Architecture of Subtitle Decoder.

The whole system implementation has not been set up. Here we give an approach for implementation in the first stage. The system will be based on the client-server model. A server stores and sends encoded subtitle data streams cyclically to set-top box using HTTP protocols via the Internet. Each encoded epoch was in a file format and stored in the server.

The subtitle decoder will be implemented as a middleware other than hardware (e.g., LSI chip set). The main function will handle subtitle processing (cf. Figure

ISBN 0-88986-321-0

6), i.e., constructing one complete display in memory. Several functions are included in constructing, e.g., run-length decoding, synchronization of video and subtitles, etc. The software architecture is shown in Figure 7. The subtitle decoder is built on top of Java API and uses Java Native Interface (JNI) to access OSD resources. In this case, OSD hardware is needed.

An alternative approach is to use set-top box video memory as OSD and built on top of Java API. No JNI and OSD hardware are required. The OSD can be implemented as transparent Java AWT component, which is in front of video display. We are implementing this approach when writing this paper.

In order to obtain good quality of decoding, some factors have to be considered. The main issue of the subtitle decoding is to keep the last valid subtitle on the display until there is a new subtitle to replace it. Using first approach, this requires the two displays data being in OSD frame buffer at the same time. If each display takes up less than half the OSD frame buffer capacity it is possible for the subtitle decoder to switch between displays smoothly.

In second approach, no off-screen frame buffer is needed to get smooth display using multi-threading techniques provided by Java. However, more time will be spent in rendering subtitles in the Java AWT component.

7. CONCLUSIONS

The paper has given a detailed description of the DVB digital television subtitle system. A decoding model in set-top box was presented. Also, an approach to realizing the interactivity was introduced. The software-based implementation approaches have been proposed.

The digital television subtitle system is an open distributed system. This open system demands the contributions from different venders for encoding, multiplexing, modulation, transmission, demultiplexing, and decoding, etc. Hence, the essential task of implementation is to ensure the interoperability. The software implementation approach proposed in this paper can ensure the interoperability.

The greatest problems in the transmission of digital television subtitles are latency and synchronization [9]. The total time to encode and compress video at the broadcast side and the time to decode and decompress this broadcast video information within digital TV receivers may be as high as 400 milliseconds [1]. This latency will be particularly burdensome during real-time subtitling, when a subtitler simultaneously creates and transmits the words.

The arrival time of the subtitle data varies a lot at decoder [4]. The dynamics vary from one program to another, from pre-prepared to live subtitles. Therefore, the bit-rate requirements should be different. Usually the pre-prepared subtitles require a high peak bit-rate. Live subtitles have less variance in bit-rate, although the peak rate may be similar and demand real-time transmission. Therefore it is very important to ensure that encoders in sever side can accommodate the variance of coding load, multiplex can provide the bit-rate as required, and that decoders can deal with all types and conditions of subtitling.

We will test the system based on the approaches proposed in this paper and discuss the results (latency and synchronization) based on the implementation. After that, broadcasting object and data carousel protocols will be used instead of HTTP protocols.

8. ACKNOWLEDGEMENTS

This work was done in the Future TV project, which was funded by the National Technology Agency of Finland together with major Finnish television, telecommunications, and digital media companies. The author would like to thank Nokia Oyj foundation for the support during the research work.

9. REFERENCES

[1] Gary D. Robson and J. Hutchins, The Advantages and Pitfalls of Closed Captioning for Advanced Television, *proceedings of International Conference on Consumer Electronics*, 1998.

[2] Softel, Subtitling Overview. Available at http://www.softel.co.uk/l4_products_subtitling_overview.php.

[3] ETS 300 743, Digital Video Broadcasting (DVB); Subtitling systems. *European Telecommunications Standards Institute*, September 1997.

[4] N. E. Tanton and P. Weitzel, DVB Subtitling in an Open Environment, *International Broadcasting Convention (IBC 99)*, Amsterdam, 10-14 September, pp. 320-326.

[5] C. Peng, A. Lugmayr, and P. Vuorimaa, A Digital Television Navigator, to appear in *Multimedia Tools and Applications*, 2002.

[6] ETS 300 468, Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB systems. *European Telecommunications Standards Institute*, January 1997.

[7] Jeffery O. Noah, A Rational approach to testing MPEG-2, *IEEE Spectrum*, May 1997, pp. 67-72.

[8] P.A. Sarginson, MPEG-2: Overview of the systems layer. BBC Research and Development Report, BBC RD 1996/2.

[9] G. Forbes, Closed Captioning Transmission and Display in Digital Television, Proceedings of the 2nd International Workshop on Digital and Computational Video, Tampa, Florida, USA 8-9 February, 2001, pp. 126 – 131.