

UNTERSUCHUNG VON VERFAHREN
ZUR SCHRITTWEISEN EINFÜHRUNG
VON SIMULCAST FÜR HDTV
UND ENTWICKLUNG EINES EMPFÄNGER-DEMONSTRATORS

Diplomarbeit
am Institut für Medien- und Phototechnik
an der Fachhochschule Köln

Autorin
Birgit Schröter
aus Düsseldorf
Matrikel-Nr. 11036639

Referent: Prof. Dr. Klaus Ruelberg, Fachhochschule Köln
Korreferent: Dr. Rainer Schäfer, Institut für Rundfunktechnik, München

München, im November 2006

INVESTIGATION ON METHODS
FOR SUCCESSIVE INTRODUCTION
OF SIMULCAST FOR HDTV
AND DEVELOPMENT OF A RECEIVER-DEMONSTRATOR

Thesis
at the department
of
Media and Phototechnology
University of Applied Sciences Cologne

Author
Birgit Schröter
Düsseldorf
Matrikel-No. 11036639

First Reviewer: Prof. Dr. Klaus Ruelberg, University of Applied Sciences Cologne
Second Reviewer: Dr. Rainer Schäfer, Institut für Rundfunktechnik, Munich

Munich, in November 2006

Kurzbeschreibung

Titel: Untersuchung von Verfahren zur schrittweisen Einführung von Simulcast für HDTV und Entwicklung eines Empfänger-Demonstrators

Autorin: Birgit Schröter

Referenten: Prof. Dr. Klaus Ruelberg, Fachhochschule Köln
Dr. Rainer Schäfer, Institut für Rundfunktechnik, München

Zusammenfassung: Simulcast bedeutet die zeitgleiche Ausstrahlung des gleichen Inhalts auf weiteren, zusätzlichen Kanälen; in diesem Fall die zeitgleiche Ausstrahlung desselben Events auf einem SD- und einem HD-Kanal. Zur automatischen Umschaltung zwischen diesen beiden Kanälen soll zukünftig eine entsprechende Signalisierung ausgewertet werden. Um zu beweisen, dass solch ein Szenario praktisch umsetzbar ist, wurde in dieser Diplomarbeit ein Empfänger-Demonstrator entwickelt.

Stichwörter: Simulcast, HDTV, DVB, EIT, Linkage Descriptor

Sperrvermerk: Die vorliegende Arbeit unterliegt keinem Sperrvermerk.

Datum: 30. November 2006

Abstract

Title: Investigation on methods for successive introduction of simulcast for HDTV and development of a receiver-demonstrator

Author: Schröter, Birgit

Reviewers: Prof. Dr. Klaus Ruelberg,
University of Applied Sciences Cologne
Dr. Rainer Schäfer, Institut für Rundfunktechnik, Munich

Abstract: Simulcast means broadcasting the same content simultaneously on further additional channels; in this case it means the simultaneous broadcasting of the same event on a SD- and on a HD-channel. For switching automatically between these channels there ought to be an appropriate signaling evaluated in the future. In this thesis a demonstrator for the receiving end has been developed in order to prove the practicability of such a scenario.

Key Words: Simulcast, HDTV, DVB, EIT, Linkage Descriptor

Confidentiality: The thesis is non-confidential

Date: 30 November 2006

Inhaltsverzeichnis

1	Einführung/Problemstellung	1
1.1	Einführung von Simulcast	2
1.2	Motivation für Simulcast	3
1.3	„Henne-Ei“-Problem	3
1.4	Aufgabenstellung	4
1.5	Strukturierung der Arbeit	5
2	Einführung von Simulcast für HDTV	6
2.1	Wege zur Ausstrahlung von HD in DVB	6
2.2	Situation bei den privaten Rundfunkanbietern	7
2.3	Situation und Modelle bei den öffentlich-rechtlichen Rundfunkanstalten	9
3	Grundlagen der genutzten Konzepte	12
3.1	DVB-Transportstrom	12
3.2	Das Hexadezimalsystem	14
3.3	Bouquet, Services und Events	15
3.4	Zusatzdaten	16
3.5	Event Information Table (EIT)	20
3.6	Linkage Descriptor	23
3.7	Finite State Machine	27
4	Untersuchte Verfahren und erweiterte Konzepte	28
4.1	IRT-Proposal für eine Erweiterung von EN 300 468 und TR 101 211	28
4.1.1	Abbildung des Proposals in einem State-Diagramm	31
4.1.2	Erweitertes Konzept für das Proposal	32
4.2	Ergänzungen zum IRT-Proposal	34
4.3	EPG im HD-Kanal	34
4.4	Verschiebungen und Störungen im Sendeablauf	37
5	Empfänger-Demonstrator	39
5.1	Versuchsaufbau und verwendete Geräte	39
5.1.1	Blockschaltbild	41
5.1.2	Hardware	42
5.1.3	Software	44
5.2	Verwendete Module	48
5.3	Funktionen des EIT-Analyse-Programms	50

5.3.1	State diagram.....	50
5.3.2	Auffinden der benötigten Daten in der Prozedur HandleEitData.....	52
5.3.3	Bedingungen a bis e.....	56
5.3.4	States 0 bis 3.....	59
5.3.5	Senden von Befehlen über das Infrarot-Modul.....	62
5.3.6	Verarbeitung von empfangenen Infrarot-Befehlen.....	63
5.3.7	Eingabe von Basisdaten über ein Config-File.....	65
5.3.8	Paralleler Schaltvorgang von Settop-Box und Analyse-Programm.	67
5.3.9	Grafische Darstellung des Programmablaufs.....	68
5.4	Ausgewählte Details zu beschrifteten Lösungsansätzen.....	71
5.4.1	Event ID.....	71
5.4.2	Austausch des HD-Receivers.....	72
5.4.3	Umstellung der Programmierung auf eine State-Machine.....	73
6	Zusammenfassung und Ausblick.....	74
6.1	Diskussion.....	74
6.2	Schritte für die praktische Umsetzung.....	75
7	Abkürzungen und Begriffe.....	77
8	Literaturverzeichnis.....	85
9	Anhang.....	88
9.1	Quellcode Unit1.....	88
9.2	IRT-Proposal.....	115
9.3	Config-Datei.....	118

Abbildungsverzeichnis

Abbildung 1: Ausstrahlung: Wege zu HD in DVB (nach [9])	6
Abbildung 2: Schema für den Gemeinschaftskanal der Dritten Programme	10
Abbildung 3: Schematischer Aufbau eines Transportstroms.....	12
Abbildung 4: Aufschlüsselung des TS-Headers	13
Abbildung 5: MPEG-2 Programm- und Transportstrom-Multiplex (nach [15])	14
Abbildung 6: Bouquet von ZDF Vision mit zugehöriger Service ID (nach [17])	16
Abbildung 7: Verkettung der PSI-Tabellen [14]	17
Abbildung 8: Program specific information, (nach [11])	17
Abbildung 9: Allgemeine Organisation der Service Information (SI), [aus EN 300 468]	19
Abbildung 10: Grafische Übersicht über die unterschiedlichen EIT-Typen.....	21
Abbildung 11: Syntax zur Event information section, nach [20].....	22
Abbildung 12: Hex Dump einer EIT actual following.....	23
Abbildung 13: Syntax-Tabelle zum Linkage Descriptor (nach [20])	24
Abbildung 14: Grafische Darstellung einer EIT mit dem TS-Analyser von Tektronix	26
Abbildung 15: Vereinfachtes State-Diagramm (nach [20])	27
Abbildung 16: HD/SD-Umschaltvorgang [nach IRT-Proposal für TR 101 211].....	29
Abbildung 17: HD/SD-Umschaltvorgang aus IRT-Proposal für TR 101 211.....	31
Abbildung 18: State Diagramm nach dem IRT-Proposal.....	31
Abbildung 19: ARD-Gemeinschaftskanal mit zwei unterschiedlichen Zusatzinformationen.....	37
Abbildung 20: Versuchsaufbau nach Sender- und Empfängerseite getrennt.....	39
Abbildung 21: Foto vom Versuchsaufbau.....	40
Abbildung 22: Blockschaltbild der im Demonstrator miteinander verbundenen Komponenten.....	41
Abbildung 23: Dektec DTA-140, PCI-Bus-Karte für DVB-ASI Input + Output.....	42
Abbildung 24: Dektec DTU-205, ASI/SDI Output Adapter.....	42
Abbildung 25: NewTec NTC/2180, L-Band Satelliten Modulator	43
Abbildung 26: Humax PR-HD 1000, Digitaler Satelliten-Receiver.....	43
Abbildung 27: IRTrans USB Modul, Infrarot Sender und Empfänger	44
Abbildung 28: Screenshot der Oberfläche des DVB-Playout-Servers	45

Abbildung 29: Prinzip mit dem die EITs über Batch-Dateien geändert werden	46
Abbildung 30: Screenshot der Oberfläche des EIT-Editors	47
Abbildung 31: Kommunikationsprinzip des Infrarotmoduls (IrTrans)	47
Abbildung 32: Ausschnitt aus einer ASCII-Remote-Datei.....	48
Abbildung 33: State Diagramm des Empfänger-Demonstrators.....	50
Abbildung 34: Vereinfachte Struktur der Prozedur HandleEitData	53
Abbildung 35: Belegung der Variablen für das DVB-Triple in der Prozedur HandleEitData	54
Abbildung 36: Event Information Table (EIT) aufgeteilt nach Bytes	54
Abbildung 37: Prinzip der schrittweisen Informationssuche in der Prozedur HandleEitData	56
Abbildung 38: Bedingung für a=true im Sourcecode	57
Abbildung 39: Bedingung für b=true im Sourcecode	57
Abbildung 40: Bedingung für c=false im Sourcecode	58
Abbildung 41: Bedingung für d=true im Sourcecode	58
Abbildung 42: Bedingung für e=true im Sourcecode	59
Abbildung 43: Vereinfachte Darstellung der case-Anweisung in der Prozedur HandleEitData	61
Abbildung 44: Sourcecode der Prozedur SendPaceChannel.....	62
Abbildung 45: Sourcecode der Verarbeitung des Infrarot-Befehls „Programm+“ .	64
Abbildung 46: Prinzip des Auslesevorgangs und Findens des richtigen Kanals aus der Config-Datei	67
Abbildung 47: Anzeige von Status 0, SD-Videobild und SD-Anzeige auf der Settop-Box.....	69
Abbildung 48: Anzeige von Status 1, HD-Videobild und HD-Anzeige auf der Settop-Box.....	69
Abbildung 49: Anzeige von Status 2, HD-Videobild und HD-Anzeige auf der Settop-Box.....	69
Abbildung 50: Anzeige von Status 3, SD-Videobild und SD-Anzeige auf der Settop-Box.....	70
Abbildung 51: Screenshot der Bedienoberfläche des EIT-Analyse-Programms...	71

Tabellenverzeichnis

Tabelle 1: Gegenüberstellung von Dezimal-, Hexadezimal- und Dualsystem (nach [19]).....	14
Tabelle 2: PID-Zuordnung für die SI, aus [20]	20
Tabelle 3: Linkage type coding (nach [20]).....	24
Tabelle 4: Liste der Bedingungen a bis c laut IRT-Proposal.....	32
Tabelle 5: Liste der Bedingungen a bis c nach dem erweiterten IRT-Proposal	33
Tabelle 6: Liste der Bedingungen a bis e für den Empfänger-Demonstrator	51
Tabelle 7: Tabellarische Darstellung der Werte der Config-Datei.....	66

1 Einführung/Problemstellung

Der Anfang für die Einführung von HDTV, einem neuen zukünftigen Fernsehstandard mit 1.080 bzw. 720 Zeilen, der den heutigen SDTV-Standard (PAL) mit 576 (digital) sichtbaren Zeilen ablösen wird, ist gemacht. Die Gerätehersteller bieten, neben hochwertigen Projektoren, eine Vielzahl von Flachdisplays mit über 80 cm Bildschirmdiagonale und einer Auflösung von etwa 1.000 Zeilen an. Nun sind die Rundfunkanbieter an der Reihe, diese Displays und Projektoren auch mit entsprechend hochwertigem Inhalt zu versorgen. Für die Programmanbieter bedeutet dies einen tief greifenden Umstellungsprozess und einen hohen finanziellen Aufwand. HDTV wird dabei in einem längerfristigen Übergangsprozess eingeführt werden [1]. Derzeit ist das Hauptproblem in Bezug auf die Einführung von HDTV ähnlich wie damals bei der Einführung des Farbfernsehens. Man muss sich überlegen, ob man den neuen Standard mit dem alten kompatibel gestaltet (Stichwort Rückwärtskompatibilität), den existierenden komplett ersetzt oder ob man ihn gleichzeitig (simultan) mit dem alten ausstrahlt, letzteres mit der Maßgabe, dass der existierende Standard später abgeschaltet wird [2]. Der erste Punkt scheint unstrittig, denn der HDTV-Standard existiert und ist nicht mit SDTV kompatibel. HDTV benutzt neben der höheren Zeilenzahl auch ein anderes Kompressionsverfahren und eine andere Modulation, so dass HDTV nicht mit digitalen Settop-Boxen dieser Generation empfangbar wäre (vgl. Kapitel 2.1).

Den existierenden Standard einfach abzuschalten, wie das beispielsweise in einigen Regionen bei der Einführung von DVB-T mit dem analogen terrestrischen Signal passiert ist, scheidet aus, weil ansonsten bei Zuschauern, die noch einen herkömmlichen Empfänger haben, die Grundversorgung nach dem Grundversorgungsauftrag nicht mehr gewährleistet wäre. Auch bei der Einführung von DVB-T erfolgte keine zeitgleiche Abschaltung des alten und Einführung des neuen Standards. Stattdessen gab es eine, wenn auch kurze, wenige Monate dauernde, Übergangsphase, in der beides, analoges terrestrisches Fernsehen und DVB-T, ausgestrahlt wurde, und in der die Kanäle sukzessive auf DVB-T umgeschaltet wurden. Des Weiteren wurde die Umstellung auf DVB-T lange

politisch vorbereitet. Eine erneute Änderung des Standards würde beim Zuschauer verständlicherweise Ärger hervorrufen. Darüber hinaus stellt sich für die Broadcaster die Frage der Finanzierung zur Einführung von HDTV. Eine neuerliche Diskussion über die Erhöhung der Rundfunkgebühren ist zurzeit politisch nicht erwünscht, auch ließe sich eine solche Erhöhung wahrscheinlich nicht durchsetzen.

Über mehrere Jahre werden also HDTV und SDTV parallel im so genannten Simulcast-Betrieb ausgestrahlt werden, der im nächsten Abschnitt näher beschrieben wird.

1.1 Einführung von Simulcast

Das Wort Simulcast ist ein Kunstwort und setzt sich aus den englischen Begriffen „simultaneous“ (zeitgleich, simultan) und „broadcast“ (Ausstrahlung, Aussendung) zusammen. Es bedeutet eine zeitgleiche Ausstrahlung des gleichen Inhalts auf weiteren, zusätzlichen Kanälen bzw. Programmen.

Das Wort „zusätzlich“ ist hierbei entscheidend, denn für jedes SD-Programm (bzw. jeden SD-Service) muss beim vollständigen Simulcast die Bandbreite für ein komplettes HD-Programm bereitgestellt werden. Dabei ist die Bandbreite, die ein HDTV-Programm benötigt größer als bei SDTV. Vereinfacht gesagt ist die Bandbreite, und somit auch die Datenmenge, bei HDTV mindestens doppelt so groß wie bei SDTV. Daher reicht die Bandbreite eines SD-Kanals bei weitem nicht aus um ein HD-Programm zu betreiben - ein nicht unerheblicher Kostenfaktor für die Programmanbieter. So kostet die Nutzung eines Satelliten-Transponders die Rundfunkbetreiber um die 6 Millionen Euro im Jahr. Die ARD betreibt davon zurzeit vier Stück. Für den Betrieb von HDTV müssten weitere Transponder angemietet werden. In Anbetracht der hohen Kosten ist dabei zu beachten, wie viele Zuschauer derzeit tatsächlich in der Lage wären, HDTV zu sehen. Das sind heute erst ungefähr 30.000, Tendenz steigend.

Es gibt dabei zwei grundsätzlich verschiedene Ansätze des Simulcast-Betriebs:

- Den ‚vollständigen‘ Simulcast-Betrieb, bei dem der Inhalt auf dem SD- und dem HD-Kanal identisch und nur die Auflösung unterschiedlich ist (vgl. Kapitel 2.2)
- Den ‚teilweisen‘ Simulcast-Betrieb, bei dem sich mehrere Rundfunkanstalten einen HD-Kanal teilen (vgl. Kapitel 2.3)

1.2 Motivation für Simulcast

Spätestens dann, wenn entsprechende Inhalte bei den Offline-Medien (Blu-ray, HD-DVD) in Form von Filmen oder Spielen in hochauflösender Qualität und in größerer Zahl verfügbar sind, werden mit dem gestiegenen Qualitätserlebnis auch die Ansprüche und Erwartungen der Zuschauer an die Übertragungsqualität der Sender steigen [1]. HDTV kommt sozusagen über die Hintertür zum Zuschauer und die Sender werden entsprechend reagieren müssen, wenn sie nicht an Attraktivität verlieren wollen. HDTV im Regelbetrieb ist nur eine Frage der Zeit.

Würde man die Ausstrahlung von SDTV aber abrupt abschalten und HDTV einführen, würde die Mehrheit der Zuschauer keinen Empfang mehr haben oder müsste sich neue Geräte kaufen. Hinzu kommt, dass gerade das öffentlich-rechtliche Fernsehen, nach dem so genannten Niedersachsenurteil des Bundesverfassungsgerichts von 1986, zur Grundversorgung der Bevölkerung verpflichtet ist. Dabei ist „Grundversorgung eindeutig nicht als Minimalversorgung zu verstehen, sondern schließt die gesamten Programmangebote in den Bereichen Bildung, Information und Unterhaltung ein“ [3]. Bei den Privatsendern ist die Verpflichtung zur Versorgung der Zuschauer etwas anders gelagert, aber die öffentlich-rechtlichen Rundfunkanstalten müssen bei der Einführung von HDTV auch den Empfang von SDTV über mehrere Jahre weiterhin gewährleisten. Es führt also kein Weg daran vorbei, für eine bestimmte Zeit Simulcast auszustrahlen.

1.3 „Henne-Ei“-Problem

Für den Fall, dass zukünftig HDTV im Simulcast-Betrieb, nach dem Modell, dass sich mehrere Sender einen Kanal teilen (wird in Kapitel 2.3 näher beschrieben), eingeführt ist, gäbe es die Möglichkeit, dem Zuschauer einen gewissen Komfort zu bieten. Der Zuschauer müsste sich dann nicht darum kümmern muss, ob eine

Sendung in HD vorliegt oder nicht. Der Receiver würde, jeweils abhängig davon, ob der Zuschauer HDTV empfangen kann oder nicht, die Umschaltung auf den HD-Kanal automatisch vornehmen, wenn eine Sendung, gleichzeitig zur Ausstrahlung in SD, auch in HD verfügbar ist. Über einen Menüeintrag in der Settop-Box sollte diese Funktion jedoch für denjenigen, der diese automatische Umschaltung nicht wünscht, abschaltbar sein.

Es gibt vom IRT (Institut für Rundfunktechnik) einen Vorschlag, der eine entsprechende Kennzeichnung in den Zusatzdaten der jeweiligen Sendung vorsieht, so dass der Receiver entsprechend erkennen kann, ob es sich um eine Sendung in HD oder in SD handelt. Die Voraussetzungen hierfür und die genaue Art der Kennzeichnung sind Gegenstand dieser Arbeit und werden in den Kapiteln 2 und 4 detailliert beschrieben.

Das Grundproblem bei der Einführung neuer Konzepte besteht häufig darin, dass keiner den ersten Schritt machen will. Die Gerätehersteller sehen z.B. keinen Anlass, eine solche Funktion in ihre Receiver zu implementieren, weil es noch keine Ausstrahlung gibt, auf die diese Funktion sinnvoll anwendbar wäre. Die Rundfunkanbieter wiederum sind der Meinung, dass eine Ausstrahlung nach diesem Simulcast-Prinzip, dass sich mehrere Sender einen Kanal teilen, keinen Sinn macht, weil es keine Geräte gibt, die die Funktion der automatischen Umschaltung beherrschen. Also ein typisches Henne-Ei-Problem, zu dessen Lösung diese Arbeit beitragen kann.

1.4 Aufgabenstellung

In der vorliegenden Arbeit werden verschiedene Szenarien für die Einführung von Simulcast für HDTV betrachtet werden. Dabei soll neben der Optimierung des existierenden IRT-Vorschlags bewiesen werden, dass die herausgearbeitete Lösung funktioniert und sich auch in einen Receiver implementieren ließe. Dafür wird ein Empfänger-Demonstrator entwickelt und programmiert, um den Ablauf, mögliche verbleibende Probleme des Vorschlags, die Funktionalität und die Bedienung anschaulich darstellen zu können.

1.5 Strukturierung der Arbeit

Im folgenden Kapitel wird zunächst die aktuelle Situation bei den privaten und öffentlich-rechtlichen Fernsehsendern beschrieben. Anschließend werden die zum Verständnis der weiteren Arbeit erforderlichen Kenntnisse über den DVB-Transportstrom, die Event Information Table, kurz EIT, und den Linkage Descriptor vermittelt. Im 4. Kapitel wird dann eine Möglichkeit zur Einführung von Simulcast auf der Basis des bereits oben erwähnten IRT-Vorschlags sowie dessen Erweiterung vorgestellt. In Kapitel 5 folgt eine genaue Beschreibung des entwickelten Empfänger-Demonstrators inklusive einer Beschreibung der verwendeten Hard- und Software und der Funktionalität der programmierten Module.

2 Einführung von Simulcast für HDTV

In diesem Kapitel wird zunächst die aktuelle Situation der privaten Rundfunkanbieter beschrieben, da diese in Deutschland zurzeit die Einzigen sind, die tatsächlich auch HDTV ausstrahlen. Anschließend folgt eine Beschreibung der Situation bei den öffentlich-rechtlichen Rundfunkanstalten, die in Zukunft auch in HD senden wollen.

2.1 Wege zur Ausstrahlung von HD in DVB

In der Technischen Spezifikation TS 101 154 [10] sind die Mindestanforderungen für Integrated Receiver-Decoder (IRD) angegeben, die alle IRDs einhalten bzw. übertreffen müssen. Eine solche Spezifikation ist notwendig, damit es eine einheitliche Basis gibt, d.h. damit es Funktionen gibt, die alle Geräte beherrschen und damit nicht jedes Gerät nach ganz anderen Regeln arbeitet. Aus dieser Spezifikation ergeben sich verschiedene Anforderungen für verschiedene Standards (SD oder HD) und Kompressionsverfahren (MPEG-2 und H.264).

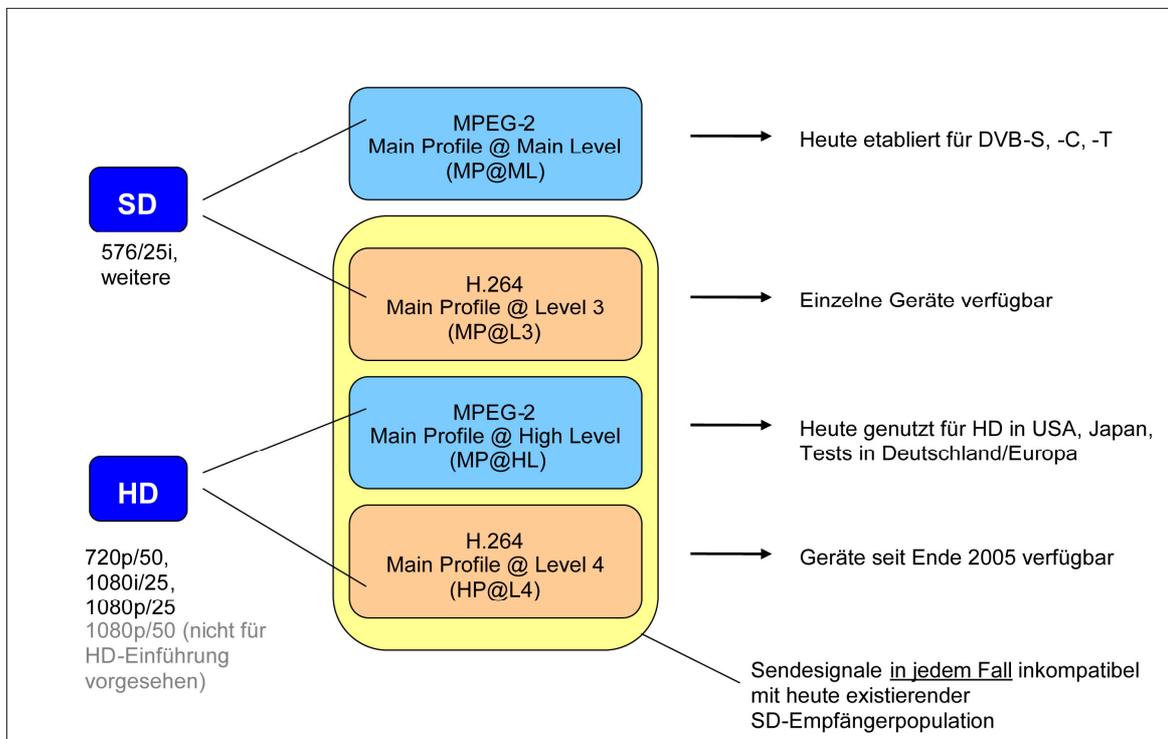


Abbildung 1: Ausstrahlung: Wege zu HD in DVB (nach [9])

Für die Ausstrahlung im DVB-Standard, gibt es prinzipiell mehrere Möglichkeiten (siehe Abbildung 1): Zum einen gibt es zwei unterschiedliche Kompressionsverfahren, zum anderen innerhalb dieser Kompressionsverfahren die Unterscheidung nach Profiles (Einschränkung der Komplexität) und Levels (Einschränkung der Parameter). Für SD ist das Kompressionsverfahren MPEG-2 mit Main Profile at Main Level (MP@ML) heute für das Satelliten- (DVB-S), Kabel- (DVB-C) und terrestrische (DVB-T) Fernsehen etabliert. Daneben gibt es für SD noch die Möglichkeit in H.264 zu übertragen (z.B. IPTV). Hierfür sind jedoch nur einzelne Geräte verfügbar. Für HD gibt es ebenso die Möglichkeit in MPEG-2 oder H.264 zu übertragen.

Aus TS 101 154 ergibt sich, dass die Sendesignale mit H.264 (SD und HD) und MPEG-2 Main Profile at High Level für HD nicht mit der heutigen SD-Empfängerpopulation kompatibel sind.

Das bedeutet, dass eine Ausstrahlung im Simulcast-Betrieb unumgänglich ist.

2.2 Situation bei den privaten Rundfunkanbietern

Es gibt bereits eine Reihe von Privatsendern in Deutschland, die in High Definition senden. In allen Fällen werden diese Sender über Satellit verbreitet, da diese bisher am zuverlässigsten die erforderliche Bandbreite für HDTV bereitstellen. Einige dieser Kanäle können aber auch über einen digitalen Kabelanschluss (z.B. Kabel BW) empfangen werden. Terrestrisch gibt es in Deutschland zurzeit (noch) kein HDTV zu sehen.

Bei den privaten Rundfunkanbietern muss man dabei zwischen Pay-TV und Free-TV unterscheiden. Die Pay-TV-Anbieter, wie z.B. Premiere, die einen speziellen HDTV-Kanal betreiben, können damit rechnen, durch dieses Angebot neue Kunden zu gewinnen. Im werbefinanzierten Free-TV (z.B. ProSiebenSat.1) dagegen ist weder eine Kosteneinsparung noch eine kurzfristige Refinanzierung zu erwarten. Die Free-TV-Anbieter wollen bzw. müssen aber ihre Attraktivität und Reichweite erhalten [1].

Der Pay-TV-Sender Premiere ist bereits mit drei Spartenkanälen mit HDTV vertreten (HD Film, HD Sport, Discovery), die aber separate HDTV-Kanäle sind und somit kein Beispiel für den Simulcast-Betrieb darstellen. Daneben gibt es den

Spielfilm- und Unterhaltungskanal Anixe HD [4], die belgischen HDTV-Kanäle HD1 und HD2/HD5 des Anbieters Euro1080. Dabei werden letztere im Timesharing-Verfahren ausgestrahlt, was bedeutet, dass HD5 inaktiv ist, wenn HD2 aktiv ist und umgekehrt [5]. Außerdem sind in Deutschland über Satellit noch einige fremdsprachige Sender, wie z.B. BBC HD, empfangbar. Des Weiteren gibt es verschiedene Demonstrationskanäle diverser Anbieter, die in Europa in HDTV ausgestrahlt werden. Sie wollen dem Zuschauer die neuen Möglich- und Fähigkeiten des neuen hochauflösenden Fernsehens schmackhaft machen: Astra HD Demo Channel, Astra HD Promo, Evenement HD, National Geo HD, Kanal D HDTV, Polsat HD, Rai Uno (HD Test), Eutelsat Test G1, Satmode HD Promo, u.a. [6], [7].

Der Medienkonzern ProSiebenSat.1 strahlt seit Oktober 2005, neben den SD-Kanälen ProSieben und Sat.1, die beiden Fernsehsender ‚ProSieben HD‘ und ‚Sat.1 HD‘ im Regelbetrieb in HDTV aus. Das Unternehmen versteht sein HDTV-Engagement dabei als Initiative zur Weiterentwicklung des Mediums Fernsehen und zum Erhalt seiner „Technologieführerschaft in der europäischen Fernsehlandschaft“ [8]. Bei ProSiebenSat.1 erfolgt die Ausstrahlung im vollständigen Simulcast-Betrieb, was bedeutet, dass der Inhalt der beiden HDTV-Kanäle dem regulären Programmablauf von Sat.1 (SD) und ProSieben (SD) entspricht. Sat.1 HD und ProSieben HD haben dabei seit dem Sendestart den Anteil ihrer nativen HD-Inhalte kontinuierlich erhöht. Da aber noch nicht alle Sendungen hochauflösend produziert sind, werden die Sendungen, die im Original nur als SD-Signal vorliegen, auf HDTV-Auflösung hochskaliert, erreichen damit aber nicht die mit HDTV mögliche Qualität. Um sich eine solche Sendung in HDTV ansehen zu können, muss der Zuschauer lediglich auf den HD-Kanal von ProSieben oder Sat.1 umschalten. Um dann auch ein Bild sehen zu können, ist entsprechendes HD-fähiges Equipment (Display, Receiver) sowie ein entsprechender Empfang von HD-Signalen (z.B. über Satellit oder digitales Kabel) erforderlich. Für den Fall, dass der Zuschauer den SD-Kanal sieht, kann eine Umschaltung auf das höherwertige Signal in Zukunft auch automatisch erfolgen, wenn im Sinne der vorliegenden Arbeit entsprechende technische Voraussetzungen beim Zuschauer gegeben sind. Die Signalisierung für das

Umschalten wird dann entsprechend ignoriert, wenn diese Voraussetzungen nicht gegeben sind. Dies wird in Kapitel 4 näher erläutert.

2.3 Situation und Modelle bei den öffentlich-rechtlichen Rundfunkanstalten

Auch die öffentlich-rechtlichen Rundfunkanstalten sind davon überzeugt, dass sich HDTV langfristig in Europa durchsetzen wird. Dabei planen sie zunächst einen schrittweisen Übergang von SDTV auf HDTV, wobei erst einmal durch größere Datenraten die Möglichkeiten zur Bildqualitätsverbesserung bei der digitalen Übertragung ausgeschöpft werden sollen. Davon profitieren vor allem die Zuschauer, die ihre Programme bereits digital empfangen. Eine Einführung von HDTV im Regelbetrieb ist frühestens für 2008/2010 geplant [1].

Zurzeit wird, neben einem Simulcast-Betrieb für das ZDF-Programm, bei der ARD die Einführung von zwei HDTV-Kanälen diskutiert: Einen für den kompletten Simulcast-Betrieb für das Programm „Das Erste“ und einen gemeinschaftlichen HDTV-Kanal für die Dritten Programme. Im ersten Fall wird es vom Prinzip her so funktionieren, wie bereits für ProSieben und Sat.1 im Abschnitt 2.2 beschrieben. Der Inhalt von ‚Das Erste SD‘ und ‚Das Erste HD‘ wird identisch sein, also vollständiger Simulcast. Der Gemeinschaftskanal für die Dritten setzt sich aus einzelnen Beiträgen der verschiedenen Landesrundfunkanstalten zusammen. So gäbe es für eine Rundfunkanstalt die Möglichkeit, eine Sendung, die in HD vorliegt, zeitgleich auf dem Gemeinschaftskanal auszustrahlen. Dieses Konzept macht für Anbieter mit einem kompletten Bouquet, wie eben die ARD, Sinn.

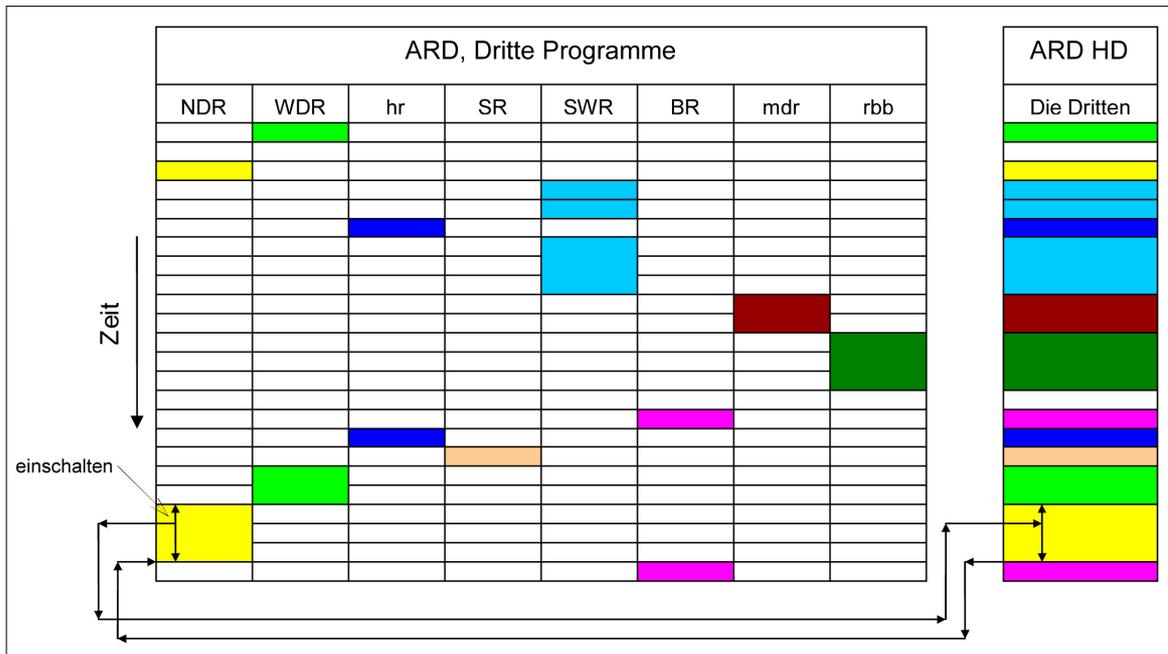
Das Prinzip dieses Gemeinschaftskanals wird in Abbildung 2 erläutert:

Schaltet der Zuschauer z.B. in eine laufende Sendung des NDR und findet der Receiver einen entsprechenden Hinweis in den Zusatzdaten, dass gerade auf dem HD-Gemeinschaftskanal eben diese Sendung auch in höherer Auflösung läuft, dann soll der Receiver auf den HD-Kanal umschalten. Der Zuschauer sieht nun die gewählte Sendung in HDTV. Endet diese Sendung und folgt im HD-Kanal keine weitere Sendung des NDR, soll der Receiver wieder auf den Herkunftskanal, den NDR, zurückschalten.

Den HD-Kanal teilen sich dabei die Dritten Programme der ARD. Wie in Abbildung 2 anhand der farbigen Markierung zu sehen ist, wird der Gemeinschaftskanal von

den verschiedenen Landesrundfunkanstalten gespeist. Die farbigen Felder markieren dabei die Sendungen, die gleichzeitig in SDTV und HDTV verfügbar sind.

Der Gemeinschaftskanal kann zwar auch als separater Kanal angewählt werden, aber vom Prinzip her soll sich der Zuschauer, wie oben beschrieben, für einen SD-Kanal entscheiden und das Umschalten im Falle, dass eine Sendung in HDTV vorliegt, seinen Geräten und der entsprechenden Signalisierung überlassen. Es wird möglicherweise, aufgrund der je nach Landesrundfunkanstalt unterschiedlichen Programmschemata, nicht möglich sein, den Gemeinschaftskanal lückenlos mit Inhalt zu füllen. Diese Lücken ließen sich dann beispielsweise mit Trailern füllen, die den Mehrwert von HDTV bewerben.



bringen, nach dem Prinzip ‚Montags ist beim WDR HD-Tag‘ und den kleineren z.B. jeweils alle zwei Wochen. Da die Abstimmung in der Vergangenheit auch für Das Erste, dessen Programm ja in unterschiedlichem Prozentsatz von den verschiedenen Landesrundfunkanstalten kommt, erfolgreich war, ist eine Koordinierung für den HD-Gemeinschaftskanal sicher auch möglich, wenngleich nicht ohne Konfliktpotential. Das Modell des Gemeinschaftskanals stellt ohnehin nur eine Übergangslösung dar, solange die Beiträge in HDTV nicht ausreichen, um einen Regelbetrieb mit einem HDTV-Vollprogramm für jede einzelne Landesrundfunkanstalt zu rechtfertigen. Im Moment steht dafür noch nicht ausreichend genug an nativen HD-Programminhalten zur Verfügung. Sollte der Umstellungsprozess auf HDTV, der ja gewollt und bewusst schrittweise umgesetzt wird, irgendwann soweit fortgeschritten sein, dass ein Großteil aller Beiträge in HD vorliegt, wird man sich neue Gedanken machen müssen. So lange ist dieses Modell eine gute Lösung, allen Landesrundfunkanstalten die Möglichkeit zu geben, überhaupt schon einmal in HD zu senden. Damit zeigen auch die öffentlich-rechtlichen Sender Präsenz und bekennen sich klar zu einer Einführung von HDTV. Dabei werden dadurch, dass erstmal ‚nur‘ zwei HD-Kanäle betrieben werden, die Kosten möglichst gering gehalten. Anders als mit einer über längere Zeit gestreckten Investition ließe sich die Einführung von HDTV vermutlich gar nicht bewältigen. Zudem eine Erhöhung der Rundfunkgebühren zur Finanzierung der Einführung von HDTV mittelfristig wahrscheinlich nicht durchsetzbar wäre. Allerdings sollten die öffentlich-rechtlichen Sender mit der Einführung von HDTV nicht zu lange warten, sonst werden sie von anderer Seite vor vollendete Tatsachen gestellt (Norm, Format, Features der Endgeräte, etc.) und können nicht mehr agieren, sondern nur noch reagieren. Es gäbe dann für sie keine oder nur noch eingeschränkt die Möglichkeit an der Gestaltung des zukünftigen Fernsehstandards und der Endgeräte beim Zuschauer mitzuwirken.

Der öffentlich-rechtliche Gemeinschaftssender Arte nutzt bereits gelegentlich die beiden Astra HD-Demokanäle zur HDTV-Verbreitung. So wurde im Oktober 2006 die Oper ‚Don Giovanni‘ europaweit in HD ausgestrahlt.

3 Grundlagen der genutzten Konzepte

In diesem Kapitel werden die Grundlagen erläutert, die zum Verständnis dieser Arbeit erforderlich sind. Zunächst werden hierbei die für die Signalisierung erforderlichen Grundlagen vom Transportstrom über die Event Information Table (EIT) bis zum Linkage Descriptor erklärt. Anschließend wird das Prinzip des in dieser Arbeit verwendeten State Diagramms beschrieben sowie ein kurzer Überblick über das Hexadezimalsystem gegeben, das in der Signalisierung von DVB verwendet und bei der Programmierung gebraucht wird.

3.1 DVB-Transportstrom

Die digitalen Video- und Audiodaten liegen am Ausgang der jeweiligen Encoder jeweils in einem Elementarstrom (ES) vor (vgl. Abbildung 3). Der Elementarstrom wird in Pakete unterschiedlicher Länge zerlegt. Die Länge eines Pakets ohne Header beträgt normalerweise maximal 65.535 Byte ($2^{16} - 1$ Byte). Diesen Elementarstrom-Stücken wird ein so genannter, minimal 6 Byte langer, Packet Header vorangestellt (in der Abbildung 3 blau gekennzeichnet). In diesem Packet Header befinden sich unter anderem Angaben über den Paketinhalt, die Paketlänge und die Zeitsynchronisation. Der nun vorliegende Packetized Elementary Stream (PES) setzt sich somit aus einem aufgeteilten (paketierte) Elementarstrom zusammen.

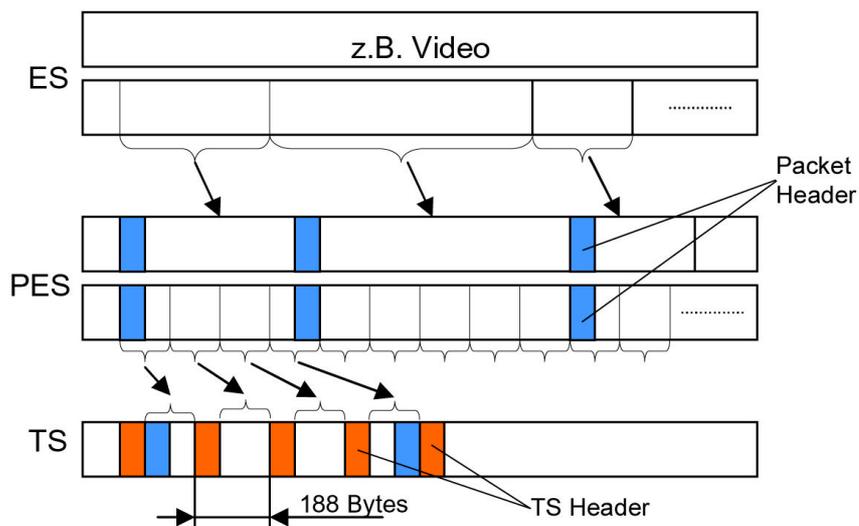


Abbildung 3: Schematischer Aufbau eines Transportstroms

Beim Multiplexen der PES-Pakete für den Transportstrom (TS) werden die PES-Pakete in lauter aufeinander folgende kurze Pakete fester Länge (184 Byte) geteilt und mit einem 4 Byte langem TS-Header (in Abbildung 3 und Abbildung 4 orange dargestellt) versehen. Abbildung 4 zeigt eine Aufschlüsselung des Inhalts des TS-Headers. So stellt z.B. der Continuity Counter (CC) am Ende des Headers die zeitlich richtige Abfolge der Transportpakete sicher. Im Folgenden (Kapitel 3.5) soll aber lediglich die PID, die zur Identifikation der Daten nach Programm, Video, Audio o.ä. dient, näher betrachtet werden.

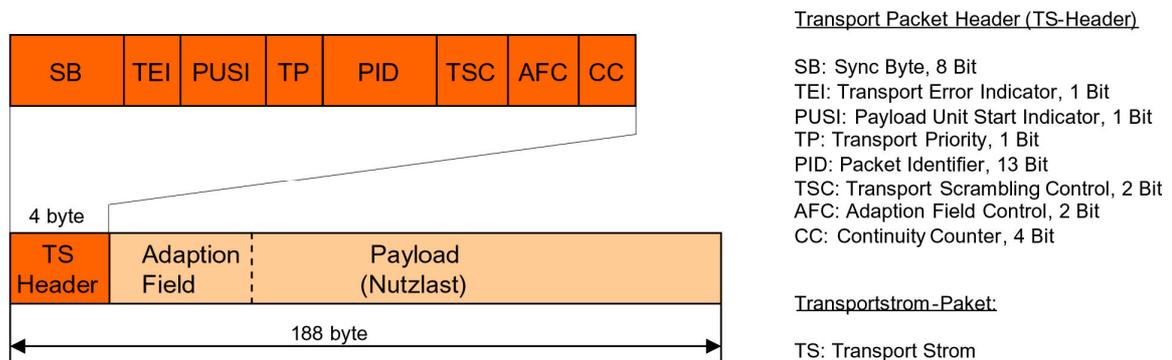


Abbildung 4: Aufschlüsselung des TS-Headers

Audio-, Video- und Daten-PES-Pakete mehrerer Komponenten eines Programms oder auch mehrerer Programme werden dann zu einem Program Stream bzw. Transport Stream gemultiplext (Abbildung 5). Der Programmstrom, auf den im Folgenden nicht weiter eingegangen wird, ist für die Nutzung in relativ fehlerfreien Umgebungen, wie z.B. Studios oder Multimedia-Applikationen, gedacht, der Transportstrom dagegen wird in Umgebungen, in denen Fehler wahrscheinlich sind, wie der Fernsehübertragung, benutzt [11], [12], [13], [14].

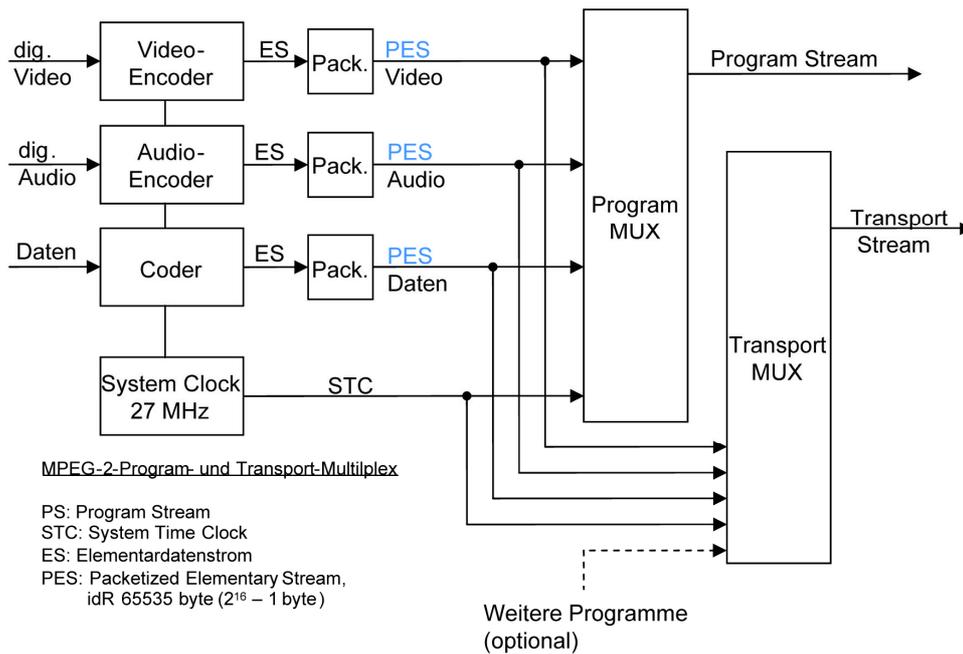


Abbildung 5: MPEG-2 Programm- und Transportstrom-Multiplex (nach [15])

3.2 Das Hexadezimalsystem

Im Hexadezimalsystem werden Zahlen in einem Stellenwertsystem mit der Basis 16 dargestellt.

In der Datenverarbeitung verwendet man häufig dieses System, um die vom Rechner kommenden Dualzahlen in einer für den Menschen übersichtlicheren Form zu notieren, denn eine Bitfolge, bestehend aus lauter Nullen und Einsen, ist in einer Hexadezimaldarstellung leichter zu lesen und schneller zu schreiben.

In dem uns vertrauten Dezimalsystem (10er-System) werden 10 Symbole zur Notation der Ziffern 0-9 verwendet. Im Hexadezimalsystem dagegen sind es 16 Ziffern, die dargestellt werden müssen. Daher verwendet man zusätzlich die Buchstaben A bis F für die Darstellung der fehlenden sechs Ziffern (siehe Tabelle 1). Vier Stellen einer Bitfolge, auch Nibble genannt, entsprechen dann einer Ziffer im Hexadezimalsystem.

Dezimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Hexadezimal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Dual	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111

Tabelle 1: Gegenüberstellung von Dezimal-, Hexadezimal- und Dualsystem (nach [19])

Um eine Hexadezimalzahl von einer Dezimalzahl unterscheiden zu können, wird die Hexadezimalzahl mit einem Präfix oder Suffix versehen. Verbreitete Schreibweisen sind zum Beispiel: 72_{16} , 72_H , $0x72$, "72, 72_h oder $\$72$. In dieser Arbeit werden die Schreibweisen $0x72$ (im Text) und $\$72$ (im Programmcode) verwendet. Ist eine Unterscheidung notwendig, kann eine Dezimalzahl zum Beispiel mit 114_{10} oder 114_D gekennzeichnet werden.

Für eine Umrechnung vom Hexadezimal- ins Dezimalsystem, muss man die einzelnen Ziffern mit der jeweiligen Potenz der Basis multiplizieren. Die Hexadezimalzahl $0x4FE$ entspricht 1278_D und wird berechnet mit $4 \cdot 16^2 + 15 \cdot 16^1 + 14 \cdot 16^0 = 1278_D$.

Eine Möglichkeit zur Umrechnung vom Dezimal- ins Hexadezimalsystem ist die Betrachtung der Divisionsreste bei der Division durch 16:

$$1278 : 16 = 79 \text{ Rest } \mathbf{14} (= E)$$

$$79 : 16 = 4 \text{ Rest } \mathbf{15} (= F)$$

$$4 : 16 = 0 \text{ Rest } \mathbf{4} \rightarrow 0x4FE [19].$$

3.3 Bouquet, Services und Events

In einem Kanal (Kabel, Satellit, terrestrisch) mit einer gewissen Kapazität können mehrere Angebote, die so genannten *Services*, untergebracht werden. Bestimmte Services können zu einem *Bouquet* zusammengefasst werden. So gehören zum Bouquet von ARD Digital 18 Fernsehprogramme: Das Erste, neun regionale Landesprogramme („Dritte Programme“) und BR alpha, Arte, 3sat, Ki.Ka, Phoenix, EinsPlus, EinsExtra, EinsFestival, sowie sämtliche Radioprogramme der Landesrundfunkanstalten und zahlreiche interaktive Dienste [16]. Zum Bouquet von ZDFVision gehören: ZDF, 3sat, Ki.Ka, ZDFinfokanal, ZDFdokukanal und ZDFtheaterkanal sowie die beiden Radioprogramme Dkultur und DLF (siehe Abbildung 6). Die Sender 3sat, Arte, Ki.Ka und Phoenix sind dabei öffentlich-rechtliche Gemeinschaftsprogramme. Deshalb werden über Satellit 3sat und Ki.Ka im Transponder des ZDF übertragen und Arte und Phoenix in dem der ARD.

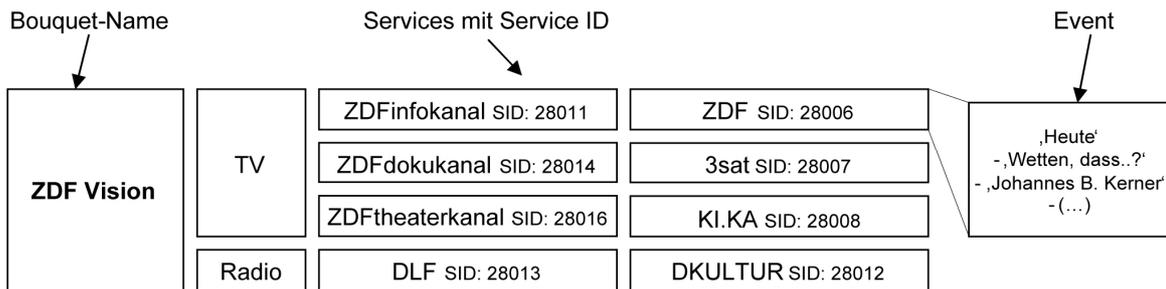


Abbildung 6: Bouquet von ZDF Vision mit zugehöriger Service ID (nach [17])

Ein *Event* ist eine „Sendung“, also ein Einzelbeitrag mit definierter Anfangs- und Endzeit wie z.B. die ‚Tagesschau‘ oder der ‚Tatort‘. Mehrere Events sind zu einem *Program* zusammengestellt.

Darüber hinaus ist *Content* die Bezeichnung für den Inhalt, also das eigentliche Programmmaterial, an dem die Rundfunkanstalt die Verwertungsrechte besitzt [13]. Nach Definition der European Broadcasting Union (EBU) wird die Kombination aus *Essence* (Video- und Audiodaten) und *Metadaten* (Zusatzinformationen wie z.B. Produktionsnummer, Inhaltsangaben, Kameramann, ...) als Content bezeichnet [13].

3.4 Zusatzdaten

Einfache Zusatzdaten, die *Program Specific Information* (PSI), werden bereits durch MPEG-2 definiert, sie ermöglichen dem Empfänger grundsätzlich die Decodierung. Weitere Informationen sind mit der in DVB definierten *Service Information* (SI) möglich.

Die PSI besteht aus vier Tabellen:

- *Program Association Table* (PAT),
- *Program Map Table* (PMT),
- *Network Information Table* (NIT) und
- *Conditional Access Table* (CAT).

Die PAT enthält eine Liste aller im Transportstrom enthaltenen Programme und zeigt auf die PIDs, die die jeweiligen programmspezifischen PMTs enthalten. Die PMT wiederum verweist auf die PIDs der entsprechenden Video-, Audio- und Daten-PES (siehe Abbildung 7). Die PAT selbst hat immer die PID 0.

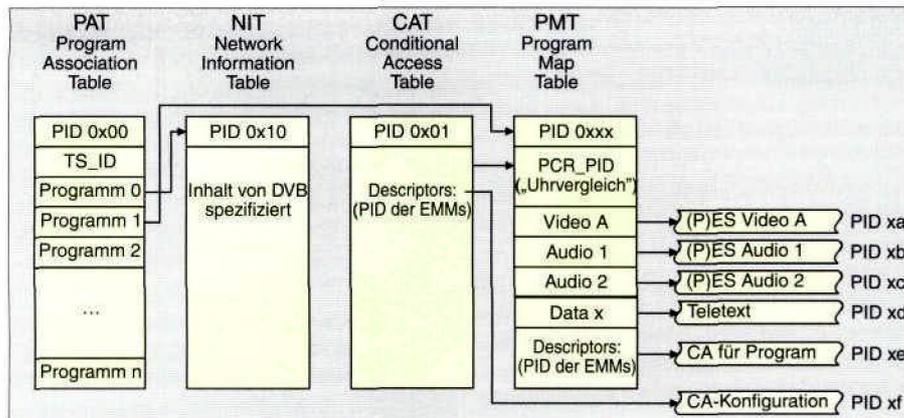


Abbildung 7: Verkettung der PSI-Tabellen [14]

Die CAT enthält Daten zur Zugriffssteuerung und die NIT enthält netzwerkspezifische Daten wie z.B. die Orbitposition oder die Transponder-Nummer [18]. Abbildung 8 zeigt eine tabellarische Übersicht der Tabellen der PSI nach ISO/IEC 13818-1.

Structure Name	Reserved PID#	Description
PAT: Program Association Table	0x00	Associates Program Number and Program Map Table PID
PMT: Program Map Table	Assigned in the PAT	Specifies PID values for components of one or more programs
NIT: Network Information Table	Assigned in the PAT	Physical network parameters such as FDM frequencies, Transponder Number, etc.
CAT: Conditional Access Table	0x01	Associates one or more (private) EMM streams each with a unique PID value

Abbildung 8: Program specific information, (nach [11])

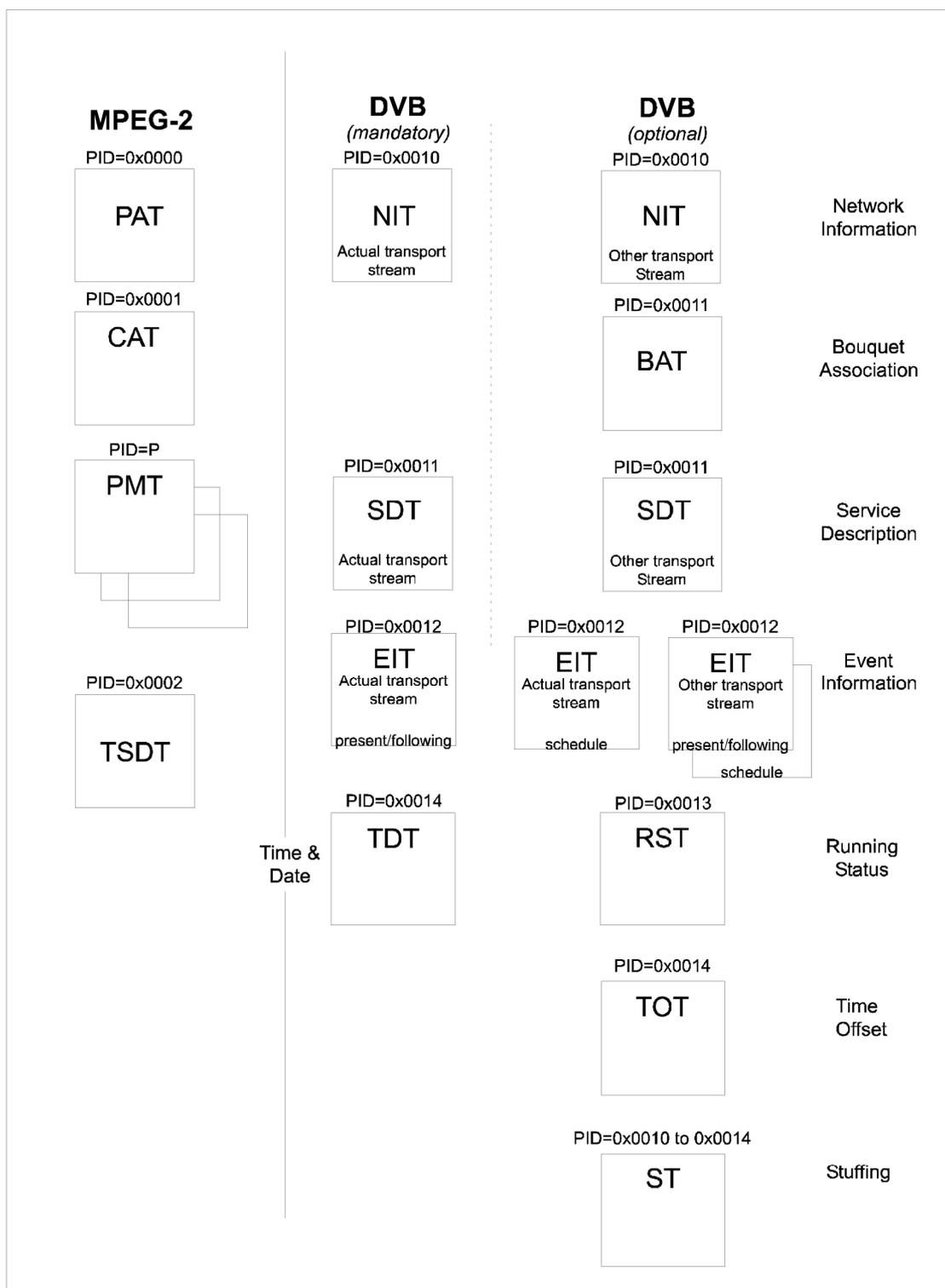
In der SI sind weitere, über die PSI hinausgehende, beschreibende Zusatzinformationen definiert:

- Network Information Table (NIT)
- Bouquet Association Table (BAT)
- Service Description Table (SDT)
- Event Information Table (EIT)
- Running Status Table (RST)
- Time and Date Table (TDT)
- Time Offset Table (TOT)

Die NIT gibt den Namen und den Typ des Netzwerks sowie die technischen Parameter für jeden Transportstrom (Frequenz, Modulation) innerhalb des Netzwerks an. Die BAT gibt den Namen des Bouquets und eine Liste der zugehörigen Services an. In der SDT ist der Name des Services (z.B. Das Erste, ZDF, RTL, Sat1, ...), der Service Provider (RTL → RTL-Gruppe), der Typ des Services (TV, Radio, Daten) sowie Sprache und Verfügbarkeit zu finden. Die EIT beschreibt den Namen des Events (,Tagesschau', ,Wetter', ,Sportschau', ...), Startzeit und Dauer, Beschreibung des Events (Live-Sendung, ...), Inhalts-Klassifikation (Spielfilm, Krimi, ...) sowie Parental Rating (geeignet ab 16, ..ab 12). Die RST gibt den Status des Events an (running, not running, pausing,...). In der TDT steht das aktuelle Datum und die Uhrzeit in UTC (Weltzeit) und in der TOT ist die Abweichung zwischen lokaler Zeit und Weltzeit zu finden [18]. Abbildung 9 zeigt die allgemeine Organisation der Service Information. Hier ist zusammenfassend dargestellt, welche Tabellen in MPEG-2 und welche in DVB definiert sind. Außerdem ist aus dieser Abbildung zu erkennen, welche Tabellen verbindlich (mandatory) und welche wahlweise (optional) als Zusatzdaten im Transportstrom enthalten sein müssen. Ferner ist die PID angegeben, die diesen Tabellen fest zugewiesen ist (siehe auch Tabelle 2, weiter unten). Einige Tabellen, wie z.B. die EIT sind mehrfach vorhanden. Sie kann für den gerade aktuellen Transportstrom (actual) und für einen anderen (other) angegeben werden (vgl. Abbildung 10). Diese verschiedenen EITs werden für die in dieser Arbeit untersuchte Signalisierung genutzt.

MPEG-2 defined

Defined in the present document



NOTE: NIT table entitled "DVB (mandatory)" should read "Actual delivery system" instead of "Actual transport stream".
 NIT table entitled "DVB (optional)" should read "Other delivery system" instead of "Other transport Stream".

Abbildung 9: Allgemeine Organisation der Service Information (SI), [aus EN 300 468]

3.5 Event Information Table (EIT)

Die Event Information Table ist u.a. für die Auswertung der Daten im Demonstrator notwendig. Deshalb wird auf diese Tabelle in diesem Abschnitt detailliert eingegangen.

Wie bereits in Kapitel 3.4 erwähnt, befinden sich in der EIT die Angaben über:

- Beginn und Dauer eines Beitrags
- Titel und Kurzbeschreibung
- Ausführliche Beschreibung
- Signalisierung von ‚baldiger Beginn‘, ‚laufend‘, ‚Pause‘
- Typ der Komponenten (Text, Audio, TV, Stereo, Mono, Untertitel, Bildseitenverhältnis)
- Klassifizierung nach Genre
- Altersbeschränkung
- Kennung eines benutzten Conditional Access-Systems, Verweise, ... [14]

Die EIT ist durch die Kennzeichnung mit der PID 18_D bzw. 0x0012 im Transportstrom eindeutig auffindbar. Die anderen Tabellen haben ebenfalls eine fest zugewiesene PID (siehe auch Tabelle 2).

Table	PID value
PAT	0x0000
CAT	0x0001
TSDT	0x0002
reserved	0x0003 to 0x000F
NIT, ST	0x0010
SDT, BAT, ST	0x0011
EIT, ST	0x0012
RST, ST	0x0013
TDT, TOT, ST	0x0014
network synchronization	0x0015
reserved for future use	0x0016 to 0x001D
DIT	0x001E
SIT	0x001F

Tabelle 2: PID-Zuordnung für die SI, aus [20]

Es gibt vier Typen von EITs, die durch ihre unterschiedlichen Table IDs differenziert werden können:

- table_id 0x4E (für actual TS, present/following event information)
- table_id 0x4F (für other TS, present/following event information)

- table_id 0x50 bis 0x5F (für actual TS, event schedule information)
- table_id 0x60 bis 0x6F (für other TS, event schedule information)

Dabei gibt es eine physikalisch/örtliche Unterscheidung zwischen dem gegenwärtigen und einem anderen Transportstrom (*actual/other*) sowie eine zeitliche Unterscheidung zwischen momentan laufendem Event und unmittelbar anschließendem Event (*present/following*) (vgl. Abbildung 10), wobei present und following lediglich durch die in der EIT angegebene Section Number (*section_number*) (siehe Abbildung 11) unterschieden wird. Dabei steht die Section Number 0 für das momentan laufende Event und die Section Number 1 für das folgende Event. Darüber hinaus gibt es noch die beiden EIT-Typen für den Sendeplan (*schedule*) des gegenwärtigen und eines anderen Transportstroms. Diese Schedule-Tabellen können z.B. für die Anzeige eines vollständigen EPG (Electronic Program Guide), der das Programm der nächsten Tage oder sogar Wochen anzeigt, im Empfänger genutzt werden.

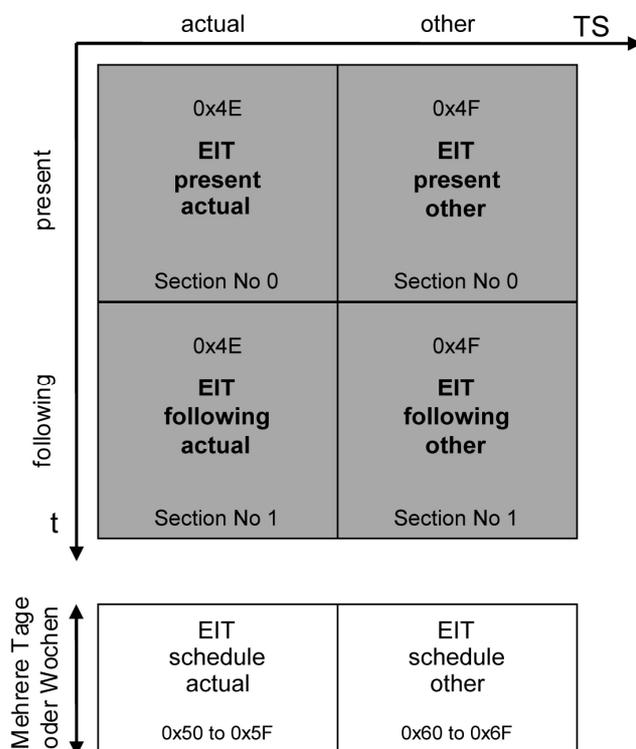


Abbildung 10: Grafische Übersicht über die unterschiedlichen EIT-Typen

Die formale Struktur, die die EIT haben muss, ist in der Norm EN 300 468 festgelegt. Dort sind die verschiedenen Informationen chronologisch und mit der

Anzahl Bits, die zur Beschreibung der jeweiligen Information zur Verfügung stehen, angegeben.

Innerhalb der EIT befinden sich zwei ineinander verschachtelte for-Schleifen. Einmal die Descriptor-Schleife (in Abbildung 11, mit blauem Rahmen markiert), in der der Linkage Descriptor, der im nächsten Kapitel beschrieben wird, zu finden ist, und einmal die Event ID-Schleife (in Abbildung 11, mit rotem Rahmen). Die Länge der Schleifen wird nicht durch die Norm bestimmt, sondern steht in dem Wert für die `descriptors_loop_length` (für die Descriptor-Schleife) bzw. lässt sich aus der `section_length` errechnen (für die Event ID-Schleife). Diese Tatsache war bei der Implementierung des Empfänger-Demonstrators von Bedeutung, da in dem Programm erstmal die benötigten Informationen herausgefiltert werden mussten. Darauf wird im Kapitel 5 weiter eingegangen.

Syntax	No. of Bits	Identifier																																								
<code>event_information_section(){</code>																																										
<table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td><code>table_id</code></td> <td>8</td> <td>uimbsf</td> </tr> </table>	<code>table_id</code>	8	uimbsf	8	uimbsf																																					
<code>table_id</code>	8	uimbsf																																								
<code>section_syntax_indicator</code>	1	bslbf																																								
<code>reserved_future_use</code>	1	bslbf																																								
<code>reserved</code>	2	bslbf																																								
<code>section_length</code>	12	uimbsf																																								
<table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td><code>service_id</code></td> <td>16</td> <td>uimbsf</td> </tr> </table>	<code>service_id</code>	16	uimbsf	16	uimbsf																																					
<code>service_id</code>	16	uimbsf																																								
<code>reserved</code>	2	bslbf																																								
<code>version_number</code>	5	uimbsf																																								
<code>current_next_indicator</code>	1	bslbf																																								
<table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td><code>section_number</code></td> <td>8</td> <td>uimbsf</td> </tr> </table>	<code>section_number</code>	8	uimbsf	8	uimbsf																																					
<code>section_number</code>	8	uimbsf																																								
<code>last_section_number</code>	8	uimbsf																																								
<table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td><code>transport_stream_id</code></td> <td>16</td> <td>uimbsf</td> </tr> </table>	<code>transport_stream_id</code>	16	uimbsf	16	uimbsf																																					
<code>transport_stream_id</code>	16	uimbsf																																								
<table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td><code>original_network_id</code></td> <td>16</td> <td>uimbsf</td> </tr> </table>	<code>original_network_id</code>	16	uimbsf	16	uimbsf																																					
<code>original_network_id</code>	16	uimbsf																																								
<code>segment_last_section_number</code>	8	uimbsf																																								
<code>last_table_id</code>	8	uimbsf																																								
<table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td><code>for(i=0;i<N;i++){</code></td> <td>16</td> <td>uimbsf</td> </tr> <tr> <td> <code>event_id</code></td> <td></td> <td></td> </tr> <tr> <td> <code>start_time</code></td> <td>40</td> <td>bslbf</td> </tr> <tr> <td> <code>duration</code></td> <td>24</td> <td>uimbsf</td> </tr> <tr> <td> <code>running_status</code></td> <td>3</td> <td>uimbsf</td> </tr> <tr> <td> <code>free_CA_mode</code></td> <td>1</td> <td>bslbf</td> </tr> <tr> <td> <code>descriptors_loop_length</code></td> <td>12</td> <td>uimbsf</td> </tr> <tr> <td> <table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td><code>for(i=0;i<N;i++){</code></td> <td></td> <td></td> </tr> <tr> <td> <code>descriptor()</code></td> <td></td> <td></td> </tr> <tr> <td> }</td> <td></td> <td></td> </tr> </table></td> <td></td> <td></td> </tr> <tr> <td> }</td> <td></td> <td></td> </tr> <tr> <td> <code>CRC_32</code></td> <td>32</td> <td>rpchof</td> </tr> <tr> <td>}</td> <td></td> <td></td> </tr> </table>	<code>for(i=0;i<N;i++){</code>	16	uimbsf	<code>event_id</code>			<code>start_time</code>	40	bslbf	<code>duration</code>	24	uimbsf	<code>running_status</code>	3	uimbsf	<code>free_CA_mode</code>	1	bslbf	<code>descriptors_loop_length</code>	12	uimbsf	<table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td><code>for(i=0;i<N;i++){</code></td> <td></td> <td></td> </tr> <tr> <td> <code>descriptor()</code></td> <td></td> <td></td> </tr> <tr> <td> }</td> <td></td> <td></td> </tr> </table>	<code>for(i=0;i<N;i++){</code>			<code>descriptor()</code>			}					}			<code>CRC_32</code>	32	rpchof	}		
<code>for(i=0;i<N;i++){</code>	16	uimbsf																																								
<code>event_id</code>																																										
<code>start_time</code>	40	bslbf																																								
<code>duration</code>	24	uimbsf																																								
<code>running_status</code>	3	uimbsf																																								
<code>free_CA_mode</code>	1	bslbf																																								
<code>descriptors_loop_length</code>	12	uimbsf																																								
<table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td><code>for(i=0;i<N;i++){</code></td> <td></td> <td></td> </tr> <tr> <td> <code>descriptor()</code></td> <td></td> <td></td> </tr> <tr> <td> }</td> <td></td> <td></td> </tr> </table>	<code>for(i=0;i<N;i++){</code>			<code>descriptor()</code>			}																																			
<code>for(i=0;i<N;i++){</code>																																										
<code>descriptor()</code>																																										
}																																										
}																																										
<code>CRC_32</code>	32	rpchof																																								
}																																										

Abbildung 11: Syntax zur Event information section, nach [20]

Abbildung 12 zeigt eine EIT als Hex Dump. Ein Zeichen-Pärchen stellt immer 1 Byte (8 Bit) dar. Die Hexadezimalzahl 4E (hier Magenta) ist die bereits weiter oben erwähnte Table ID und gibt hier an, dass die vorliegende EIT vom gegenwärtigen

Transportstrom, *present/following* ist. Bei weiterer Betrachtung der Section Number (01, hier grün markiert) erkennt man, dass es sich hier um eine *following* handelt.

Auch das so genannte DVB-Triple, bestehend aus der Original Network ID, der Transportstrom ID und der Service ID, ist in diesem Hex Dump zu finden. 0x6DDA (28122_D) ist hier die Service ID (hr-info), 0x044D (1101_D) die Transport Strom ID und 0x0001 (1_D) die Original Network ID. Mit diesen drei Werten ist jeder Transportstrom weltweit eindeutig beschrieben, denn die Original Network ID ist bei DVB in der ETSI-Norm ETR 162 registriert.

	1				5					10				15	16	
0000	4E	F0	4B	6D	DA	DF	01	01	04	4D	00	01	01	4E	84	41
0010	D0	C3	11	42	00	00	06	00	20	30	4D	0C	64	65	75	07
0020	05	48	65	73	73	65	6E	00	50	06	F2	03	02	64	65	75
0030	69	03	F7	1B	2A	5F	04	00	00	00	05	82	0D	31	32	3A
0040	34	32	31	34	2E	30	33	23	30	30	EB	18	DD	75		

Abbildung 12: Hex Dump einer EIT actual following

3.6 Linkage Descriptor

Wie auch in den meisten anderen Tabellen, kann in der EIT ein so genannter *Descriptor* vorkommen (vgl. Abbildung 11, blauer Rahmen). Es gibt viele verschiedene Descriptoren, so z.B. einen *Content Descriptor*, der eine Eingruppierung des jeweiligen Events in ein Genre (Movie, Sports, ...) ermöglicht oder den *Service Descriptor*, der den Namen des Service Providers (ARD, ZDF, ...) sowie dessen *Service Type* (digital television service, digital radio sound service, Teletext service, ...) angibt. Für diese Arbeit ist der so genannte *Linkage Descriptor* relevant, deshalb werden die anderen Descriptoren nicht näher betrachtet.

Der Linkage Descriptor wird benutzt, um einen Link zu einem anderen Service anzugeben. Wenn dieser Descriptor in der Descriptor-Schleife auftaucht, verlinkt er auf einen Service, dessen DVB-Triple im Linkage Descriptor angegeben ist (siehe Abbildung 13, gelbe Markierung).

Alle Descriptoren werden durch eine zweistellige Hexadezimalzahl, den *Descriptor Tag* (in Abbildung 13 durch einen blauen Rahmen markiert), eindeutig gekennzeichnet. Der Wert des Descriptor Tags für den Linkage Descriptor ist die

Hexadezimalzahl 0x4A (74_D). Mit dieser Zahl ist der Linkage Descriptor unter den anderen Descriptoren auffindbar.

Syntax	No. of bits	Identifier
linkage_descriptor(){		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
transport_stream_id	16	uimsbf
original_network_id	16	uimsbf
service_id	16	uimsbf
linkage_type	8	uimsbf
for (i=0;i<N;i++){		
private_data_byte	8	bslbf
}		
}		

Abbildung 13: Syntax-Tabelle zum Linkage Descriptor (nach [20])

Der so genannte *Linkage Type* gibt die Art der Information an, auf die ‚verlinkt‘ werden soll. Für die Kennzeichnung von HD und SD-Events sollen zukünftig aus dem für die spätere Nutzung vorgesehenen Bereich (in Tabelle 3, gelb markiert) zwei Linkage Types vergeben werden, die auf einen HD-Service bzw. einen SD-Service verweisen.

Linkage_type	Description
0x00	reserved for future use
0x01	information service
0x02	EPG service
0x03	CA replacement service
0x04	TS containing complete Network/Bouquet SI
0x05	service replacement service
0x06	data broadcast service
0x07 to 0x7F	reserved for future use
0x80 to 0xFE	user defined
0xFF	reserved for future use

Tabelle 3: Linkage type coding (nach [20])

So gäbe es dann einen Linkage Type, der im SD-Service vorkommt, und auf den zugehörigen HD-Content verweist. Im entsprechenden HD-Service gäbe es dann einen Linkage Type, der wieder zurück auf den SD-Service verweist. Weitere Details werden im Kapitel 4.1 erklärt.

Abbildung 14 zeigt eine mit dem TS-Analyser von Tektronix erstellte grafische Darstellung eines Transportstroms, der im Empfänger-Demonstrator verwendet wurde. Dieser TS-Analyser wertet einen Hex Dump ähnlich dem in Abbildung 12

entsprechend aus und bereitet ihn grafisch auf. Gut zu erkennen ist, dass in diesem TS drei Descriptoren vorkommen: Der Short Event Descriptor, der Component Descriptor und der Linkage Descriptor. Die in der Syntax-Tabelle (Abbildung 13) aufgeführten Werte sind hier noch mal grafisch dargestellt.

SI/PSI Table View for table_id = 78 [0x4e EIT]

TableHeader	table id	section syntax indicator	section length	service id	version number	current next indicator	section number	last section number
	78 [0x4e EIT]	1	61	555 [0x22b]	1 [0x1]	1	0 [0x0]	1 [0x1]

transport stream id	original network id	segment last section number	last table id
9999 [0x270f]	1 [0x1]	1	78 [0x4e]

event information

event id	event id	start time	duration	running status	free CA mode	descriptors loop length
9998 [0x270e]	9998 [0x270e]	2006/8/21 0:00:00 [0xd2d0000000]	23:00:00	4 (running)	0	34

descriptors

short event descriptor	descriptor tag	descriptor length	ISO 639 language code	event name length	event name	text length	text
	77 [0x4d]	7	DEU	2	P0	0	

component descriptor	descriptor tag	descriptor length	stream content component type	component tag	ISO 639 language code	text
	80 [0x50]	12	267 (reserved for future use)	1	DEU	HVideo

linkage descriptor	descriptor tag	descriptor length	transport stream id	original network id	service id	linkage type
	74 [0x4a]	9	9999 [0x270f]	1 [0x1]	556 [0x22c]	11 (reserved for future use)

private data

private data byte	private data byte
100 [0x64]	100 [0x64]

private data byte	private data byte
101 [0x65]	101 [0x65]

CRC 32
1325933933 [0x4f08256d]

Abbildung 14: Grafische Darstellung einer EIT mit dem TS-Analyser von Tektronix

3.7 Finite State Machine

Die Finite State Machine (auch ‚endlicher Automat‘ oder ‚Zustandsmaschine‘) ist ein Modell, das ein Programmverhalten bestehend aus Zuständen, Zustandsübergängen und Aktionen visualisieren kann [21]. Im so genannten Moore-Modell wird das Verhalten nur mit kreisförmig dargestellten Zuständen (*states*) und Zustandsübergängen (*transitions*), die durch pfeilförmige Verbindungslinien gekennzeichnet werden, dargestellt. Der Pfeil gibt dabei die Richtung vor, in der eine Aktion ausgeführt werden kann, er ist sozusagen eine ‚Einbahnstraße‘.

Am Beispiel eines automatischen Fensters in einem Gewächshaus verdeutlicht Abbildung 15 das Prinzip eines State-Diagramms nach dem Moore Modell. Für das Fenster gibt es zwei Zustände: Offen (Zustand 1) oder geschlossen (Zustand 2). Außerdem gibt es zwei Bedingungen für den Zustandsübergang (*transition condition*): a ist die Bedingung von ‚Offen‘ nach ‚Geschlossen‘ und b die Bedingung von Zustand 2 nach Zustand 1. Dabei könnte die Bedingung a möglicherweise lauten: Die Temperatur im Gewächshaus ist kleiner/gleich 20°C. Die Bedingung b lautet dann entsprechend: Die Temperatur im Gewächshaus ist größer als 20°C.

Das Verhalten der Steuerung für das Gewächshausfenster lässt sich aus Abbildung 15 gut ablesen. Ist das Fenster offen (Zustand 1) und die Temperatur fällt unter 20°C, dann wäre die Bedingung a wahr und das Fenster würde schließen (Wechsel in den Zustand 2). Würde die Temperatur hier wiederum über 20°C steigen, würde sich das Fenster wieder öffnen, usw.

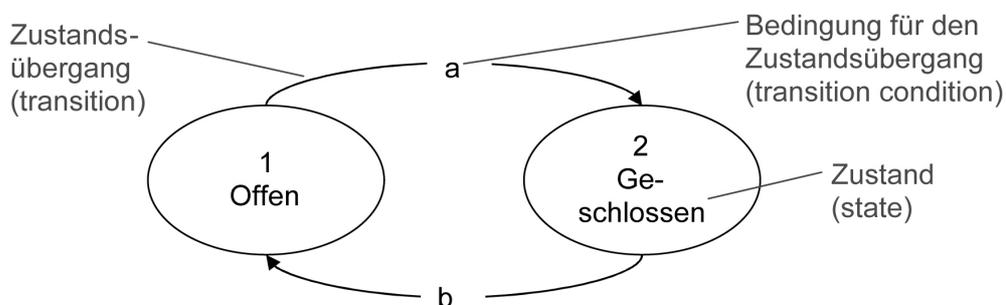


Abbildung 15: Vereinfachtes State-Diagramm (nach [20])

4 Untersuchte Verfahren und erweiterte Konzepte

Bei einer Ankündigung wie „Die nun folgende Sendung wird Ihnen präsentiert in HDTV“ erwartet der technisch weniger versierte Zuschauer eine Sendung in höherer Auflösung und wundert sich, dass er keinen Unterschied zu der Qualität der vorhergehenden Sendung erkennt. Dem Zuschauer, der noch gar kein HD-fähiges Equipment besitzt, kann man vielleicht noch erklären, dass das mit den bei ihm vorhandenen Geräten nicht geht, aber bei demjenigen Zuschauer, der einen HD-ready Fernseher sowie eine HD-fähige Settop-Box sein eigen nennt und einen entsprechenden Empfang von HD-Fernsehsignalen hat, wird das schon schwieriger. Dieser Zuschauer erwartet, dass auf seinem Display automatisch das höherwertige Signal dargestellt wird, wenn ein solches vorhanden ist. Ähnlich wie die Umschaltung der Bildseitenverhältnisse 4:3 und 16:9, die automatisch erfolgt (oder erfolgen sollte), sollte die Umschaltung auf einen HDTV-Kanal entsprechend automatisch passieren.

4.1 IRT-Proposal für eine Erweiterung von EN 300 468 und TR 101 211

Im IRT wurde im September 2005 ein Vorschlag (Original siehe Anhang) zur Erweiterung von EN 300 468 und TR 101 211 erarbeitet. Dieses Dokument schlägt die Nutzung des existierenden Linkage-Descriptors (siehe Erklärung im Kapitel 3.6) mit der Einführung neuer, zusätzlicher Linkage Types zur Kennzeichnung von HD- und SD-Inhalten in der Service Information (SI) des DVB-Standards vor.

Die Linkage Types sollen laut dem Vorschlag durch die Hexadezimalzahlen 0x0B (11_D) und 0x0C (12_D) gekennzeichnet werden. Diese Zahlen waren im damals gültigen Standard noch nicht belegt bzw. mit ‚reserved for future use‘ gekennzeichnet.

Wenn ein Event gleichzeitig auf einem SD- und einem HD-Service ausgestrahlt wird, befindet sich der Linkage Type 0x0B im SD-Service und verweist auf ein, vom Inhalt her, identisches Event im HD-Service. 0x0C dagegen verweist im HD-Service auf den zugehörigen SD-Service (vgl. Umschaltvorgang zwischen Service A und B in Abbildung 16).

Ein HD-fähiger Receiver würde in Zukunft diese Linkage Types erkennen und verarbeiten, ein Receiver, der nicht HD-fähig ist, würde dagegen diese Werte einfach ignorieren. Somit ist auch die Kompatibilität gewährleistet, denn für den Zuschauer ohne HD-Equipment ist die Information, dass ein HD-Content auf einem speziellen HD-Kanal vorliegt, irrelevant, weil er sie nicht nutzen kann.

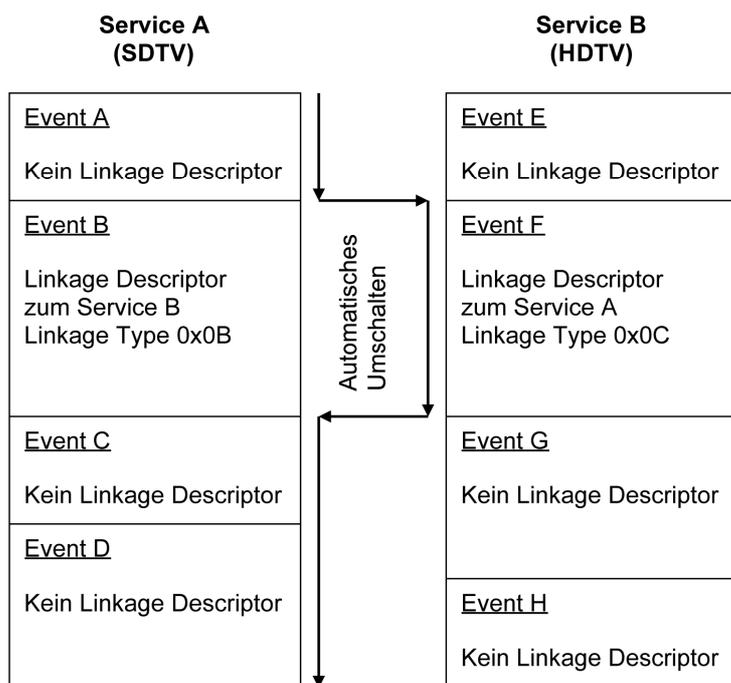


Abbildung 16: HD/SD-Umschaltvorgang [nach IRT-Proposal für TR 101 211]

Nach dem IRT-Proposal soll der aktuelle SD-Kanal nach dem Linkage Type 0x0B durchsucht werden. Wird er gefunden, soll überprüft werden, ob das dazugehörige Event im SD-Kanal auch als „present“ gekennzeichnet ist. Es muss folglich die EIT für den ‚actual‘ TS und der ‚present/following‘ Event Information, also die EIT mit der Table ID 0x4E, nach dem Linkage Type durchsucht werden. Wird der ‚richtige‘ Linkage Type 0x0B gefunden, soll der Receiver automatisch auf den im Linkage Descriptor angezeigten HD-Kanal umschalten. Dabei soll er sich aber den Service merken, der den Umschaltvorgang ausgelöst hat (Herkunftskanal). Im HD-Kanal sollte dann entsprechend in der EIT mit der Table ID 0x4E der Linkage Type 0x0C vorhanden sein. Die Interpretation dieses Linkage Types sei nur zulässig, wenn ein oben beschriebener Umschaltvorgang vorher stattgefunden hat. Solange der Linkage Type 0x0C in der EIT present vorhanden ist, soll der Receiver auf dem derzeitigen HD-Kanal bleiben. Sobald in einer neuen EIT present der Verweis auf

den Herkunftskanal nicht mehr vorhanden ist, soll der Receiver automatisch auf den Herkunftskanal zurückschalten.

Ist der Link vom HD- auf den SD-Kanal (Linkage Type 0x0C) nach dem Schalten auf den HD-Kanal dort nicht sofort vorhanden - was der Fall sein könnte, wenn z.B. die EITs im SD und HD-Kanal nicht absolut synchron sind - soll der Receiver nicht zurückschalten, sondern auf dem HD-Kanal bleiben, bis der Linkage Type 0x0C auftaucht und anschließend verschwindet.

In Abbildung 17 ist die Darstellung aus dem IRT-Proposal zu sehen: In Event J befindet sich ein Linkage Descriptor, der auf Service B verweist. Dort, in Service B, wird das ‚Gegenstück‘, der Linkage Descriptor zurück zum Service C, gefunden und erst wenn dieser Linkage Descriptor wieder verschwindet, also am Ende von Event G, erfolgt die Schaltung zurück auf Service C.

Dieser Vorschlag ist für die Veröffentlichung in einem Standard noch etwas unpräzise und lässt verschiedene Interpretationen zu. Es ist potenziellen Nutzern eines solchen Standards ggf. nicht ganz klar, welchen Sinn dieser Ablauf in der praktischen Anwendung machen soll, da die Programmschemata von Service B und C nicht sinnvoll zusammenpassen. Unverständlich wäre dann, warum zu Beginn von Event J mitten in das bereits laufende Event F hineingeschaltet werden soll und der Receiver dann bis zum Ende von Event G (also über die ‚Eventgrenze‘ hinaus) auf dem HD-Service bleibt, wo es sich doch offenbar in Service C um ein Event, in Service B aber um $1 \frac{1}{2}$ Events ($\frac{1}{2}$ wegen des unvollständigen Anfangs) handelt.

Event F könnte ein Beispiel dafür sein, dass es möglich ist, mehrere Linkage Descriptoren in einem Event unterzubringen.

Unter anderem ist es Ziel dieser Arbeit, eine möglichst genaue und eindeutige Beschreibung des von den Endgeräteherstellern zu implementierenden Verhaltens zu entwickeln, sowie dafür geeignete Wege zur Dokumentation auszuwählen. Aus diesem Grund wird der Ablauf im folgenden Kapitel noch mal in einem State Diagramm beschrieben.

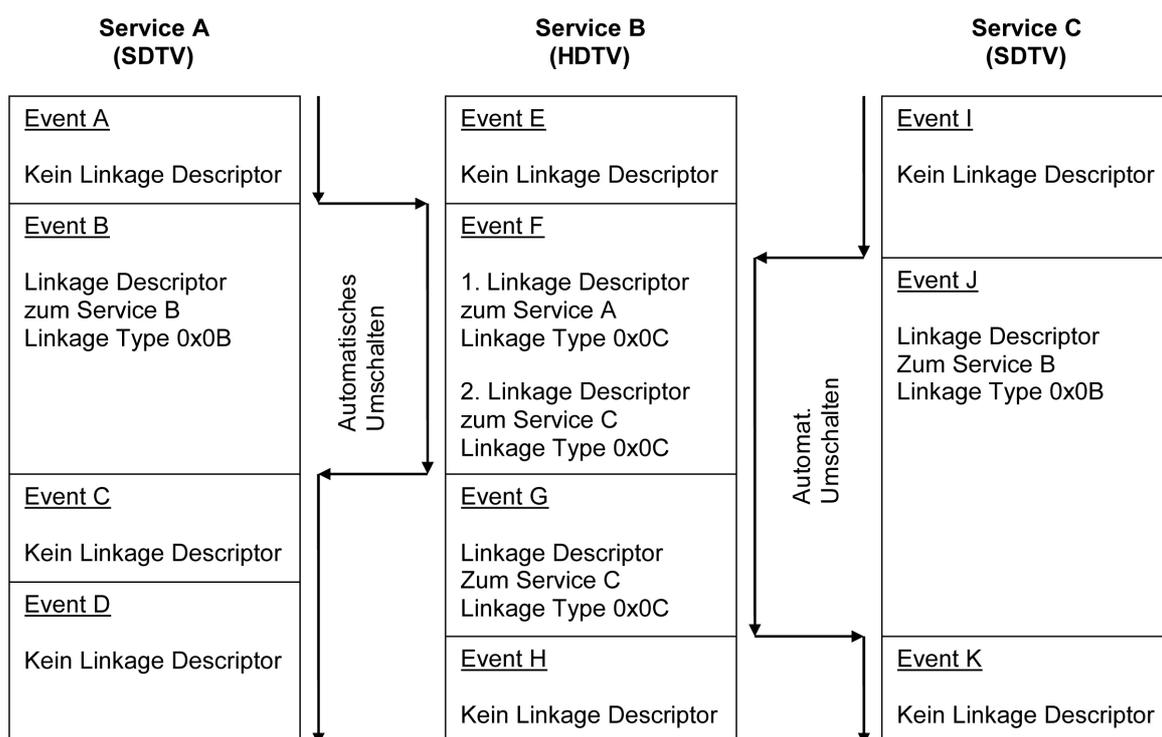


Abbildung 17: HD/SD-Umschaltvorgang aus IRT-Proposal für TR 101 211

4.1.1 Abbildung des Proposals in einem State-Diagramm

Im Laufe der Durchführung der Arbeit zeigte sich, dass Algorithmen am Besten durch State-Diagramme (vgl. Kapitel 3.7) beschrieben werden. In dem unten abgebildeten State-Diagramm (Abbildung 18) ist der Ablauf der Schaltung von SD nach HD und wieder zurück gemäß des IRT-Vorschlags (vgl. Kapitel 4.1) dargestellt.

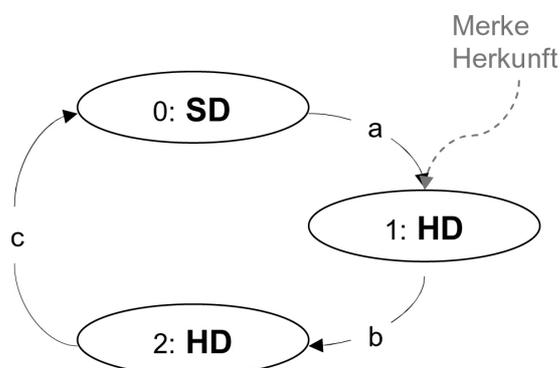


Abbildung 18: State Diagramm nach dem IRT-Proposal

Bedingung a	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	LinkageType muss vorhanden sein und er muss 0x0B sein
Bedingung b	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	LinkageType 0x0C muss vorhanden sein
Bedingung c	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	Der LinkageType 0x0C muss verschwinden (mit einem LinkageDvbTriple auf den Herkunftskanal)

Tabelle 4: Liste der Bedingungen a bis c laut IRT-Proposal

Befindet sich der Receiver auf einem SD-Kanal (Status 0 in der Abbildung 18) und wird in diesem aktuellen SD-Kanal der Linkage Type 0x0B gefunden und ist das zugehörige Event zusätzlich noch als ‚present‘ gekennzeichnet, so ist die Bedingung a erfüllt und der Receiver soll auf den HD-Kanal (Status 1 in der Abbildung 18) schalten.

Dabei soll er sich merken, von welchem SD-Kanal er gekommen ist, da er später wieder dorthin zurückschalten muss.

Befindet sich der Receiver nun auf dem HD-Kanal (Status 1), so soll er dort warten bis der Linkage Type 0x0C auftaucht (Bedingung b) und dann auf Status 2 umschalten.

Verschwindet der Linkage Type 0x0C wieder, soll der Receiver wieder auf den SD-Kanal schalten (von Status 2 nach Status 0).

Bei der Erstellung des Empfänger-Demonstrators war zunächst der Vorschlag zur Erweiterung von EN 300 468 und TR 101 211 der Ausgangspunkt, von dem aus begonnen wurde, das Konzept zu implementieren.

Es stellte sich jedoch im Laufe der Arbeit heraus, dass der Vorschlag präzisiert werden muss, um daraus einen lauffähigen Demonstrator zu entwickeln. Ein erweitertes Konzept wird deshalb im nächsten Abschnitt vorgestellt.

4.1.2 Erweitertes Konzept für das Proposal

Der Vorschlag vom IRT ist zwar richtig, aber an einigen Stellen ist der doch recht komplizierte Sachverhalt so ausgedrückt, dass der Vorschlag Spielraum für

Interpretationen zulässt. Das ist für eine Norm bzw. Technische Richtlinie denkbar ungünstig. Einige Ungenauigkeiten sind erst im Laufe der Programmierung des Empfänger-Demonstrators und den damit verbundenen Überlegungen aufgefallen. Die Bedingungen, die anfangs nicht ganz klar waren, sind hier nun genauer beschrieben:

- Die Bedingung b darf nur dann wahr sein, wenn der Linkage Type 0x0C gefunden wird und zusätzlich mit seinem DVB-Triple auf den Herkunftskanal verweist. Denn wenn der Linkage Type 0x0C zwar vorhanden ist, aber beispielsweise von einem vorangegangenen Event mit ganz anderem Herkunftskanal ‚übrig geblieben‘ ist, soll der Demonstrator darauf nicht reagieren, sondern auf den Linkage Type 0x0C mit dem richtigen Herkunftskanal horchen.
- Ferner muss die Bedingung c wahr werden, wenn kein Linkage Type 0x0C mehr gefunden wird oder wenn das Herkunfts-DVB-Triple sich ändert. Denn wenn auf dem HD-Kanal im direkten Anschluss an das aktuelle HD-Event ein weiteres HD-Event gesendet wird, das aber von einem ganz anderen Herkunftskanal kommt, soll die Schaltung zurück auf Status 0 trotzdem erfolgen. Der Linkage Type 0x0C wäre dann zwar noch vorhanden, hätte aber einen Verweis auf einen ganz anderen SD-Kanal.

Die präzisierten Bedingungen im Überblick:

Bedingung a	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	LinkageType muss vorhanden sein und er muss 0x0B sein
Bedingung b	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	LinkageType 0x0C muss <i>mit einem LinkageDvbTriple auf den Herkunftskanal</i> vorhanden sein
Bedingung c	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	Der LinkageType 0x0C muss verschwinden <i>oder</i> LinkageDvbTriple auf den Herkunftskanal muss verschwinden

Tabelle 5: Liste der Bedingungen a bis c nach dem erweiterten IRT-Proposal

4.2 Ergänzungen zum IRT-Proposal

Mit den im vorherigen Abschnitt genannten Ergänzungen würde der Empfänger-Demonstrator von EN 300 468 bzw. TR 101 211 funktionieren. Jedoch gibt es noch weitere Punkte zu berücksichtigen.

- Wenn sich der Receiver im Status 1 befindet und der ‚richtige‘ Linkage Type 0x0C aus irgendeinem Grund nicht vorhanden ist, würde die Rückschaltung auf den HD-Kanal nie erfolgen und der Receiver würde ohne Eingreifen des Zuschauers immer auf dem HD-Kanal bleiben. Deshalb wurde im Demonstrator ein Timeout eingebaut, das nach einer bestimmten Zeit, in der der Linkage Type 0x0C nicht gefunden wird, auf den Herkunfts-SD-Kanal zurückschaltet. (siehe Kapitel 5.3.1)
- Eine weitere Möglichkeit wäre, dass der Receiver im Status 1 die EIT-present und die EIT-following nach dem Linkage Type durchsucht. Findet er in beiden Tabellen den Linkage Type 0x0C nicht, soll er sofort, d.h. ohne Timeout-Wartezeit auf den SD-Kanal zurückschalten. Doch welche Information ist in der EIT-following des HD-Kanals zu finden? Die Information des SD-Kanals oder die des HD-Kanals (siehe auch Kapitel 4.3)? Ist die Information in der EIT-following die des nächsten Events im SD-Kanal, dann ist diese Lösung so nicht realisierbar. Daher wurde diese Variante auch nicht für die Realisierung im Demonstrator ausgewählt.

4.3 EPG im HD-Kanal

Würde ein HD-Gemeinschaftskanal für die Dritten Programme eingeführt werden, müsste man sich auch Gedanken darüber machen, welche Zusatzinformationen dem Zuschauer angezeigt werden sollen, wenn er sich auf dem HD-Kanal befindet. Was passiert, wenn der Zuschauer auf dem HD-Kanal die Videotext-, die EPG- oder die Infotaste drückt? Passiert nichts, also ist die Taste inaktiv? Oder wird die Information des HD-Kanals, der aber von verschiedenen Rundfunkanstalten gespeist wird, angezeigt? Oder wird die Information des Herkunftskanals angezeigt?

Eine Taste, die nicht funktioniert, nur weil man sich auf dem HD-Kanal befindet, ist unlogisch und würde zudem den Zuschauer verärgern. Diese Möglichkeit scheidet

also aus. Die Anwendung einer der beiden anderen Möglichkeiten hängt von der Philosophie der Betreiber ab. Versteht man den Kanal als eigenständigen HD-Kanal, muss auch die Information des HD-Kanals angezeigt werden.

Der in Kapitel 2.3 beschriebene Ansatz und der entwickelte Empfänger-Demonstrator gehen davon aus, dass der Zuschauer sich vom Prinzip her für einen SD-Kanal entscheidet und falls eine Sendung dieses SD-Kanals auch in HD auf dem HD-Kanal verfügbar ist, der Receiver automatisch umschaltet. Der Zuschauer würde sich also unter Umständen gar nicht bewusst sein, dass er sich auf einem anderen Kanal befindet. Deshalb sollten sich die anwählbaren Zusatzinformationen in diesem Fall auch auf den Herkunfts kanal beziehen.

In einem zweiten Fall wurde der HD-Gemeinschaftskanal direkt angewählt. Hier möchte der Zuschauer beim Drücken der EPG-Taste auf der Fernbedienung auch wissen, was als nächstes auf dem HD-Kanal kommt, unabhängig davon, von welcher Rundfunkanstalt das HD-Event ursprünglich kommt.

Es muss also eine Fallunterscheidung für die Anzeige der Zusatzdaten (das betrifft vor allem die EIT-following und die EIT-schedule) auf dem HD-Kanal gemacht werden:

- Hat vorher ein Schaltvorgang vom SD- auf den HD-Kanal stattgefunden? Dann zeige die Zusatzinformationen des SD-Kanals bzw. Herkunftskanals. oder
- Wurde der HD-Kanal direkt angewählt? Dann zeige die Zusatzinformationen des eigenständigen HD-Kanals durchgängig.

Hat vorher ein Schaltvorgang stattgefunden, so ist dies dem Programm bekannt, denn dann wurden die Linkage Descriptoren entsprechend ausgewertet, das Programm befände sich quasi in Status 1 bzw. 2 (vgl. Abbildung 18). In diesem Falle könnte das Programm vor dem Schaltvorgang die Zusatzinformationen des SD-Kanals (Herkunftskanals) speichern und dann im HD-Kanal aus dem Speicher abrufen. Das Speichern ist notwendig, da ein Empfänger nicht gleichzeitig auf zwei Kanälen auf die EITs ‚horchen‘ kann, dafür bräuchte man ein zweites Empfängermodul, das für diesen Einsatz zu teuer sein dürfte. Selbst mit einem zweiten Empfängermodul könnte es je nach technischer Infrastruktur beim Empfänger zu Problemen kommen. Das Signal muss nämlich bei

Satellitenempfang für die beiden Empfänger in horizontale und vertikale Polarisation sowie Low- und High-Band unterscheidbar sein. Somit werden zwei Antennenanschlüsse benötigt.

Der Nachteil dieser Variante wäre, dass die Information für die Dauer des HD-Events statisch bleibt und nicht aktualisiert werden kann, da die Daten eben aus einem Speicher und nicht ‚live‘ abgerufen werden. Dass die Daten nicht aktualisiert werden können, ist nicht problematisch, da sie sich sowieso nur auf ein Event beziehen und danach ungültig werden. Hat entsprechend vorher kein Schaltvorgang stattgefunden, werden eben die Informationen für den HD-Kanal angezeigt.

Viel eleganter dagegen ist folgende Lösung (siehe Abbildung 19): Man richtet zwei HD-Kanäle (mit unterschiedlichen Service IDs) ein, die aber auf dieselben Elementarströme verweisen (*Shared PID*) und somit nur geringfügig mehr Datenrate benötigen, dafür aber beide Möglichkeiten zur Anzeige von Zusatzdaten ermöglichen.

Der eine der beiden Kanäle ist dann praktisch über die Linkage Descriptoren erreichbar und kann einen der hinteren Programmplätze in der Sendertabelle einnehmen. In diesem Kanal müssten jedem Event andere Zusatzinformationen (sowohl EIT-present, EIT-following als auch die vollständige EIT-schedule) mitgegeben werden und zwar in der Regel die des (SD-)Herkunftskanals.

Der andere Kanal ist eigenständig anwählbar, belegt möglicherweise auch einen der vorderen Plätze in der Sendertabelle und gibt dann auch die Zusatzinformationen so an, dass die Informationen der einzelnen HD-Events, so wie sie nacheinander im HD-Kanal ablaufen, angezeigt werden.

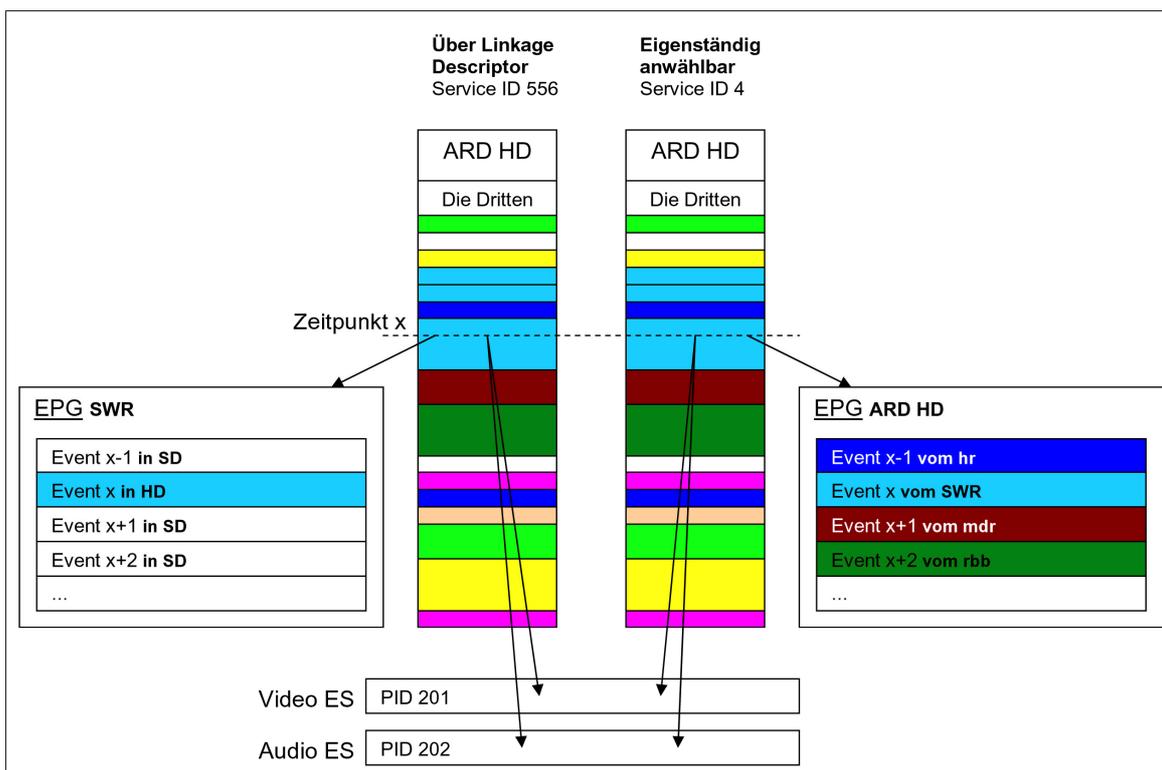


Abbildung 19: ARD-Gemeinschaftskanal mit zwei unterschiedlichen Zusatzinformationen

4.4 Verschiebungen und Störungen im Sendeablauf

Eine wichtige Frage im Zusammenhang mit dem HD-Gemeinschaftskanal und den jeweiligen SD-Landesrundfunkanstalten ist, wie im Falle einer Störung verfahren wird. Sowohl auf dem HD- als auch auf dem SD-Kanal können aus den verschiedensten Gründen Störungen auftreten. Im schlimmsten Fall kann es sogar zu einem zeitweisen Totalausfall von Bild, Ton und/oder Zusatzdaten kommen. Das Problem, das hier betrachtet werden soll, ist, wie sich die Störung auf den Schaltvorgang von SD nach HD und umgekehrt verhält. Ist der SD-Service fehlerhaft, kann möglicherweise, auf Grund des fehlenden Linkage Descriptors, keine Schaltung nach HD stattfinden, obwohl eventuell der HD-Service nicht betroffen ist. In diesem Fall müsste manuell eine Umschaltung auf den eigenständig anwählbaren HD-Service durch den Zuschauer erfolgen, da die automatische Signalisierung gestört ist.

Für den Fall, dass (der über den Linkage Descriptor erreichte) HD-Service gestört ist, gibt es einen ‚Fall Back‘ auf den SD-Service nach einer bestimmten Zeit, in der

kein entsprechender Linkage Descriptor in den Zusatzdaten gefunden wird. Dieses Vorgehen wird in Kapitel 5.3.1 näher erläutert.

Eine weitere mögliche „Störung“ kann eine Verschiebung im Sendeablauf sein, wenn sich beispielsweise, aus aktuellem Anlass, die Sendungen um eine bestimmte Zeit verschieben, eine Live-Sendung überzogen wird oder eine Programmänderung stattfindet. Unwahrscheinlich ist, dass dann das Programmschema im HD-Kanal so viel ‚Luft‘ hat, dass dort ebenfalls eine Verschiebung möglich ist. Wahrscheinlicher ist jedoch, dass eine Verschiebung im HD-Kanal nur schwer möglich ist. Zum einen durch die sowieso schwierige Programmplanung des HD-Services, zum anderen, weil möglicherweise ab einem bestimmten Zeitpunkt vor Sendestart kurzfristig keine Änderung mehr möglich ist. Das ganze Szenario der automatischen Umschaltung von SD auf HD macht natürlich nur Sinn, wenn die beiden in der Auflösung unterschiedlichen Events synchron zueinander laufen.

5 Empfänger-Demonstrator

Der Aufbau und die Funktionsweise des im Rahmen dieser Diplomarbeit entwickelten Empfänger-Demonstrators werden im nun folgenden Kapitel vorgestellt. Zunächst werden die verwendeten Geräte und Programme erklärt, anschließend wird im Detail auf das entwickelte Programm zur Analyse der EIT und zur Steuerung des Demonstrators eingegangen.

5.1 Versuchsaufbau und verwendete Geräte

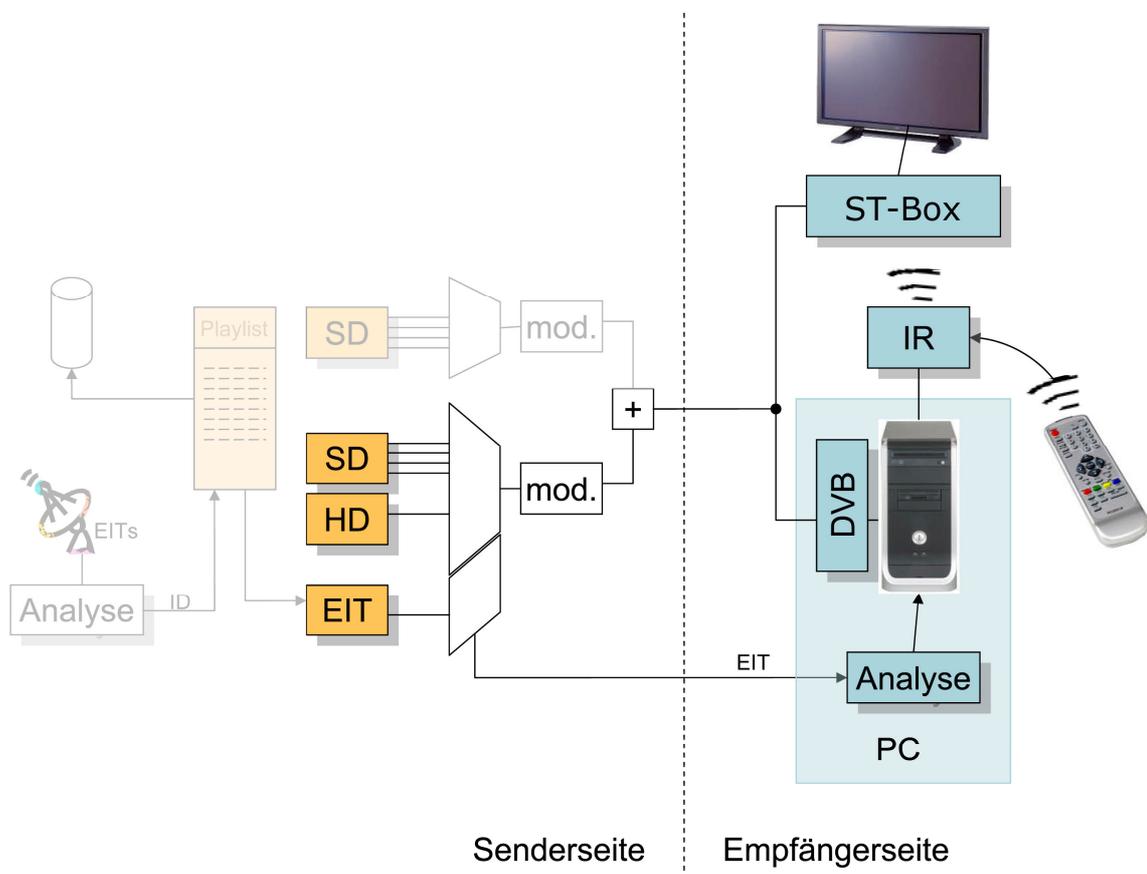


Abbildung 20: Versuchsaufbau nach Sender- und Empfängerseite getrennt

Die Anforderung an den Demonstrator war, dem Nutzer ein realistisches Gefühl bei der Bedienung zu vermitteln. Es sollte eine ‚state-of-the-art‘ Settop-Box verwendet werden, wobei die ‚fehlenden‘ Zusatzfunktionen in einem separaten PC implementiert werden sollten. Die Funktionsweise sollte einer direkt in den Receiver implementierten Funktion möglichst nahe kommen. Das bedeutet, der Demonstrator sollte aussehen wie eine Settop-Box und sich in der Bedienung

auch so anfühlen. ‚Look and feel‘ sollten dabei einem ‚echten‘ Produkt möglichst ähnlich sein, sprich die üblichen Interaktionen sollten über eine gewöhnliche Fernbedienung erfolgen können.

Die Alternative wäre gewesen, dieselben Funktionen ohne Settop-Box, aber stattdessen mit einer in den Rechner eingebauten Receiver-Karte zu realisieren. Es wäre jedoch eine ‚ominöse graue Kiste‘ gewesen und die unmittelbare Anwendbarkeit des entwickelten Konzepts auf ‚echte‘ Settop-Boxen wäre weniger augenfällig gewesen.

Doch wie steuert man einen Receiver, an dessen Funktionen man nicht so ohne weiteres Änderungen vornehmen kann? Die einzige Möglichkeit bestand darin, den Receiver über einen Infrarot-Sender zu steuern, dessen Verhalten wiederum durch ein Programm im Rechner beeinflusst wird. Wenn man jetzt den Rechner versteckt, wirkt die Bedienung des Receivers nicht wesentlich anders, als wenn die Funktion bereits in den Receiver integriert wäre.



Abbildung 21: Foto vom Versuchsaufbau

5.1.1 Blockschaltbild

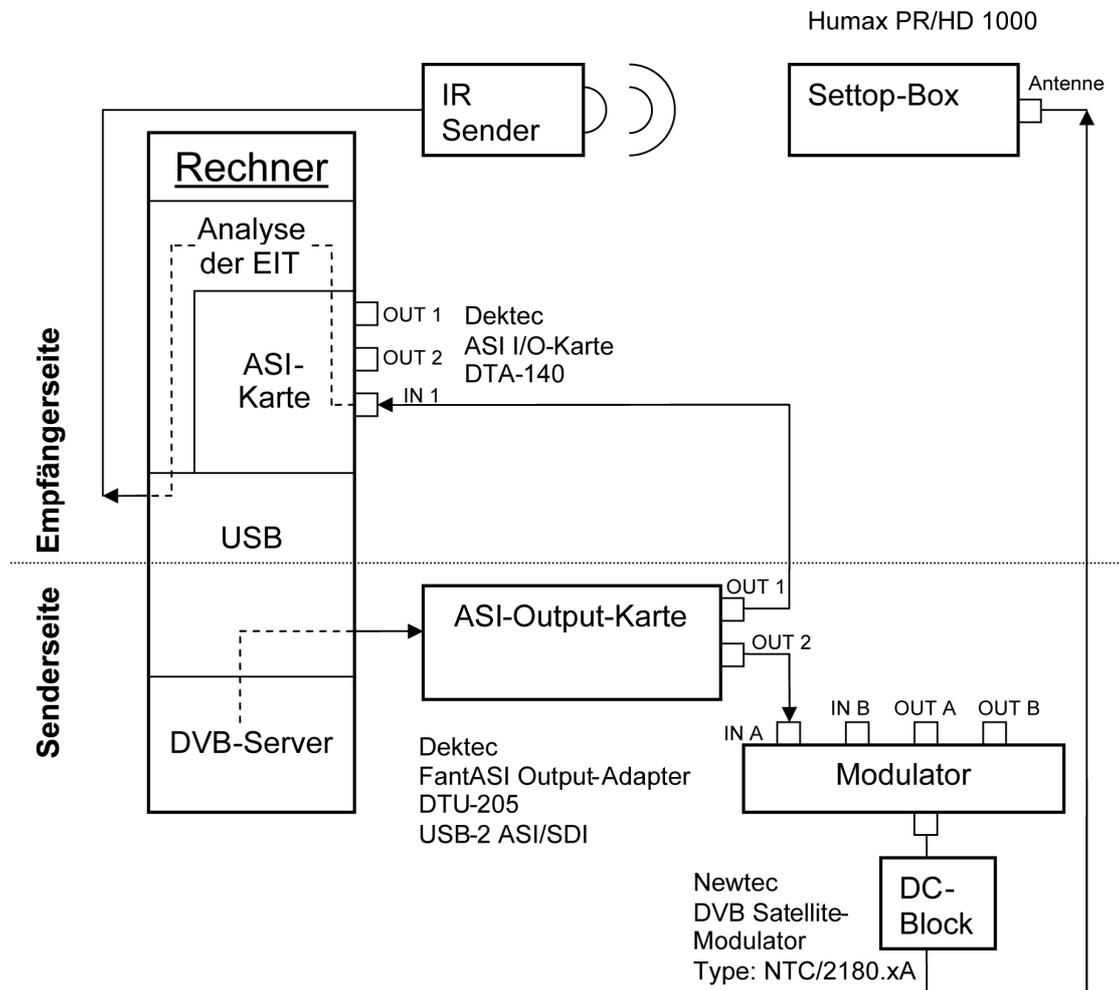


Abbildung 22: Blockschaltbild der im Demonstrator miteinander verbundenen Komponenten

Der realisierte Demonstrator besteht im Kern aus einem PC, der das Videosignal ausspielt, wieder einliest, analysiert und, je nach Analyse-Ergebnis, das angeschlossene Infrarot-Modul steuert. Daneben existiert ein paralleler Aufbau, bestehend aus einem Modulator, einer Settop-Box und einem daran angeschlossenen Display, der es ermöglicht, den analysierten Transportstrom vom Rechner über eine Output-Karte auf einem Display darzustellen.

Abbildung 22 stellt den Aufbau schematisch dar. Zunächst spielt der DVB-Playlist-Server, eine vom IRT entwickelte Software, die in diesem Kapitel noch näher beschrieben wird, einen Transportstrom mit mehreren Services über eine am USB-Anschluss angeschlossene ASI-Output-Karte aus. Diese teilt das Signal auf

zwei verschiedene Wege auf. Der erste Signalweg geht über einen Modulator mit DC-Block, der eine Übertragung von Gleichstrom-Spannungen auf nachfolgende Geräte verhindern soll [22], an die Settop-Box. An die Settop-Box kann dann ein HD-fähiges Display angeschlossen werden. Der zweite Signalweg geht quasi ‚als Parallelempfang‘ über eine ASI-Karte zurück in den Rechner. Dort wird das Signal dann von dem EIT-Analyse-Programm entsprechend ausgewertet und der Infrarot-Sender gesteuert.

5.1.2 Hardware

Die wesentlichen Hardware-Bausteine für den Empfänger-Demonstrator sind, neben dem PC, zwei ASI-Karten, ein HD-Satelliten-Receiver und ein Infrarot-Modul, das sowohl senden als auch Infrarot-Befehle empfangen kann.

ASI-Input-Karte

Über die im Rechner eingebaute DVB-ASI-Karte DTA-140 der Firma Dektec liest das EIT-Analyse-Programm die Daten ein.

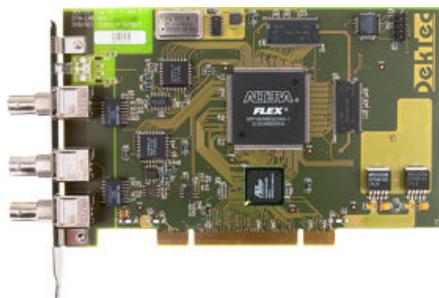


Abbildung 23: Dektec DTA-140, PCI-Bus-Karte für DVB-ASI Input + Output

ASI-Output-Karte

Über den via USB angeschlossenen ASI/SDI Output Adapter DTA-205 der Firma Dektec werden die Transportströme mit dem DVB-Playout-Server ausgespielt.



Abbildung 24: Dektec DTU-205, ASI/SDI Output Adapter

Modulator

Der Modulator bereitet das von der ASI-Output-Karte kommende Signal durch Modulation nach dem DVB-S-Standard und Mischung der Ausgangsfrequenz auf den Eingangsfrequenzbereich von Consumer-Settop-Boxen so auf, dass der Receiver es interpretieren kann. Er simuliert damit ein vom Satelliten kommendes Signal direkt auf der Frequenz, die auch von einem Universal-LNB einer Satellitenempfangsanlage ausgegeben würde.



Abbildung 25: NewTec NTC/2180, L-Band Satelliten Modulator

Receiver

Der Humax-Receiver PR-HD 1000 wandelt das vom Modulator kommende Signal in ein für den Fernseher empfangbares Signal um. Er demoduliert das Eingangssignal, erstellt einen Transportstrom und dekodiert diesen, um die eigentlichen Elementarströme zu erhalten.



Abbildung 26: Humax PR-HD 1000, Digitaler Satelliten-Receiver

Infrarot-Sender und –Empfänger

Das USB-Infrarot-Modul der Firma IRTrans dient als Infrarot-Sender, der dem Receiver Infrarot-Befehle schickt und somit die Settop-Box steuert. Gleichzeitig dient dieses Modul als Empfänger für die mit der Fernbedienung an die Settop-Box gesendeten Infrarot-Befehle. Infrarot-Befehle bestehen aus einer Folge von kurzen Lichtimpulsen, die auf eine Trägerfrequenz aufmoduliert sind. Die meisten Infrarot-Leuchtdioden arbeiten bei einer Wellenlänge von 950 nm. Bei der Infrarot-Technik muss eine Sichtverbindung zum Empfangsgerät bestehen, wobei helle Flächen, wie zum Beispiel die Zimmerdecke, das Licht reflektieren [23].

Bei diesem Gerät hat man die Möglichkeit, über eine mitgelieferte Software Infrarot-Befehle von bestehenden Fernbedienungen dem Infrarot-Modul zu lehren.

Die Software übernimmt dann die Codierung und Abspeicherung der Infrarot-Befehle in einer Datenbank. Weitere Details zur Steuerung des IRTrans folgen in Kapitel 5.1.3.



Abbildung 27: IRTrans USB Modul, Infrarot Sender und Empfänger

5.1.3 Software

Die zum Betrieb des Demonstrators erforderliche Software ist der DVB-Playout-Server, die Software zur Steuerung des IRTrans und das EIT-Analyse-Programm. Das EIT-Analyse-Programm wurde mit Delphi, einer auf Pascal basierenden, objektorientierten Programmiersprache, programmiert. Zum Erstellen der EITs wurde zusätzlich ein EIT-Editor verwendet. Auf die einzelnen Programme wird in den folgenden Abschnitten detaillierter eingegangen.

DVB-Playout-Server

Der DVB-Playout-Server, eine im IRT entwickelte Software, ist eine leistungsstarke Playout-Lösung zum Senden und Testen von SD- und HD-DVB Services. Diese Software kann als autonomes Playout-System für das Testen von MHP-Applikationen und Set-Top Boxen sowie zu Demozwecken und in Playout-Systemen verwendet werden. Das Programm kann außerdem verschiedene Transportstrom-Files für Remote-Playout bzw. Remote-Uplink erstellen und Transportströme re-multiplexen [24].

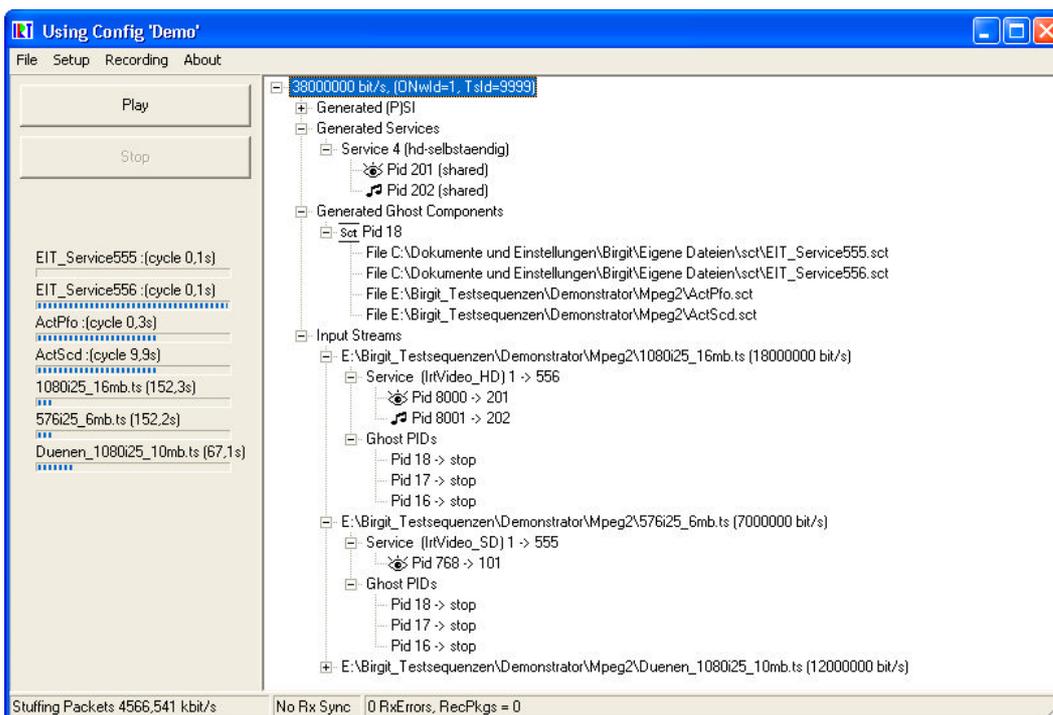


Abbildung 28: Screenshot der Oberfläche des DVB-Playout-Servers

Im Empfänger-Demonstrator multiplext der DVB-Server die von einer Datei ausgewählten Transportströme und Services mit den erstellten EITs zu einem neuen Transportstrom. Dabei erkennt der Server automatisch eine Änderung der EIT, wenn ‚auto-update‘ aktiviert ist. ‚Auto-update‘ ist eine Funktion des DVB-Servers, die diesen dazu bringt, die EIT erneut einzulesen, wenn der Server in dem Verzeichnis, in dem sich auch die EIT befindet, zusätzlich eine Datei mit dem Dateinamen der EIT und der Dateierweiterung ‚.update‘ findet. Damit kann ein Sendeablauf durch die Änderung der EIT simuliert werden. Dies wird dadurch erreicht, dass eine bestimmte EIT über den Aufruf einer entsprechenden Batch-Datei auf die ‚Master‘-EIT, auf die auch der DVB-Server zugreift, kopiert wird. Gleichzeitig erzeugt dieses Batch-File eine Datei mit der Dateierweiterung ‚.update‘, so dass der Server dazu veranlasst wird, die Daten der EIT erneut einzulesen.

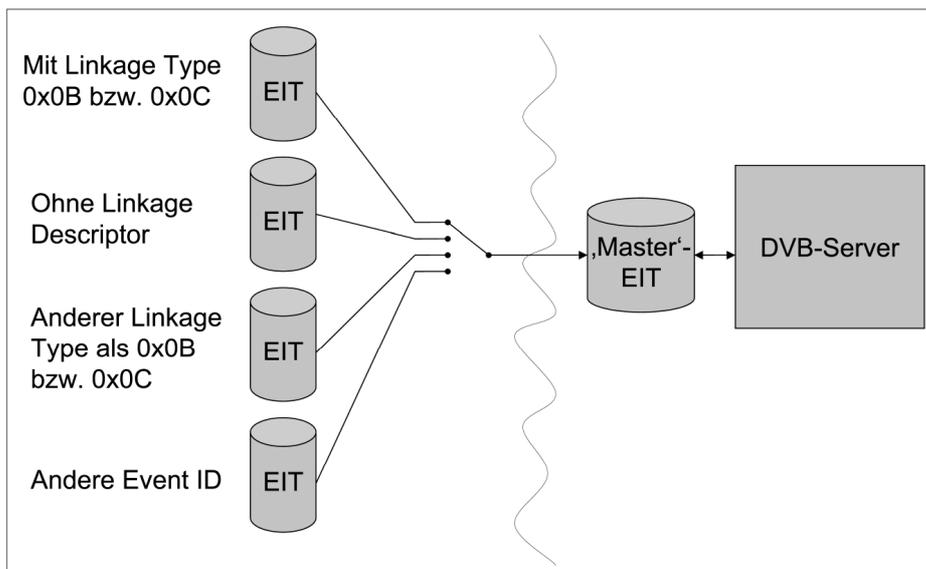


Abbildung 29: Prinzip mit dem die EITs über Batch-Dateien geändert werden

Für die unterschiedlichen Zustände im Empfänger-Demonstrator wurden verschiedene EITs angelegt:

Für den SD-Service mit der Service ID 555

- Mit Linkage Type 0x0B
- Vollständig ohne Linkage Descriptor
- Mit einem anderen Linkage Descriptor
- Mit einer anderen Event ID als die übrigen EITs

Für den HD-Service mit der Service ID 556

- Mit Linkage Type 0x0C
- Vollständig ohne Linkage Descriptor
- Mit einem anderen Linkage Descriptor

Diese Zustände (vgl. auch Abbildung 29) können dem DVB-Server über Batch-Dateien mitgeteilt werden. Man kann also bei der Vorführung des Demonstrators bewusst z.B. den Linkage Descriptor hinzufügen und wieder herausnehmen. Für den Demonstrator hat das den Vorteil, dass man das Verhalten schneller, individueller und in kürzeren Intervallen zeigen kann, als das beispielsweise bei einem automatisierten Ablauf der Fall wäre.

EIT-Editor

Mit dieser ebenfalls am IRT entwickelten und im Rahmen der Arbeit erweiterten Software ist es möglich, EITs zu erstellen. Man kann das DVB-Triple, also die

Service ID, die Transportstrom ID und die Original Network ID, angeben und wahlweise einen Linkage Descriptor mit der entsprechenden Information, wohin verlinkt werden soll, eintragen. Außerdem lässt sich der EIT noch eine Event ID zuordnen. Mit diesem Programm wurden alle EITs erstellt, die über die Batch-Dateien in den DVB-Server ‚eingehängt‘ werden können.

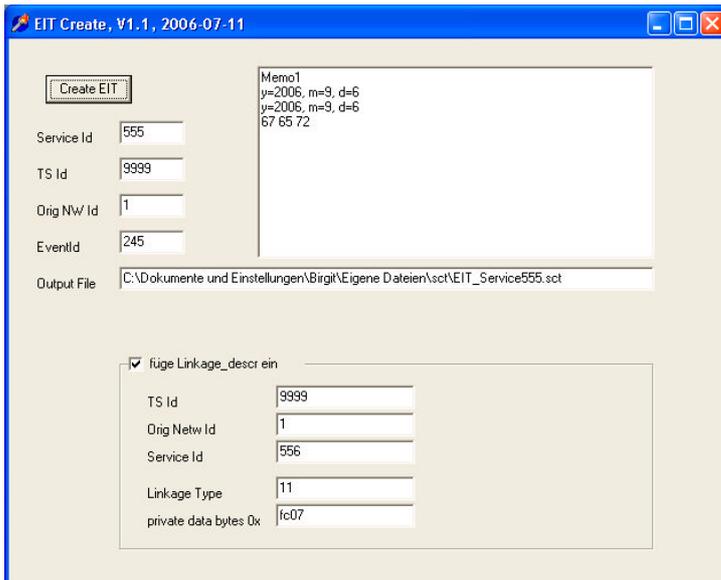


Abbildung 30: Screenshot der Oberfläche des EIT-Editors

IRTrans-Steuersoftware

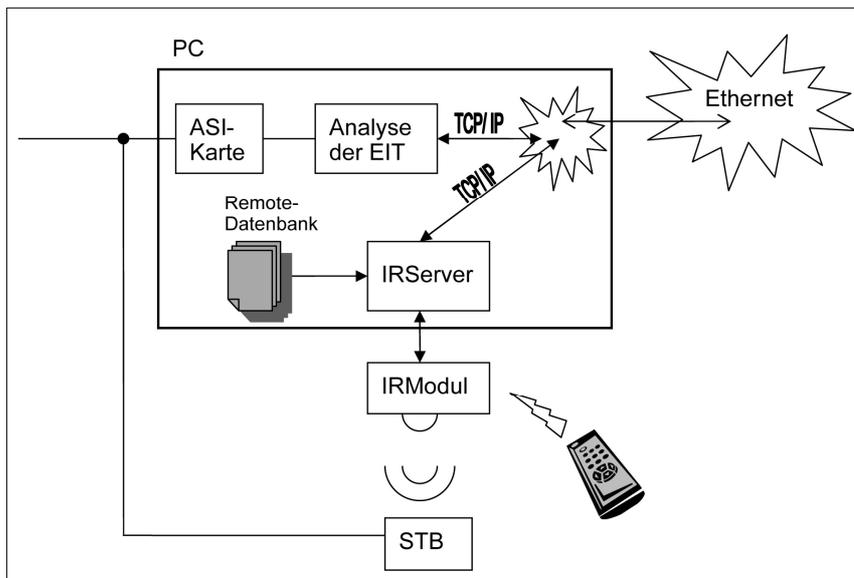


Abbildung 31: Kommunikationsprinzip des Infrarotmoduls (IrTrans)

Die gesamte Software, die das IRTrans-Infrarotmodul steuert, besitzt eine Client / Server Struktur, so dass jeder Zugriff auf das Sende- und Empfangsmodul (*IRModul*) über den Server (*IRServer*) laufen muss. Der *IRServer* muss also ständig im Hintergrund laufen, um das *IRModul* nutzen zu können. Der *IRServer* benutzt dabei das TCP/IP Interface zur Kommunikation mit anderen Clients [25].

In einer Infrarotdatenbank (Remote-Datenbank) werden die vorher gelernten Remote-Infrarotbefehle in Form von ASCII Dateien (siehe Abbildung 32) abgespeichert und durch den *IRServer* verwaltet. Aus dieser Datenbank können die Befehle dann abgerufen werden. Ein Infrarot-Befehl kann mit der unter ‚Remote‘ angegebenen Bezeichnung sowie mit dem unter ‚Commands‘ angegebenen Befehl angesprochen und gesendet werden. Auch die Identifizierung eines empfangenen Signals läuft über diese Datenbank.

```

humax.rem
[REMOTE]
[NAME]humax
[TIMING]

[COMMANDS]
[prog1][RAW]71[FREQ]38[D]0 32 792 0 16 416 536 592 520 592 536 592 512 592 544
584 544 592 536 592 520 584 520 1744 512 592 544 584 520 1720 544 584 520 616
512 1720 544 584 496 1744 512 1728 536 592 544 584 520 592 536 568 536 616 520
584 544 592 512 616 512 1720 544 1696 544 1720 512 1720 544 1696 536 1720 520
[prog2][RAW]71[FREQ]38[D]0 32 768 0 16 432 520 592 536 592 520 608 496 616 512
616 536 592 520 592 536 592 536 1704 536 592 536 592 520 1712 544 592 512 592
544 1720 512 592 544 584 544 592 512 1720 544 592 488 616 536 592 544 568 536
592 536 1720 520 1720 544 584 520 1720 512 1720 544 1720 512 1720 520 1720 544
[prog3][RAW]71[FREQ]38[D]0 32 784 0 16 416 520 608 520 616 512 592 544 584 520
616 488 616 536 592 544 592 512 1720 512 616 520 592 536 1720 496 616 536 592
512 1728 536 592 512 1720 520 608 544 1696 544 584 544 592 512 592 520 608 544
592 512 592 536 1728 512 592 536 1720 496 1744 536 1704 536 1720 520 1720 536

```

Abbildung 32: Ausschnitt aus einer ASCII-Remote-Datei

Die im Demonstrator verwendeten Infrarot-Codes mussten im so genannten RAW-Format angelegt werden, da der verwendete Humax-Receiver mit keinem der im Standardformat kodierten Befehle anzusprechen war. Das RAW-Format wird verwendet um Befehle zu lernen, die anders nicht darstellbar sind.

5.2 Verwendete Module

Bei der Entwicklung des Empfänger-Demonstrators konnten bereits vorhandene Komponenten genutzt werden. Auch wurden, wie in der Programmierung häufig üblich, Schnittstellen definiert. Das Programm wird hierbei in Bereiche geteilt, für

die möglicherweise jeweils ein anderer Programmierer zuständig ist. Dabei müssen die Schnittstellen so klar formuliert werden, dass eindeutig ist, welche Parameter und Funktionen für den eigenen Bereich benötigt und welche an einen anderen Programmierer übergeben werden müssen. Der Vorteil dieser Schnittstellendefinition ist, dass man sich auf einen bestimmten Teil des Programms konzentrieren kann, ohne sich um die Details des restlichen Programms kümmern zu müssen. Das bedeutet, dass man bestimmte Werte bzw. Daten zur Verfügung hat, deren genaues Zustandekommen aber unerheblich ist.

Zu Beginn der Programmierarbeit konnte auf ein Programm, das die SDT, PAT und PMT auswertet, zurückgegriffen werden. In diesem Programm war bereits die Möglichkeit, Daten von der ASI-Karte auszulesen, enthalten. Diese Funktion konnte somit für den Demonstrator genutzt werden.

Bei der Programmierung der Kommunikation mit dem IR-Modul konnte außerdem ein Sourcecode in C++ als Beispiel verwendet werden, der bei der Installations-CD für den IRTrans enthalten war. Die Übersetzung nach Delphi und Integration in das EIT-Analyse-Programm wurde mit der Unterstützung eines Programmierers durchgeführt.

Das Gesamtprogramm läuft in drei parallelen Threads. Ein Thread ist in der Informatik ein bestimmter Ausführungsstrang. So läuft im EIT-Analyse-Programm neben dem Haupt-Thread ein Thread (DektecThread), der sich ständig, päckchenweise Daten von der ASI-Karte holt und ein Thread (IrReceiveThread), der ständig überprüft, ob über das IR-Modul ein Infrarot-Befehl empfangen wurde.

Detektiert der IrReceiveThread ein Kommando und soll dieser Befehl übersetzt und wieder gesendet werden, dann führt das Programm den jeweiligen Befehl aus und wartet dann weiter auf den nächsten empfangenen Infrarot-Code.

Der DektecThread ruft immer dann, wenn er eine Section mit der PID 18 (0x0012) findet die Prozedur zur Analyse der EIT (HandleEitData) auf. Danach ruft der DektecThread weiter Daten von der ASI-Karte ab, bis er wieder eine entsprechende Section findet.

Im Haupt-Thread werden u.a. die Form (grafische Bedienoberfläche) mit ihren Buttons, Statusanzeigen und Eingabefeldern definiert und erzeugt. Daneben

laufen eine Reihe von Prozeduren, wie z.B. die Prozedur zum Einlesen des Config-Files (siehe 5.3.7) und - teilweise - die Prozedur zum Senden eines Infrarot-Kommandos (siehe 5.3.5) im Haupt-Thread.

Der Schwerpunkt bei der Programmierung dieses Demonstrators war im Wesentlichen die Analyse der Event Information Table (EIT). Hierfür wurden zahlreiche Prozeduren und Funktionen programmiert. Diese werden im Folgenden erläutert.

5.3 Funktionen des EIT-Analyse-Programms

Wie in Kapitel 4.1 beschrieben, baut der Empfänger-Demonstrator auf einem Vorschlag zur Erweiterung von EN 300 468 und TR 101 211 auf. Bei der Implementierung des Vorschlags sind einige Schwierigkeiten aufgetreten, die weitere Ergänzungen des Vorschlags notwendig gemacht haben. Im folgenden Abschnitt wird das State Diagramm (vgl. Kapitel 3.7) vorgestellt, mit dessen Funktionsablauf das EIT-Analyse-Programm programmiert wurde. Außerdem wird auf einige wesentlichen Stellen des Quelltextes des EIT-Analyse-Programms eingegangen.

5.3.1 State diagram

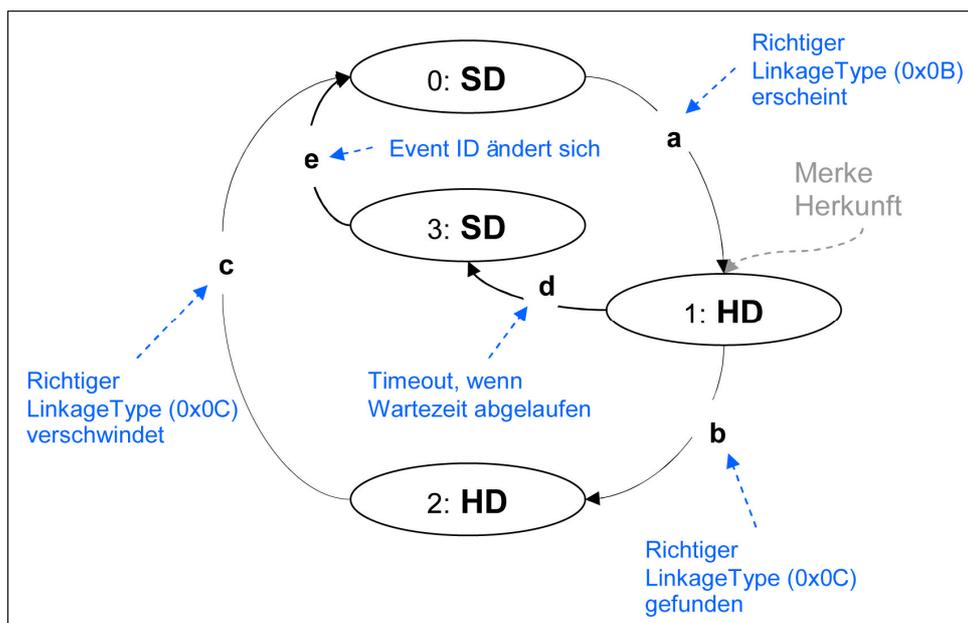


Abbildung 33: State Diagramm des Empfänger-Demonstrators

Bedingung a	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	LinkageType muss vorhanden sein und er muss 0x0B sein
Bedingung b	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	LinkageType 0x0C muss <i>mit einem LinkageDvbTriple auf den Herkunftskanal</i> vorhanden sein
Bedingung c	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	Der LinkageType 0x0C muss verschwinden <i>oder</i> LinkageDvbTriple auf den Herkunftskanal muss verschwinden
Bedingung d	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	Die Wartezeit für das Auftauchen des Linkage Types 0x0C ist abgelaufen
Bedingung e	EIT muss von aktueller Service ID sein
	EIT muss ‚present‘ sein
	Wenn die Event ID sich gegenüber der Herkunftsevent ID ändert, d.h. eine neue Sendung startet

Tabelle 6: Liste der Bedingungen a bis e für den Empfänger-Demonstrator

Im Unterschied zu den vorherigen State Diagrammen (siehe Kapitel 4.1.2 und Kapitel 4.2) wurde hier ein Timeout hinzugefügt (Bedingung d), dessen Schaltzeit ca. 6 Sekunden beträgt. Diese Zeit setzt sich aus einer Schätzung der Zeit zum sicheren Auffinden einer EIT (die EIT wird zwar alle 2 Sekunden wiederholt, aber es ist möglich, dass der Receiver die erste Sendung der EIT nur unvollständig mitbekommt) und der Schaltzeit der verwendeten Settop-Box zusammen. Ist im SD-Kanal (korrekt oder irrtümlich) der Linkage Descriptor mit dem Linkage Type 0x0B vorhanden (d.h. das Programm befindet sich im Status 1), aber im HD-Kanal wird kein (oder der falsche) Linkage Descriptor gefunden, dann würde nun das EIT-Analyse-Programm auf den Herkunft-SD-Kanal (Status 3) zurückschalten, wenn es 6 Sekunden lang keinen Linkage Type 0x0C bzw. ein Linkage Type 0x0C, der nicht auf den Herkunftskanal verweist, findet. Der Status 3 wurde als Zwischenstation eingefügt, damit der Receiver nicht ständig zwischen SD und HD hin- und herschaltet. Da der Linkage Type 0x0B im SD-Kanal (Status 0) wahrscheinlich noch vorliegt, würde der Receiver sonst sofort wieder auf den HD-

Kanal schalten (Status 1), dort aber möglicherweise wieder keinen Linkage Type 0x0C finden und wieder auf den SD-Kanal zurückschalten, usw. Das bedeutet, das Programm würde aus dieser Schleife nicht wieder heraus kommen, der Receiver würde ständig zwischen HD und SD hin- und herschalten. Deshalb schaltet das Programm erst wieder auf den Status 0, wenn eine neue Sendung beginnt, d.h. sich die Event ID ändert (Bedingung e). Erst dann ist, bei Auftauchen des Linkage Types 0x0B, wieder ein Schalten auf den HD-Kanal (Status 1) möglich. Möchte man jedoch bewusst einen erneuten Versuch unternehmen, auf den HD-Kanal zu kommen, so muss der Zuschauer einmal einen anderen Kanal mit der Fernbedienung anwählen und anschließend wieder auf den eigentlichen Kanal zurückschalten. Bei einem Kanalwechsel oder beim Aufrufen des Analyse-Programms startet das Programm immer im Status 0. Auf diese Weise wird sichergestellt, dass auch im Falle einer fehlerhaften oder unvollständigen Signalisierung maximal 1 Schaltvorgang für den Zuschauer sichtbar wird.

5.3.2 Auffinden der benötigten Daten in der Prozedur HandleEitData

„HandleEitData“ ist eine Prozedur aus dem EIT-Analyse-Programm, in der die benötigten Daten aus dem Transportstrom ausgelesen und abgespeichert, die Stati 0 bis 3 definiert, die Bedingungen a bis e definiert und abgeprüft werden sowie der Sendebefehl an das Infrarot-Modul ausgelöst wird.

Um die beiden ineinander verschachtelten for-Schleifen der EIT abzubilden (vgl. Abbildung 11), musste entsprechend eine ähnliche Struktur für die Prozedur, die die EIT auswertet, gewählt werden. Abbildung 34 zeigt eine vereinfachte Darstellung der Struktur von HandleEitData. Im oberen, weißen Bereich werden Table ID, Section Length, Service ID, Section Number, Transportstrom ID und Original Network ID aus der EIT herausgesucht und auf entsprechende Variablen gespeichert.

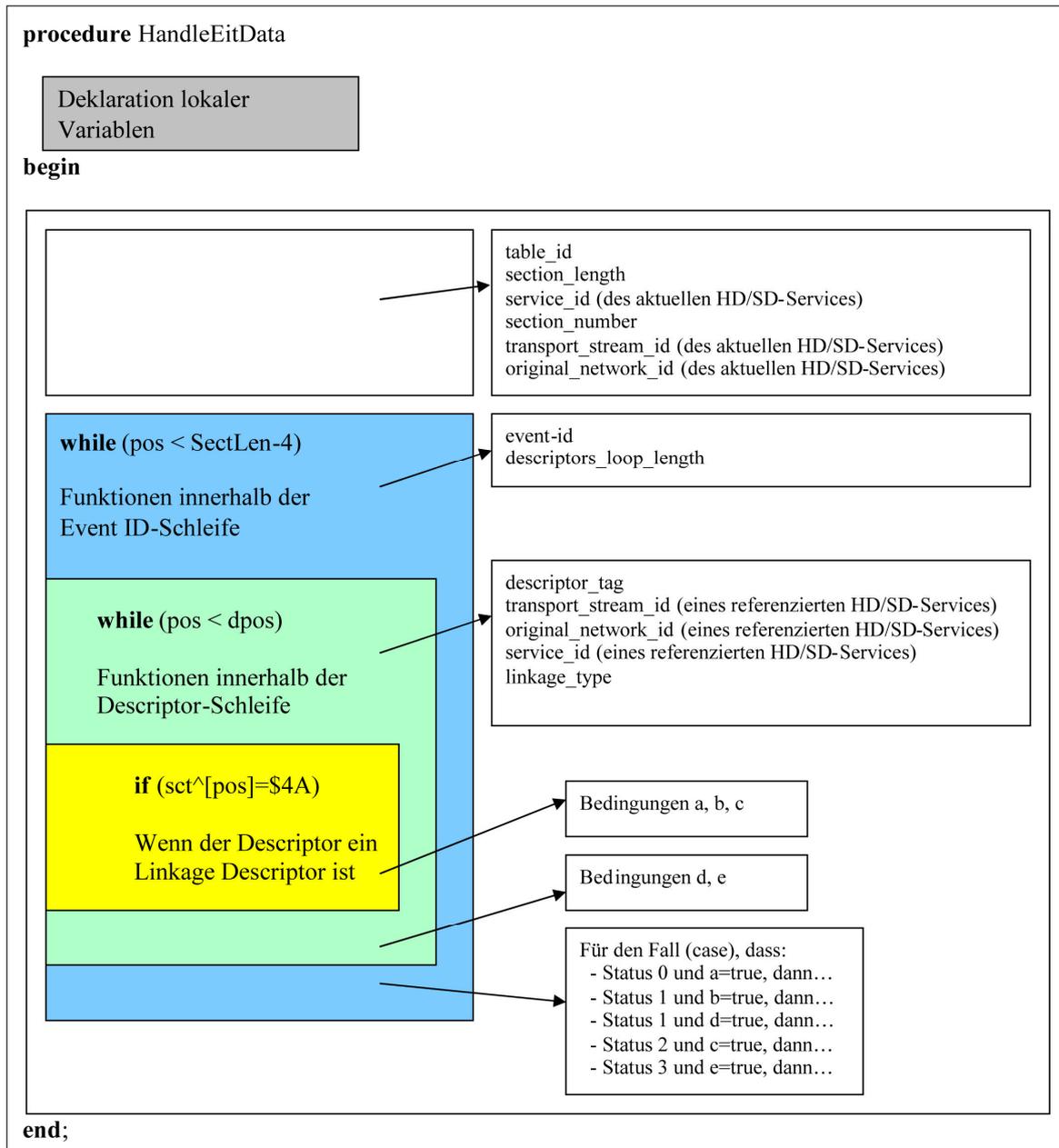


Abbildung 34: Vereinfachte Struktur der Prozedur HandleEitData

Außerhalb der beiden for-Schleifen ist das Auffinden der Werte noch recht einfach, da man sie byteweise abfragen kann. In Abbildung 35 ist beispielhaft die Belegung des DVB-Triples, bestehend aus der Service ID (hier: SvId), der Original Network ID (hier: NwId) und der Transportstrom ID (hier: TsId) dargestellt. Die Service ID ist laut der Norm 2 Byte oder 16 Bit lang (vgl. Abbildung 11 und Abbildung 36) und steht an Position 3 und 4 der Tabelle (Zählung beginnend von Null). Um die beiden Bytes addieren zu können, muss das höherwertigere Byte vor der Addition

um zwei Stellen nach links verschoben werden (siehe Abbildung 35). Dabei entspricht der Verschiebung nach links eine Multiplikation mit 100 im Hexadezimal-System bzw. mit der Zahl 256 im Dezimalsystem.

```
SvId := 256*sct^[3] + sct^[4]; {service_id aus dem Header der EIT}
NwId := 256*sct^[10] + sct^[11]; {original network id aus dem Header der EIT }
TsId := 256*sct^[8] + sct^[9]; {transport stream id aus dem Header der EIT}
```

Abbildung 35: Belegung der Variablen für das DVB-Triple in der Prozedur HandleEitData

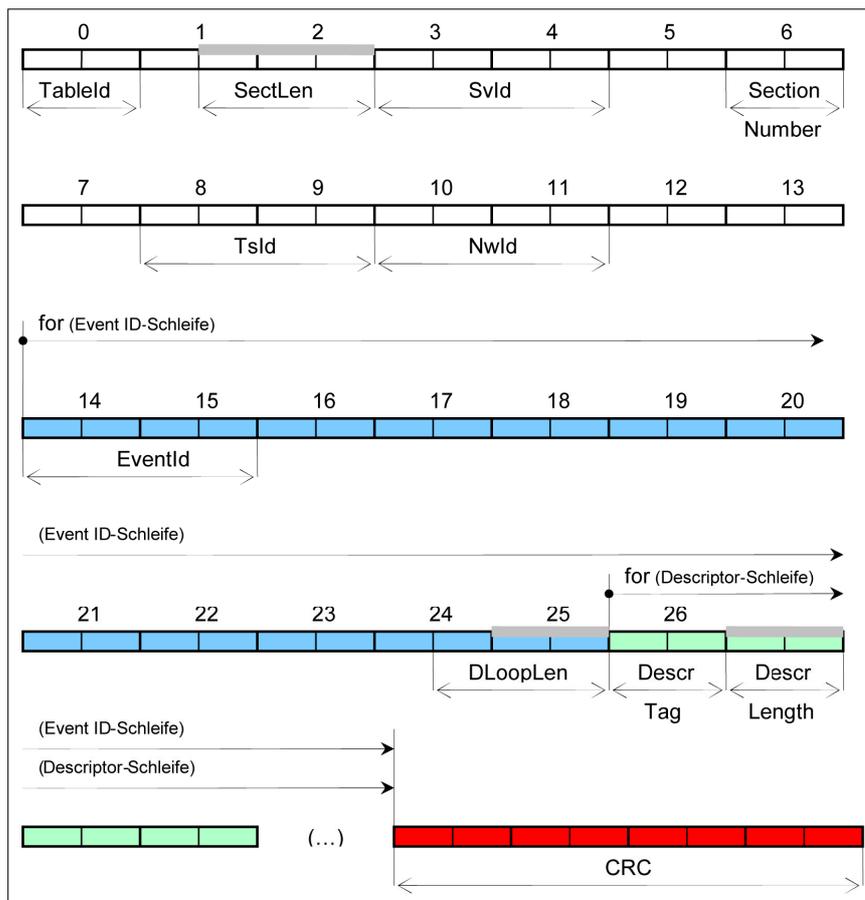


Abbildung 36: Event Information Table (EIT) aufgeteilt nach Bytes

Das Auffinden von Informationen innerhalb der Event-ID- und der Descriptor-Schleife ist etwas schwieriger, da deren Länge sich erst durch Angaben innerhalb der EIT ergeben. Man muss sich also von einer Schleife zur nächsten ‚durchhangeln‘, die Informationen lesen und entsprechend verwerten oder verwerfen (siehe Abbildung 37).

Am Anfang jeder Event-ID-Schleife muss die Information der ‚Descriptors_loop_length‘, die Länge der Descriptor-Schleife, ermittelt werden. Je

nach Längenangabe können dann in der Descriptor-Schleife unterschiedlich viele Descriptoren vorkommen. Am Anfang jedes Descriptors wiederum steht der Descriptor Tag, der angibt, um welche Art von Descriptor es sich handelt (z.B. Linkage Descriptor), sowie die Länge des jeweiligen Descriptors (*descriptor_length*) (vgl. Kapitel 3.6).

Mit diesen Informationen kann man sich nun schrittweise die Informationen herausuchen. Die Variable ‚*pos*‘ im Quellcode gibt dabei die jeweils aktuelle Position an (schwarze senkrechte Pfeile in Abbildung 37). Eine weitere Variable ‚*dpos*‘ gibt jeweils den Anfang der nächsten bzw. das Ende der vorangegangenen Event-ID-Schleife an.

Die Bedingung für die Event-ID-Schleife (blau markiert in Abbildung 34, Abbildung 36 und Abbildung 37) ist *while (pos < SectLen-4)*. Also solange die aktuelle Position ‚*pos*‘ kleiner ist als die gesamte Section-Länge (*section_length*) abzüglich der 4 Byte für den CRC (rot markiert), sollen die Befehle im Sourcecode ausgeführt werden, d.h. dass u.a. die Event ID und die Descriptor-Loop Länge herausgesucht werden. Die Descriptor-Schleife (grüne Markierung), die mehrere verschiedene Descriptoren enthält, hat im Programm die Bedingung *pos < dpos* und wird entsprechend so lange ausgeführt wie die aktuelle Position kleiner ist als das Ende der Descriptor-Schleife. Dabei wird für jeden Descriptor überprüft, ob es sich um einen Linkage Descriptor handelt. Falls nein, wird lediglich die Länge des Descriptors ermittelt, um mit dieser Information zum nächsten Descriptor springen zu können. Wird dagegen ein Linkage Descriptor gefunden, d.h. ist der Descriptor Tag gleich 0x4A (*if sct^[pos] = \$04A*), dann wird u.a. die Information, welcher HD- oder SD-Service referenziert ist (Linkage DVB-Triple), herausgesucht sowie die Umschalt-Bedingungen a bis c überprüft und ggf. gesetzt.

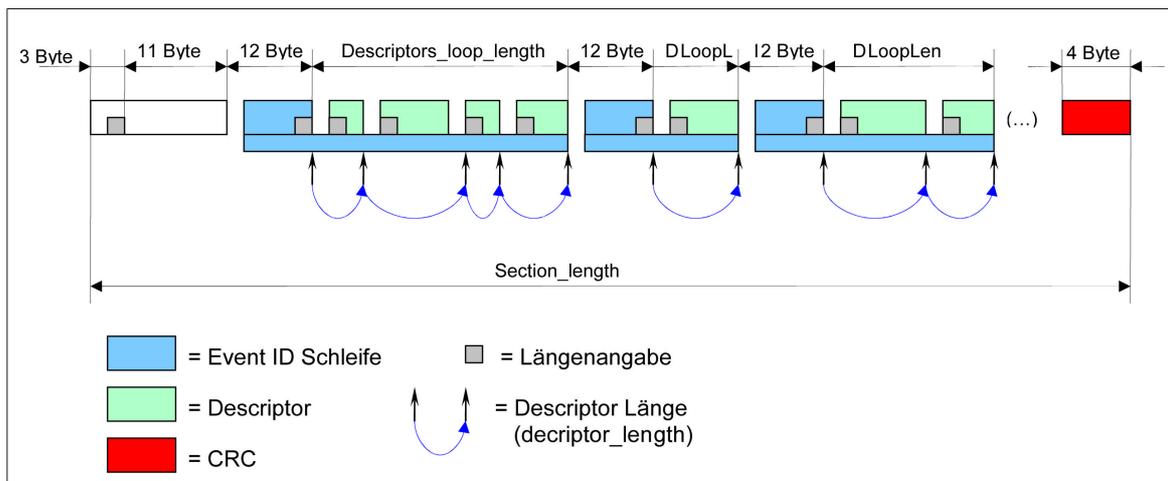


Abbildung 37: Prinzip der schrittweisen Informationssuche in der Prozedur *HandleEitData*.

5.3.3 Bedingungen a bis e

Die Bedingungen zum Umschalten auf einen anderen Zustand werden in diesem Abschnitt näher erläutert (vgl. auch Tabelle 6). Die Bedingungen a bis c können dabei nur dann *true* (bzw. *false* bei Bedingung c) werden, wenn ein Linkage Descriptor gefunden wurde, das Programm sich also in der in Abbildung 34 gelb markierten if-Schleife innerhalb der Prozedur *HandleEitData* befindet. Die Bedingungen d und e dagegen müssen auch dann auf *true* gesetzt werden können, wenn kein Linkage Descriptor gefunden wurde. Deshalb werden sie in der Descriptor-Schleife erst nach der if-Bedingung gesetzt.

Bei allen Bedingungen wird überprüft, ob es sich bei dem gerade gefundenen DVB-Triple um das des Kanals handelt, der momentan eingeschaltet ist (z.B. blauer Rahmen in Abbildung 38) sowie ob es sich um eine EIT-present (z.B. roter Rahmen in Abbildung 38) handelt.

Wenn außerdem der Linkage Type noch vom SD- auf den HD-Kanal referenziert (Linkage Type 0x0B; grüner Rahmen Abbildung 38), dann wird die boolesche Variable a auf *true* (wahr) gesetzt (gelber Rahmen).

```

if (NwId = CurrentChannelNwId) and
    (TsId = CurrentChannelTsId) and
    (SvId = CurrentChannelSvId) and
    (SectionNumber=$0) and
    (LinkageType=$B) then begin
    a:= TRUE;
    Form1.Info (TimeToStr(now) + ': a:= ' + BoolToStr(a));
    Form1.Info(' '); {Leerzeile}
end;

```

Abbildung 38: Bedingung für a=true im Sourcecode

Bei Bedingung b wird ebenfalls das DVB-Triple, sowie die EIT-present überprüft. Wenn der Linkage Type vom HD- auf den SD-Service referenziert (Linkage Type 0x0C; grüner Rahmen in Abbildung 39) und auch noch das in dem Linkage Descriptor angegebene DVB-Triple auf den SD-Kanal verweist, von dem gerade hergeschaltet wurde (magentafarbener Rahmen), dann wird b gleich *true* gesetzt (gelber Rahmen).

```

if (NwId = CurrentChannelNwId) and
    (TsId = CurrentChannelTsId) and
    (SvId = CurrentChannelSvId) and
    (SectionNumber=$0) and
    (LinkageType=$C) and
    (InLinkageReferencedNwId = Form1.HerkunftsNwId) and
    (InLinkageReferencedTsId = Form1.HerkunftsTsId) and
    (InLinkageReferencedSvId = Form1.HerkunftsSvId) then begin
    b:= TRUE;
    Form1.Info (TimeToStr(now) + ': b:= ' + BoolToStr(b));
    Form1.Info(' '); {Leerzeile}
end;

```

Abbildung 39: Bedingung für b=true im Sourcecode

Um das Verschwinden des Linkage Types 0x0C erfassen zu können, wurde bei Bedingung c eine inverse Logik gewählt. Dabei ist c immer true, es sei denn die Bedingungen aus Abbildung 40 treffen zu, dann wird die Bedingung c auf *false* gesetzt. So kann man sagen, dass ein Wechsel von einem Status in einen anderen immer dann stattfinden kann, wenn die entsprechende Bedingung *true* ist. Das macht das Programm etwas übersichtlicher und leichter verständlich.

Die Bedingungen für die Variablen b und c sind hierbei im Programmcode identisch. Trotzdem wurde entschieden, zwei verschiedene if-Bedingungen zu schreiben. Falls einmal etwas an dem Programm geändert werden sollte, z.B.

zusätzliche States oder Übergangsbedingungen definiert und deshalb auch existierende geändert werden, könnte es sein, dass man versehentlich einen Zusammenhang übersieht. Wenn man dann an einer gemeinsamen Bedingung etwas ändert, dann könnte es sein, dass das Programm nicht mehr wie erwartet funktioniert. Aus diesem Grund ist es besser, von vornherein zwei eigenständige if-Bedingungen zu formulieren.

```

if (NwId = CurrentChannelNwId) and
    (TsId = CurrentChannelTsId) and
    (SvId = CurrentChannelSvId) and
    (SectionNumber=$0) and
    (LinkageType=$C) and
    (InLinkageReferencedNwId = Form1.HerkunftsNwId) and
    (InLinkageReferencedTsId = Form1.HerkunftsTsId) and
    (InLinkageReferencedSvId = Form1.HerkunftsSvId) then begin
    c:= FALSE;
    Form1.Info (TimeToStr(now) + ': c:= ' + BoolToStr(c));
    Form1.Info(' '); {Leerzeile}
end;

```

Abbildung 40: Bedingung für $c=false$ im Sourcecode

Beim Übergang von Status 0 auf Status 1, also beim Schalten vom SD- auf den HD-Kanal, wird die aktuelle Systemzeit auf die globale Variable *TimerStart* geschrieben. Ist im weiteren Verlauf des Programms die Differenz zwischen aktueller Systemzeit und der auf der Variablen *TimerStart* gespeicherten Zeit größer als 6 Sekunden, wird die Variable *d* auf *true* gesetzt (siehe Abbildung 41). Vorausgesetzt das DVB-Triple und die EIT-present stimmen.

```

if ((now-Form1.TimerStart) > 6/(24*3600)) and {6s}
    (NwId = CurrentChannelNwId) and
    (TsId = CurrentChannelTsId) and
    (SvId = CurrentChannelSvId) and
    (SectionNumber=$0) then begin
    d:= TRUE;
    //Form1.Info ('TimerStart: ' + DateTimeToStr(Form1.TimerStart));
    Form1.Info (TimeToStr(now) + ': d:= ' + BoolToStr(d));
    Form1.Info(' '); {Leerzeile}
end;

```

Abbildung 41: Bedingung für $d=true$ im Sourcecode

Bedingung *e* (siehe Abbildung 42) wird dann *true*, wenn sich die Event ID gegenüber der Event ID des Kanals, von dem hergeschaltet wurde, ändert. Auch hier müssen wieder das DVB-Triple und die EIT-present überprüft werden.

```

if (EventId <> Form1.HerkunftsEventId) and
  (NwId = CurrentChannelNwId) and
  (TsId = CurrentChannelTsId) and
  (SvId = CurrentChannelSvId) and
  (SectionNumber=$0) then begin
  e:= TRUE;
  Form1.Info (TimeToStr(now) + ': e:= ' + BoolToStr(e));
  Form1.Info(' '); {Leerzeile}
end;

```

Abbildung 42: Bedingung für e=true im Sourcecode

5.3.4 States 0 bis 3

Die einzelnen States aus dem State-Diagramm (Abbildung 33) wurden im Programmcode innerhalb der Prozedur *HandleEitData* mit einer so genannten case-Anweisung realisiert. Die case-Anweisung wird verwendet, um eine Verzweigung zu den entsprechenden Quelltext-Zeilen vorzunehmen. Dabei ist die case-Anweisung einer Folge von if-Anweisungen oftmals vorzuziehen, da die Logik einer case-Anweisung normalerweise leichter zu verstehen ist als die von komplex geschachtelten if-Anweisungen. Darüber hinaus wird der Quelltext in case-Anweisungen schneller ausgeführt [26].

Die case-Anweisung fragt im Empfänger-Demonstrator den Wert der Variable für den aktuellen Status (*ChannelStatus*) ab. Diese Variable kann, analog zu den verschiedenen States, den Wert 0, 1, 2 oder 3 annehmen. Abhängig von diesem Wert führt das Programm dann die unter der entsprechenden Ziffer angegebenen Quelltext-Zeilen aus (siehe Abbildung 43). So wird zum Beispiel, wenn der Status gerade Null ist, überprüft, ob die Bedingung a gleich wahr ist (*if (a=TRUE) then begin*). Ist dies der Fall, werden verschiedene Aktionen ausgeführt, u.a. wird ein Infrarot-Befehl über den IRTrans an die Settop-Box geschickt, um auf den HD-Kanal zu schalten.

Ist der Wert für *ChannelStatus* gleich eins, wird in diesem Fall eine Unterscheidung vorgenommen, ob die Bedingung b oder die Bedingung d wahr ist. In Abhängigkeit davon werden unterschiedliche Aktionen ausgeführt. So wird z.B. der Status auf 2 gesetzt, wenn die Bedingung b wahr ist oder z.B. der Status auf 3 gesetzt (sowie wieder auf den SD-Kanal geschaltet), wenn die Bedingung d wahr ist.

Eine Übersicht über die Aktionen, die in dem jeweiligen Status bzw. case erfolgen, ist in Abbildung 43 dargestellt. Sie gibt das, was im Quellcode programmiert ist, in ‚normalen‘ Worten wieder, also nicht in einer Programmiersprache, wohl aber in der in der für case-Anweisungen üblichen Struktur.

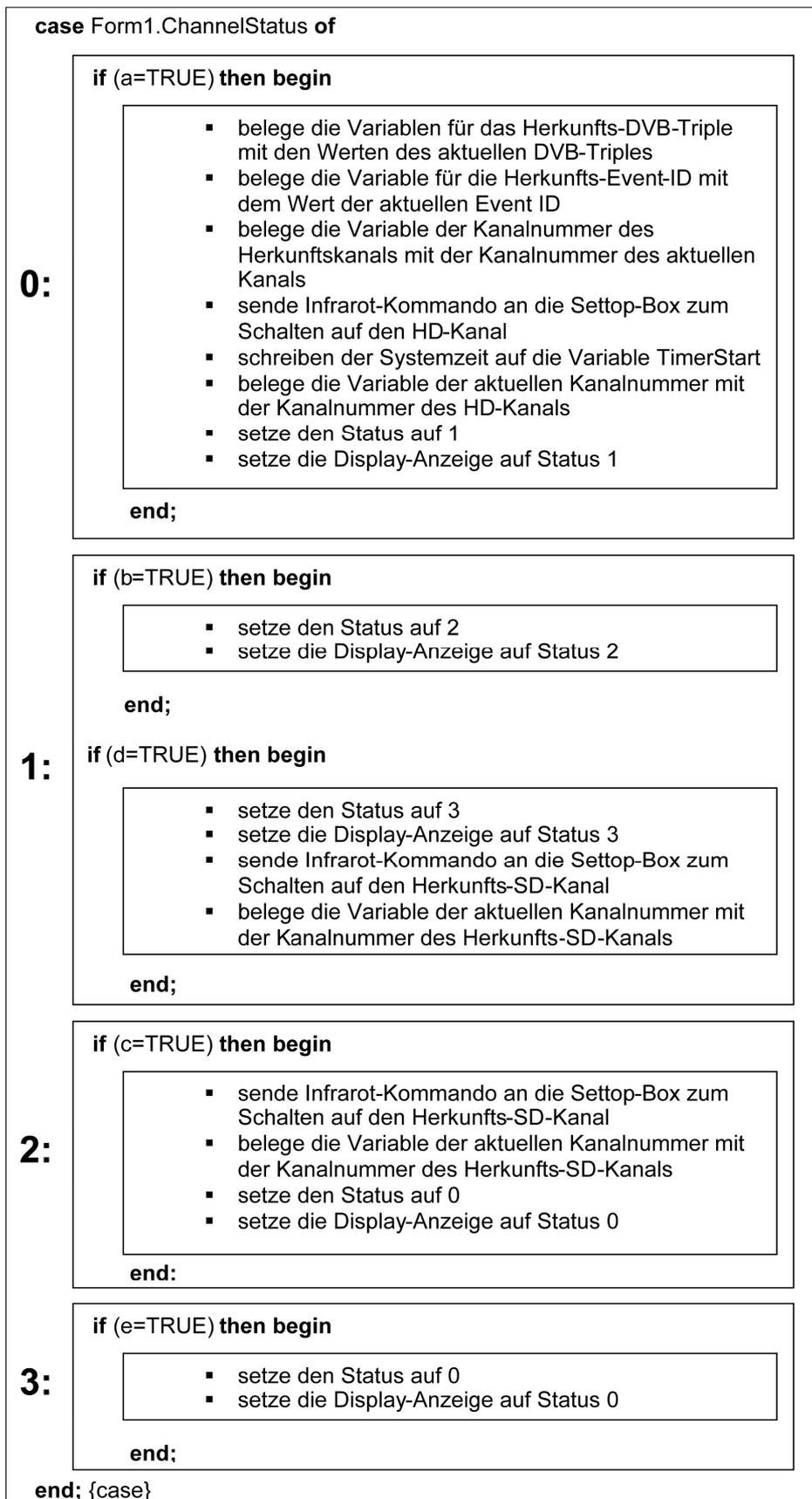


Abbildung 43: Vereinfachte Darstellung der case-Anweisung in der Prozedur HandleEitData

5.3.5 Senden von Befehlen über das Infrarot-Modul

Der Sendebefehl zum Umschalten auf den HD-Kanal bzw. SD-Kanal wird in den entsprechenden case-Anweisungen ausgelöst. In der case-Anweisung selbst werden dabei lediglich zwei weitere Prozeduren aufgerufen, denen die zum Ausführen benötigten Werte ‚mitgegeben‘ werden. Die aufgerufenen Prozeduren wiederum setzen dann das Infrarot-Kommando entsprechend dieser übergebenen Werte ab.

Um Befehle mit mehreren Ziffern (beispielsweise die Ziffernfolge 918, die hier der Programmplatz für den HD-Kanal ist) senden zu können, muss die mehrstellige Zahl zunächst in ihre einzelnen Ziffern zerlegt werden. Diese Ziffern können dann nacheinander gesendet werden. Die ist erforderlich, weil in der Remote-Datenbank nur die einzelnen Befehle, also diejenigen, die einer Taste auf der Fernbedienung entsprechen, abgespeichert sind. Eine Befehlsfolge erfordert deshalb auch eine Abfolge einzelner Befehle, die dem IR-Modul zuvor gelehrt und in der Datenbank abgespeichert wurden.

Die Prozedur *SendPaceChannel* zerlegt die ihr übergebene Ziffernfolge (*PaceProgNo*) in ihre einzelnen Ziffern. Die Zahl wird in einen Textstring umgewandelt (*IntToStr*) und dann, angefangen mit dem ersten Zeichen, zeichenweise an die zweite Prozedur *SendIrCommand* übergeben. Nach dem Senden des Befehls für das erste Zeichen wird eine bestimmte Zeit gewartet (*WaitMs*), damit der gerade abgesetzte Befehl vor dem Senden des nächsten Befehls sicher ausgeführt werden kann. Diese Wartezeit ist variabel und kann der Prozedur *SendPaceChannel* in Abhängigkeit der Erfordernisse mit übergeben werden (vgl. Abbildung 44).

```

procedure SendPaceChannel (PaceProgNo, WaitMs: integer); {Programmnummer aus mehreren
  Ziffern, Verzögerung zwischen den Einzelbefehlen}
var ChannelString: string;
    i: integer;
begin
  ChannelString:= IntToStr (PaceProgNo);
  for i:=1 to length(ChannelString) do begin
    if (i>1) then sleep(WaitMs);
    Form1.SendIrCommand (CRemote, 'prog' + ChannelString[i]);
    end;
  end; {SendPaceChannel}

```

Abbildung 44: Sourcecode der Prozedur *SendPaceChannel*

Die Prozedur *SendIrCommand* stellt die Verbindung mit dem Infrarot-Modul her und greift auf die Remote-Datenbank zu, sucht dort das benötigte Kommando heraus und sendet diese via Infrarot-Modul an die Settop-Box (vgl. Abbildung 31).

5.3.6 Verarbeitung von empfangenen Infrarot-Befehlen

Das Programm ist auch in der Lage, zur Laufzeit Befehle einer Fernbedienung zu empfangen und zu verarbeiten, da der Empfang in einem eigenen Thread läuft. Um die an die Settop-Box gesendeten Befehle filtern zu können, wurde eine andere Fernbedienung als die eigentlich zur Settop-Box gehörende Fernbedienung verwendet. Eine Verwendung der Originalfernbedienung war deshalb nicht möglich, weil der Receiver direkt auf die gesendeten Befehle reagiert hätte, was er aber in bestimmten Situationen nicht soll. Der Befehl der ‚fremden‘ Fernbedienung wird nun vom Infrarot-Modul empfangen, vom Rechner analysiert und ggf. in einen ‚Original-Befehl‘ der Settop-Box übersetzt sowie über das Infrarot-Modul an die Settop-Box gesendet. Die durch die Rechenzeit bedingte Zeitverzögerung ist dabei kaum wahrnehmbar. Der Vorteil der Verwendung einer anderen Fernbedienung besteht darin, dass man noch mehr machen kann als eine bloße Übersetzung des Befehls. Man kann die Befehle filtern und das Verhalten abhängig machen von der Art des empfangenen Befehls oder dem gerade aktuellen Status des Programms. Außerdem besteht die Möglichkeit, bestimmte Tasten inaktiv zu schalten. Der Befehl wird dann zwar korrekt empfangen und verarbeitet, aber nicht an die Box weitergeleitet. Im Statusfenster des Empfänger-Demonstrators erscheint dann eine Textzeile ‚Dies ist kein gültiges Kommando‘. Das ist z.B. der Fall bei der Taste *EPG*. Da die Information des EPG in diesem Empfänger-Demonstrator nicht mitgesendet bzw. absichtlich deaktiviert wird, kann ein Druck dieser Taste auch keine sinnvolle Anzeige verursachen. Aus diesem Grund wurde die Funktion inaktiv geschaltet.

Ein weiteres Beispiel stellt die Verarbeitung der ‚Zap‘-Befehle *Programm+* und *Programm-* (vgl. Abbildung 45) dar. In diesen Fällen muss nämlich eine Unterscheidung im PC vorgenommen werden, auf welchem Kanal sich die Settop-Box zurzeit befindet (blauer bzw. roter Kasten in Abbildung 45). Auf den SD-Kanälen kann der Befehl wie gewohnt übersetzt werden. Auf dem HD-Kanal dagegen, der in diesem Empfänger-Demonstrator auf einem der hinteren

Programmplätze (hier Programmplatz-Nummer 918) liegt, soll das Programm beim Empfang des Befehls *Programm+* den Befehl ‚Herkunfts kanal + 1‘ schicken. Wird also auf dem HD-Kanal beispielsweise gerade eine Sendung des NDR übertragen und befindet sich der (SD-)NDR auf dem Programmplatz mit der Nummer 8, so soll an den Receiver beim Drücken der Taste *Programm+* der Befehl ‚Programm 9‘ und nicht ‚Programm 919‘ gesendet werden.

In Abbildung 45 ist die Realisierung des Befehls *Programm+* dargestellt.

```

if (Kommando='prog+') then begin
  if (Form1.CurrentPaceChannel = 9) then begin
    Form1.Info('Es sind nur einstellige Ziffern belegt');
    end {Form1.CurrentPaceChannel = 9}
  else
    if (Form1.CurrentPaceChannel = Form1.ChannelList[0].PaceChannel) then begin
      if (Form1.HerkunftsChannelSd = 9) then begin
        Form1.Info('Es sind nur einstellige Ziffern belegt');
        end {Form1.HerkunftsChannelSd = 9}
      else if (Form1.HerkunftsChannelSd <> 9) then begin
        Form1.SendIrCommand (CRemote, 'prog' +
          IntToStr((Form1.HerkunftsChannelSd)+1));
        Form1.SendIrCommand(CRemote, 'ok');
        Form1.LastRemoteKlick:= now;
        sleep (200);
        Form1.Info('Befehl prog' + IntToStr(Form1.HerkunftsChannelSd +1) + ' wurde
          übersetzt und gesendet');
        Form1.CurrentPaceChannel:= ((Form1.HerkunftsChannelSd)+1);
        Form1.ChannelStatus:= 0;
        DisplayStatus (0);
        end {Form1.HerkunftsChannelSd <> 9}
      end {if Form1.CurrentPaceChannel = Form1.ChannelList[0].PaceChannel}
    else
      if (Form1.CurrentPaceChannel <> Form1.ChannelList[0].PaceChannel) then begin
        Form1.SendIrCommand(CRemote, 'prog+');
        Form1.Info('Befehl prog+ wurde übersetzt und gesendet');
        Form1.CurrentPaceChannel:= ((Form1.CurrentPaceChannel)+1) ;
        Form1.ChannelStatus:= 0;
        DisplayStatus (0);
        end {Form1.CurrentPaceChannel <> Form1.ChannelList[0].PaceChannel}
    end {if Kommando prog+}

```

Abbildung 45: Sourcecode der Verarbeitung des Infrarot-Befehls „Programm+“

Für den Demonstrator war es ausreichend, wenn lediglich die Programmplätze 1 bis 9 anwählbar sind. Demnach müssen aber andere Ziffernkombinationen entsprechend gefiltert und blockiert werden. Dies geschieht einmal durch einen Timeout, der Befehle abweist, die vor Ablauf dieser Wartezeit eintreffen. Zum anderen wird nach der Übersetzung der jeweiligen Ziffer der Befehl ‚OK‘ hinterhergeschickt, um den Umschaltvorgang zu beschleunigen und die Eingabe von mehrstelligen Ziffern zu verhindern.

Des Weiteren wird durch eine Abfrage verhindert, dass man durch Drücken der Taste *Programm+* und *Programm-* nicht über 9 (if/else if - Unterscheidung im roten Kasten, Abbildung 45) bzw. unter 1 gelangt.

5.3.7 Eingabe von Basisdaten über ein Config-File

Auch das EIT-Analyse-Programm benötigt etwas Vergleichbares zu der Sendertabelle in der Settop-Box. Das Analyseprogramm muss ebenfalls wissen, welcher Service sich auf welchem Senderplatz befindet. Das kann je nach verwendeten Services unterschiedlich sein oder sich auch mal ändern. Um diese veränderlichen Daten nicht mühsam im Sourcecode suchen und an vielen Stellen ändern zu müssen, wurde eine Config-Datei erstellt, in der die entsprechenden Daten eingetragen wurden und dort auch bei Bedarf geändert werden können. Beim Starten des EIT-Analyse-Programms werden diese Daten ausgelesen (Prozedur *ReadConfigFile*) und auf Variablen, mit denen dann weitergerechnet wird, abgespeichert. Es müssen also nur einmal (bzw. nur wenn sich etwas ändert) und auch nur an einer Stelle (nämlich in der Config-Datei) die Daten eingetragen werden.

Tabelle 7 zeigt eine Auflistung der ‚Sendertabelle‘, die aus der Config-Datei ausgelesen wird (das Original ist im Anhang zu finden). Da für die Demonstration ein HD- und ein SD-Kanal ausreichen, werden momentan nur die Kanäle mit der Service ID (*ChannelSvld*) 555 für SD und 556 für HD im Programm verwendet. Kommen senderseitig weitere SD-Kanäle hinzu, können diese durch ein Update der Config-Datei problemlos in das Programm integriert werden. Auch eine Angabe/Abfrage der Transponderfrequenz ist möglich, ist aber für den Betrieb des Demonstrators mit ASI-Schnittstellen nicht erforderlich.

Record No.		ChannelNwld	ChannelTsld	ChannelSvld	PaceChannel	Frequenz
0	HD	1	9999	556	918	Xy
1	SD	1	9999	555	1	Xy
2	SD	1	9999	22	2	Xy
3	SD	1	9999	23	3	Xy
4	SD	1	9999	24	4	Xy
5	SD	1	9999	25	5	Xy
6	SD	1	9999	26	6	Xy
7	SD	1	9999	27	7	Xy
8	SD	1	9999	28	8	Xy
9	SD	1	9999	29	9	Xy

Tabelle 7: Tabellarische Darstellung der Werte der Config-Datei

Daneben muss in die Config-Datei noch die Anzahl der Sender (*ElementCount*) und der Startkanal des Receivers (*PaceStartChannel*), das ist der Kanal in dem die Box beim Einschalten startet und sich meist über das Menü der Settop-Box einstellen lässt, eingetragen werden. Ansonsten lässt sich das Verhalten der Settop-Box und des Analyse-Programms nicht synchronisieren.

Gespeichert werden die Daten in einem so genannten ‚Array of records‘.

Ein Record ist dabei ein Verbund aus Komponenten unterschiedlichen Typs. Nach Tabelle 7 wäre ein Record eine Zeile in dieser Tabelle. Der Record mit der Nummer 0 hätte die Komponenten Network ID (*ChannelNwld*), Transportstrom ID (*ChannelTsld*), Service ID (*ChannelSvld*), Programmplatznummer (*PaceChannel*) und Frequenz.

Ein Array (Feld) ist dabei eine Anordnung oder Reihe von gleichen Elementen in einer festgelegten Art und Weise. In diesem Beispiel wäre das Array die gesamte Tabelle mit den immer in gleicher Weise aufgebauten Records (hier Record 0 bis 9).

Abfragen lässt sich die Programmplatznummer des HD-Kanals im Quellcode beispielsweise mit folgender Zeile:

```
HdPaceChannel:= ConfigFile.ReadInteger('0','PaceChannel',0);
```

Dabei ist '0' die Record-Nummer, 'PaceChannel' die gesuchte Komponente und die 0 der Default-Wert, wenn in der Config-Datei nichts Passendes steht. Die anderen Komponenten in den verschiedenen Records lassen sich analog dazu ansprechen.

5.3.8 Paralleler Schaltvorgang von Settop-Box und Analyse-Programm

Taucht ein Linkage Descriptor im Transportstrom auf, dann steht im Linkage Descriptor auch in Form des DVB-Triples die Information wohin verlinkt werden soll. Mit dieser Information soll das Programm die dazugehörige Programmplatznummer heraussuchen (dies geschieht in der Funktion *GetChannelListPositionFromDvbTriple*) und dann die Box sowie sich selbst auf diese Nummer umschalten. Das Prinzip des Auslesevorgangs und das Finden des richtigen Kanals ist in Abbildung 46 dargestellt: Mit den Informationen des DVB-Triples durchsucht das Programm die Config-Datei nach den drei Werten des DVB-Triples und gibt die dazugehörige Programmplatznummer zurück.

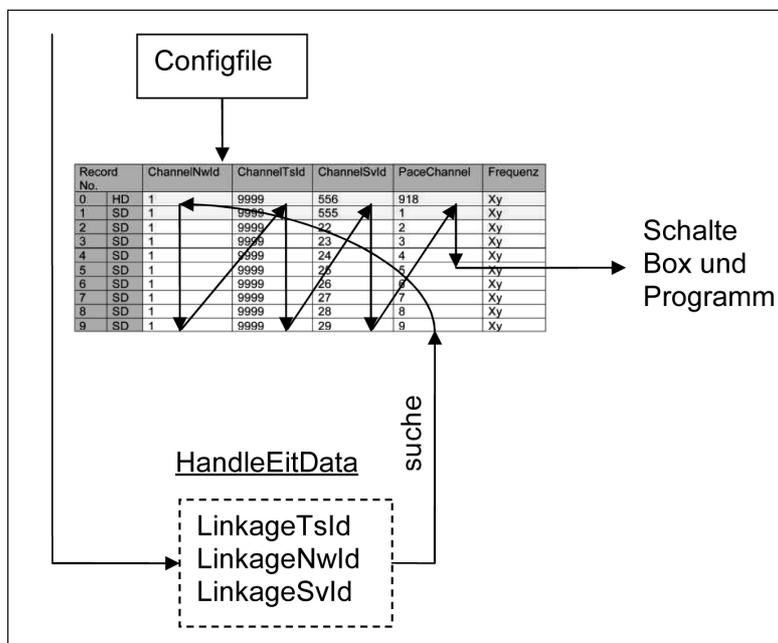


Abbildung 46: Prinzip des Auslesevorgangs und Findens des richtigen Kanals aus der Config-Datei

Wie erwähnt, muss zu jeder Zeit sichergestellt sein, dass Settop-Box und Analyse-Programm sich immer auf demselben Kanal befinden. Die Settop-Box muss sich dabei vor dem Start des Programms auf dem in der Config-Datei eingetragenen Startkanal (*PaceStartChannel*) befinden, um eine gemeinsame Startbedingung zu schaffen. Also muss zusätzlich zum Umschalten der Box immer auch das Analyse-Programm selbst auf den neuen Kanal umschalten und die EIT-Suche ab diesem Zeitpunkt auf dem neuen Kanal fortsetzen. Nach dem Einschalten der Settop-Box und dem Start des Analyseprogramms gibt es dann jedoch keine

weitere Kontrollmöglichkeit mehr, ob beide noch synchron laufen, da der Status der Settop-Box dem Programm nicht bekannt ist, denn dafür gibt es keine Verbindung. Dieses mögliche Problem wird aber in dem Augenblick, in dem die Funktion in die Settop-Box integriert ist, gelöst sein. Denn dann gibt es nicht mehr zwei Sendertabellen, die synchronisiert werden müssen, sondern nur noch eine.

Falls das Programm und die Box doch unerwarteterweise auseinander laufen, weil beispielsweise der Infrarot-Befehl von der Box nicht richtig erkannt wurde, gibt es auf der Bedienoberfläche einen Reset-Button, der die Box einmal aus- und wieder einschaltet, die Config-Datei erneut einliest und das Programm wieder in den Status 0 versetzt. Seit der Verwendung der Humax-Box wurde diese Funktion jedoch nur noch selten gebraucht, da die Infrarot-Befehle sehr zuverlässig empfangen werden. Bei der vorher verwendeten Pace-Box war das anders. Eine Recherche in diversen Internet-Foren hatte ergeben, dass auch andere User Probleme mit der Pace-Fernbedienung haben. Aus diesem Grunde und nach Rückschluss mit der Herstellerfirma des IRTrans, wurde die Pace-Box gegen eine Humax-Box ausgetauscht (siehe Kapitel 5.4.2).

5.3.9 Grafische Darstellung des Programmablaufs

Um den Status und den Ablauf des EIT-Analyse-Programms nicht nur in einer Textzeile im Statusfenster auszugeben, wurde das State-Diagramm in die Bedienoberfläche übernommen. Dabei zeigt die wechselnde gelbe Markierung der States an, in welchem Status sich das Programm gerade befindet. Die Bedingungen a bis e sind farblich so markiert, dass der Zustand erkennbar ist. Dabei steht rot für *false* (unwahr) und grün für *true* (wahr).

In den folgenden vier Abbildungen sind die vier States sowie die Zustände der Bedingungen dargestellt. Jeweils rechts neben dem State-Diagramm ist ein Ausschnitt aus dem Video, je nach Status in SD oder in HD, zu sehen. Darunter ist die Information, die auf dem Display der Settop-Box angezeigt wird, abgebildet. Dabei handelt es sich bei den beiden Abbildungen auf der rechten Seite jeweils um das, was der Zuschauer sieht, bei dem auf der linken Seite um das, was intern im Programm passiert. Der Zuschauer kennt somit nur zwei verschiedene Zustände (SD oder HD), das Programm dagegen die bereits erläuterten vier Stati.

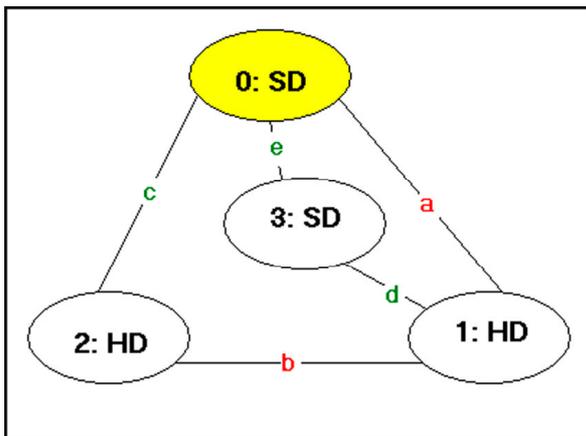


Abbildung 47: Anzeige von Status 0, SD-Video bild und SD-Anzeige auf der Settop-Box

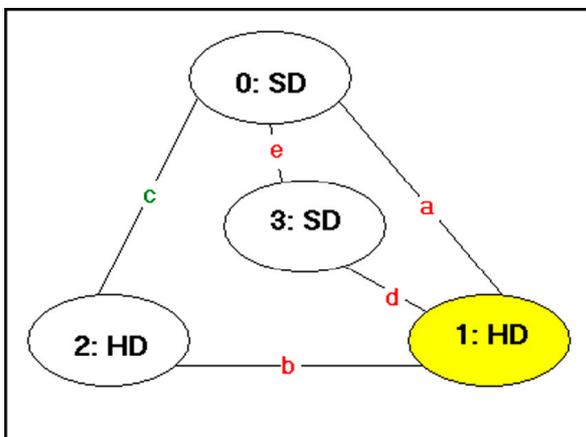


Abbildung 48: Anzeige von Status 1, HD-Video bild und HD-Anzeige auf der Settop-Box

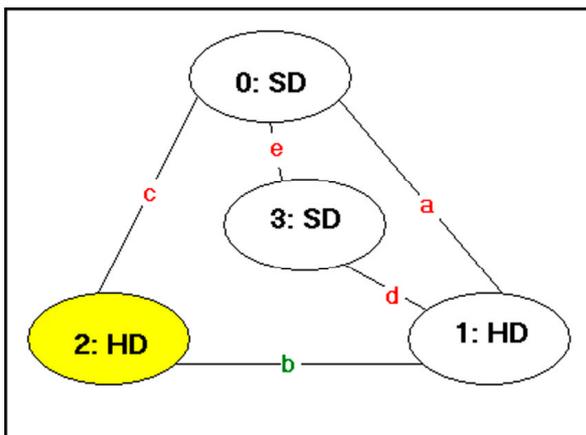


Abbildung 49: Anzeige von Status 2, HD-Video bild und HD-Anzeige auf der Settop-Box

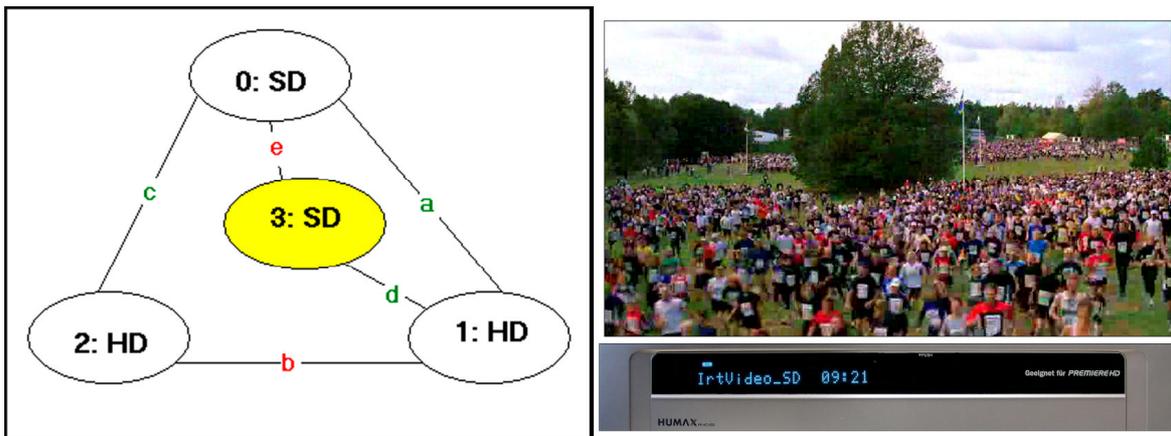


Abbildung 50: Anzeige von Status 3, SD-Videobild und SD-Anzeige auf der Settop-Box

Abbildung 51 zeigt die Bedienoberfläche des EIT-Analyse-Programms. Auf der linken Seite werden verschiedene Zustände angezeigt. Ganz oben befindet sich die Anzeige, ob der Status gerade SD oder HD ist, darunter der genaue Status (*ChannelStatus*) als Zahl. Danach folgen die Anzeigen für den aktuellen Kanal (*CurrentPaceChannel*), die Service ID im Linkage Descriptor (*LinkageSvcId*), die Kanalnummer des SD-Herkunftskanals (*HerkunftsChannelSd*) sowie die Zustände der Bedingungen a bis e.

Rechts neben der gerade erläuterten grafischen Darstellung des State-Diagramms befindet sich ein Statusfenster für diverse Textmeldungen, an dem sich gut das detailliertere Verhalten des Programms ablesen lässt.

Im unteren Teil befinden sich eine Reihe von Buttons zur Modifikation des Linkage Types. Diese Buttons rufen dabei lediglich die in Kapitel 5.1.3 beschriebenen Batch-Dateien auf. Sie dienen der Vereinfachung der Programmtests und spielen im praktischen Betrieb des Demonstrators keine Rolle.

Außerdem existieren, neben dem bereits erwähnten Reset-Button, noch ein Button zum manuellen Setzen des Status über ein Eingabefeld (*SetChannelStatus*), ein Button um die Box an- bzw. auszuschalten sowie ein Button zum Testen der Infrarotverbindung und Senden des Infrarot-Kommandos *Programm1*.

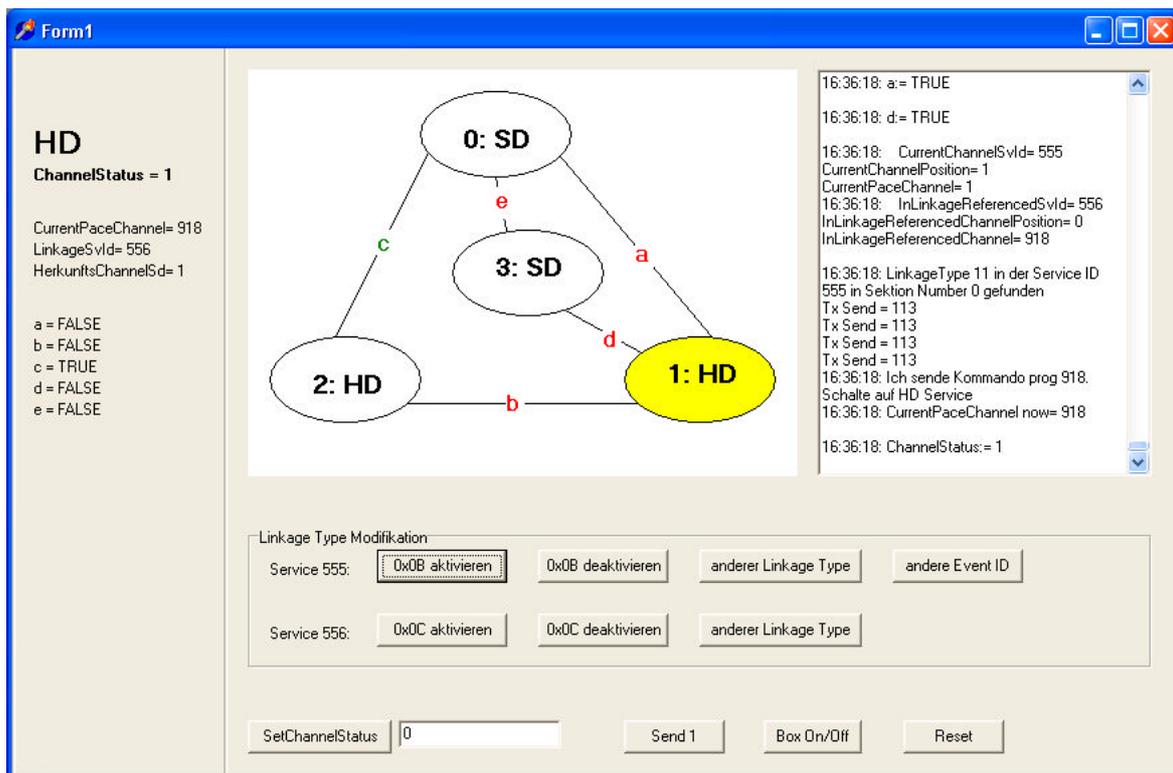


Abbildung 51: Screenshot der Bedienoberfläche des EIT-Analyse-Programms

5.4 Ausgewählte Details zu beschrifteten Lösungsansätzen

Bei der Realisierung des Demonstrators sind einige Schwierigkeiten aufgetreten, die jedoch alle im Laufe der Arbeit gelöst werden konnten. Im Folgenden werden diese Lösungen kurz vorgestellt.

5.4.1 Event ID

Die Überlegung war zunächst, beim Schaltvorgang von HD zurück auf SD die Event ID als Schaltkriterium zu benutzen, also zurückzuschalten, wenn die Event ID sich ändert bzw. ein neues Event beginnt. Dabei ergibt sich die Schwierigkeit, dass einer der verschiedenen SD-Herkunfts-Services möglicherweise dieselbe Event ID verwendet wie ein anderer Herkunft-Service. Das ist möglich, weil die Event IDs nur innerhalb eines Services eindeutig sein müssen. Dies bedeutet, dass, für den Fall dass das jeweilige HD-Event immer dieselbe Event-ID hat wie das SD-Event, die Event-IDs innerhalb des HD-Kanals nicht mehr eindeutig sind. Es könnte also passieren, dass (im HD-Service) eine Sendung des SWR dieselbe Event-ID hat wie die darauf folgende Sendung des MDR. Damit würde das

Programm nicht auf den SD-Kanal zurückschalten, wenn die Änderung der Event-ID das Schaltkriterium von HD zurück auf SD wäre.

Das Problem wurde so gelöst, dass die Event-ID nur noch für die Bedingung *e* (Schaltung von Status 3 nach Status 0, siehe 5.3.1) von Bedeutung ist. In diesem Fall werden aber auch nur SD-Event-IDs miteinander verglichen, sodass es für das Programm gleichgültig ist, wie die Event IDs im HD-Kanal sind. Viel eleganter ist es stattdessen, zu vergleichen, ob sich das DVB-Triple im Linkage Descriptor, das auf den jeweiligen Herkunftskanal verweist, ändert.

Mit dieser Lösung ist man praktisch unabhängig von der Event ID. Der Vorteil daran ist, dass ein unnötiger Schaltvorgang verhindert werden kann, wenn im HD-Kanal zwei Events desselben ‚Ursprungs-Services‘ aufeinander folgen, da nur noch verglichen wird, ob sich die Herkunft ändert, nicht aber, ob ein neues Event beginnt.

5.4.2 Austausch des HD-Receiver

Zunächst wurde in dem Demonstrator eine HD Digital-Receiver der Firma Pace verwendet, weil dieser bei 720p50 stabiler läuft und mit H.264 besser umschaltet als die später verwendete Humax-Box. Es gab jedoch große Probleme, die Settop-Box über den Infrarot-Sender zu steuern. Wie die Firma IRTrans per E-Mail bestätigte, sind bei den Infrarot-Codes zur Steuerung der Pace-Box die Impulse offenbar so kurz, dass sie an der Grenze der Möglichkeiten von Infrarot-Empfängern liegen. Die kurzen Infrarot-Kommandos, die Pace benutzt, sind offenbar die, die auch für den Datentransfer über kurze Distanzen, z.B. von einem Rechner zu einem Mobiltelefon, gedacht waren und nicht die langsameren, für längere Strecken und Fernbedienungen entwickelten Befehle [27]. Auf Grund der sehr kurzen Impulse konnten die Infrarot-Kommandos nicht zuverlässig detektiert werden, was zur Folge hatte, dass die Pace-Box unzuverlässig auf gesendete Befehle reagierte. Nach zahlreichen vergeblichen Versuchen, eine verlässlich funktionierende Remote-Datei zu erstellen, wurde beschlossen einen anderen HD-Receiver, eine Settop-Box der Firma Humax, für den Demonstrator zu verwenden.

Im Quelltext ist dagegen noch häufig die Bezeichnung *Pace*, beispielsweise als Teil eines Variablennamens, zu lesen. Da der Name der Variable aber keinen

Einfluss auf die Funktion des Programms hat und, trotzdem dass jetzt ein *Humax*-Receiver verwendet wird, auch mit der Bezeichnung *Pace* klar ist, was gemeint ist, wurde der Name im Quellcode nicht geändert.

5.4.3 Umstellung der Programmierung auf eine State-Machine

In einer der frühen Versionen des EIT-Analyse-Programms war die Programmierung noch nicht so klar (anhand eines State-Diagramms) strukturiert, sondern mit vielen stark ineinander verschachtelten Bedingungen aufgebaut. So war es zum einen recht schwierig, den Programmcode zu verstehen, zum anderen trat plötzlich die Situation ein, dass sich das Programm unvorhergesehen verhalten hat. In den vielen sich aufeinander beziehenden Bedingungen ist demnach der Überblick verloren gegangen, bzw. man hat bestimmte Zusammenhänge in der Programmierung übersehen. Zu diesem Zeitpunkt wurde die Entscheidung getroffen, die Programmierung noch mal völlig neu, aber diesmal klar strukturiert zu gestalten. Es wurde das State-Diagramm entwickelt und anhand dessen die Programmierung in States und Bedingungen umformuliert. Damit war die Übersicht über die einzelnen Abhängigkeiten wieder hergestellt, das Programm verhielt sich wie erwartet und konnte konsequent und strukturiert erweitert werden.

6 Zusammenfassung und Ausblick

6.1 Diskussion

Der für die Entwicklung des Empfänger-Demonstrators zugrunde gelegte Vorschlag des IRT für eine Erweiterung der Normen EN 300 468 und TR 101 211 (IRT-Proposal) erwies sich für eine Implementierung als zu unpräzise. Diese Ungenauigkeiten konnten beseitigt und das Verfahren konnte darüber hinaus verbessert werden. Dieses verbesserte Modell wurde zu einem lauffähigen Demonstrator programmiert und somit der Beweis für dessen Realisierbarkeit erbracht. Der Demonstrator verhält sich dabei fast genau so wie eine in den Receiver fest integrierte Funktion. Der Unterschied ist dabei lediglich die Tatsache, dass, aufgrund des in sich abgeschlossenen Systems, direkt in die Software des Receivers kein Eingriff erfolgen kann und die Steuerung über ein zusätzliches Infrarotmodul und einen PC erfolgen muss. Der Benutzer kann aber mit der Fernbedienung die Settop-Box wie gewohnt steuern. Dabei wurde das Ziel, das gleiche ‚Look and Feel‘-Erlebnis wie bei der Bedienung einer herkömmlichen Settop-Box, auch bei der Bedienung des Demonstrators zu erreichen, voll erfüllt. Die aufgetretenen Schwierigkeiten während der Programmierung konnten gelöst werden und somit ein für einen Demonstrator bestens geeignetes, lauffähiges Programm erstellt werden.

Ein Einsatz im realen Fernsehbetrieb ist möglich. Nach einer Implementierung einer solchen Funktion durch die Settop-Boxen-Hersteller und der Klärung der abschließend in Kapitel 6.2 genannten ‚Restfragen‘ kann ein solcher Gemeinschaftssender tatsächlich auf Sendung gehen. Diese Diplomarbeit hat das gezeigt. Natürlich ist zu diesem Zeitpunkt noch ungewiss, wie sich Hersteller und Rundfunkanstalten entscheiden, aber mit dieser Arbeit liegt für die Empfängerseite ein vollständiges Konzept vor, das bereits mit dem Demonstrator auf seine Realisierbarkeit hin überprüft wurde. Die Broadcaster sind schon mit Hilfe dieses Demonstrators in der Lage eine entsprechende (Test-)ausstrahlung zu starten, ohne auf die Verfügbarkeit entsprechender Empfänger angewiesen zu sein.

In der Zukunft, wenn sich HDTV mehr und mehr durchsetzt, die einzelnen Rundfunkanstalten mehr und häufiger in HDTV senden wollen und die

Finanzierung dies zulässt, könnte nach dem in dieser Arbeit beschriebenen Prinzip ein zweiter oder dritter Gemeinschaftskanal gestartet werden. Dann würden sich entweder entsprechend weniger Rundfunkanstalten einen Sender teilen oder die jeweiligen Events würden auf die verschiedenen Gemeinschaftskanäle verteilt. Beide Versionen sind realisierbar, da durch das im Linkage Descriptor enthaltene DVB-Triple eindeutig ist, wohin geschaltet werden soll. Somit ist das entworfene Szenario sogar zukunftssicher und es kann schrittweise entweder zu einem vollständigen Simulcast oder auch einer kompletten Abschaltung der SD-Kanäle entwickelt werden.

6.2 Schritte für die praktische Umsetzung

Anhand der vorliegenden Diplomarbeit liegt nun eine ausgereifte Dokumentation vor, welche Schritte für die Einführung von Simulcast für HDTV empfängerseitig notwendig sind. Für die Senderseite (dort lag nicht der Schwerpunkt dieser Arbeit) wurden durch die Arbeit folgende verbleibende Fragen, die bei der Aufnahme eines praktischen Betriebs abschließend geklärt werden sollten, identifiziert und formuliert:

- Welche Information soll in der EIT-following und der EIT-schedule übertragen werden? Die des SD- oder die des HD-Kanals? (zwei Lösungsvarianten wurden im Kapitel 4.3 beschrieben)
- Wie soll im Falle einer Störung von SD- oder HD-Service verfahren werden? (siehe Kapitel 4.4)
- Was passiert, wenn sich der Programmablauf bei einem der SD-Services kurzfristig ändert? (siehe Kapitel 4.4)
- Wie kann man senderseitig dafür sorgen, dass die Linkage Descriptoren immer zuverlässig und zeitlich exakt synchron gesendet werden? Die Signalisierung sollte ähnlich exakt sein, wie die Umschaltung zwischen 4:3 und 16:9.
- Die Umschaltung sollte möglichst schnell funktionieren. Sollte ein schwarzer Bildschirm während des Umschaltvorgangs unvermeidlich sein, sollte der Receiver eine entsprechende Meldung (beispielsweise ‚Es wird auf HDTV umgeschaltet‘) auf dem Display ausgeben.

Bis auf den ersten Punkt, der eine von den Programmgestaltern zu beantwortende Verfahrensfrage darstellt, geben die weiteren Punkte Orientierungshilfen zur Optimierung der Performance bei der praktischen Einführung.

Mit der in dieser Arbeit entwickelten Dokumentation, dem Demonstrators und der Klärung der hier genannten Fragen ist eine ausreichende Grundlage geschaffen, um den ‚teilweisen‘ Simulcast-Betrieb unmittelbar in die Praxis umzusetzen.

7 Abkürzungen und Begriffe

A

ARD Arbeitsgemeinschaft der öffentlich-rechtlichen Rundfunkanstalten der Bundesrepublik Deutschland: Zusammenschluss der Landesrundfunkanstalten.

ASI-Karte Asynchronous Serial Interface: Das AS-Interface ist ein Standard für die Übertragung von Transportströmen.

B

BAT Bouquet Association Table, Tabelle in der → SI

Batch-Datei Datei, die Aufgaben nacheinander bearbeitet. Batch heißt wörtlich übersetzt ‚Listenabarbeitung‘ oder ‚Stapelverarbeitung‘.

Blu-ray Disc neben der → HD-DVD ein designierter Nachfolger der DVD.

Boolesche Variable Element, das immer einen von zwei Werten annimmt: ‚wahr/falsch‘, ‚true/false‘ oder ‚1/0‘.

Bouquet Virtuelle Gruppierung mehrerer digitaler Programme, die sich auch über mehrere → Transponder erstrecken können, zu einem Gesamtangebot.

BR Bayerischer Rundfunk: Landesrundfunkanstalt für den Freistaat Bayern

Broadcaster Rundfunkbetreiber

C

C++ Objektorientierte Programmiersprache basierend auf der Programmiersprache C.

CAT Conditional Access Table, Tabelle in der → PSI

CC Continuity Counter: Stellt die zeitlich richtige Abfolge der Transportpakete im → TS sicher.

Coder Kodierer: Gerät, das eine Kodierung erzeugt.

Config-Datei	Konfigurationsdatei: Datei in die Konfigurationen für ein Programm eingetragen und/oder abgespeichert werden.
CRC	<u>C</u> yclic <u>R</u> edundancy <u>C</u> heck. Zyklische Redundanzprüfung: Prüfsumme, die eine Überprüfung der Daten auf fehlerfreie Übertragung ermöglicht.
<u>D</u>	
DC	<u>D</u> irect <u>C</u> urrent: Gleichstrom
DC-Block	Bauteil zum Einschrauben zwischen SAT-Kabel und Modulator. Der DC-Block verhindert ein Übertragen von DC-Spannungen auf nachfolgende Komponenten.
Delay	Englisches Wort für (Signal-) Verzögerung.
Delphi	Name einer von der Firma Borland entwickelten objektorientierten Programmiersprache, die aus Programmiersprache Pascal hervorgegangen ist.
DLF	<u>D</u> eutschland <u>f</u> unk, Hörfunksender.
DVB	<u>D</u> igital <u>V</u> ideo <u>B</u> roadcasting. Projekt zur Standardisierung von Verfahren zur Übertragung von digitalen Inhalten (Audio, Video, Zusatzdienste).
DVB-C	<u>D</u> igital <u>V</u> ideo <u>B</u> roadcasting – <u>C</u> able. DVB-Übertragung über Kabel.
DVB-S	<u>D</u> igital <u>V</u> ideo <u>B</u> roadcasting – <u>S</u> atellite. DVB-Übertragung über Satellit.
DVB-T	<u>D</u> igital <u>V</u> ideo <u>B</u> roadcasting – <u>T</u> errestrial. Terrestrische (=erdgebundene) Verbreitung der Fernsehsignale in der Atmosphäre.
DVB-Transportstrom	siehe Kapitel 3.1
DVB-Triple	Das DVB-Triple besteht aus drei Werten mit welchen jeder → Transportstrom weltweit eindeutig beschrieben ist: Original Network ID, Transportstrom ID und Service ID.

DW	<u>D</u> eutsche <u>W</u> elle, Rundfunkanstalt für den deutschen Auslandsrundfunk mit Sitz in Bonn.
<u>E</u>	
EBU	<u>E</u> uropean <u>B</u> roadcasting <u>U</u> nion, Zusammenschluss nationaler Rundfunkanstalten aus 54 Ländern mit Sitz in Genf.
EIT	<u>E</u> vent <u>I</u> nformation <u>T</u> able, Tabelle in der → SI.
EN	<u>E</u> uropäische <u>N</u> orm. Standards, die von einem der europäischen Standardisierungskomitees ratifiziert worden sind.
Encoder	→ Coder
EPG	<u>E</u> lectronic <u>P</u> rogram <u>G</u> uide: Elektronischer Programmführer, der im DVB-Datenstrom übertragen wird.
ES	<u>E</u> lementary <u>S</u> tream = Elementarstrom. Ein ES enthält nur Video- <i>oder</i> Audio- <i>oder</i> sonstige Daten.
ETR	<u>E</u> TSI <u>T</u> echnical <u>R</u> eport: Informatives Dokument des → ETSI, das von einem technischen Ausschuss zur Veröffentlichung freigegeben wird.
ETSI	<u>E</u> uropean <u>T</u> elecommunications <u>S</u> tandards <u>I</u> nstitute, eine der drei großen Normungsorganisationen in Europa.
<u>E</u>	
Free-TV	Frei empfangbares, unverschlüsseltes Fernsehen.
<u>G</u>	
<u>H</u>	
H.264	Oder MPEG-4/AVC: Standard zur Videokompression
HD	<u>H</u> igh <u>D</u> efinition: hohe Auflösung.
HD-DVD	neben der → Blu-Ray Disc ein mögliches Nachfolgeformat der DVD.
HDTV	<u>H</u> igh <u>D</u> efinition <u>T</u> elevision, engl. für hochauflösendes Fernsehen

Header	Paketkopf. Bits oder Bytes am Beginn eines Datenpakets vor dem eigentlichen Nutzsignalanteil.
Hex Dump	Hexadezimaler Speicherabbild. Darstellung des Speicherinhalts in Form von Hexadezimalzahlen.
HR	<u>H</u> essischer <u>R</u> undfunk, Landesrundfunkanstalt für das Bundesland Hessen.
Humax-Box	Eine → Settop-Box bzw. ein → Receiver der Firma Humax.
I	
ID	<u>I</u> dentifizierung, Identifikation: Merkmal oder Vorgang zum eindeutigen Erkennen eines Objekts.
If-Anweisung	Anweisung in der Programmierung, die nur unter einer bestimmten Bedingung (wenn...) ausgeführt wird.
IP	<u>I</u> nternet <u>P</u> rotokoll: Netzwerkprotokoll in Computernetzen.
IR	<u>I</u> nfrarot: nicht sichtbares Licht im Wellenlängenbereich von etwa 780 nm bis 1mm.
IRD	<u>I</u> ntegrated <u>R</u> eceiver (and) <u>D</u> ecoder: Integration von → Receiver und Decoder. Gemeint ist die → Settop-Box
IRT	<u>I</u> nstitut für <u>R</u> undfunk <u>t</u> echnik mit Sitz in München: Zentrales Forschungs- und Entwicklungszentrum der öffentlich-rechtlichen Rundfunkanstalten in Deutschland, Österreich und der Schweiz.
IRTrans	Firmen- und Produktname des Infrarotsteuerungssystems.
J	
K	
Kabel BW	Kabel Baden-Württemberg GmbH & Co. KG mit Sitz in Heidelberg, Kabelbetreiber in Baden-Württemberg.
L	
Linkage Descriptor	siehe Kapitel 3.6

LNB	<u>L</u> ow <u>N</u> oise <u>B</u> lock: Teil einer Satellitenempfangsanlage.
<u>M</u>	
MDR	<u>M</u> itteld <u>e</u> utscher <u>R</u> undfunk, Landesrundfunkanstalt für die Länder Sachsen, Sachsen-Anhalt und Thüringen.
MHP	<u>M</u> ultimedia <u>H</u> ome <u>P</u> lattform: interaktive Dienst-Plattform im digitalen Fernsehen.
MPEG-2	Standard zur Video- und Audiocodierung mit verlustbehafteter Kompression.
Multiplexen	Verfahren, um mehrere logische Kanäle gleichzeitig über einen physikalischen Kanal zu übertragen.
<u>N</u>	
NDR	<u>N</u> ord <u>d</u> eutscher <u>R</u> undfunk: Landesrundfunkanstalt für die Länder Hamburg, Niedersachsen, Schleswig-Holstein und Mecklenburg-Vorpommern
NIT	<u>N</u> etwork <u>I</u> nformation <u>T</u> able: Tabelle in der → SI
<u>O</u>	
<u>P</u>	
Pace-Box	Eine → Settop-Box bzw. ein → Receiver der Firma Pace.
Packet-Header	siehe → Header
PAL	<u>P</u> hase <u>A</u> lternating <u>L</u> ine, europäischer Farb-TV-Standard. 625 Zeilen interlaced, 25 Hz Vollbildfrequenz
PAT	<u>P</u> rogram <u>A</u> ssociation <u>T</u> able: Tabelle in der → PSI
Pay-TV	Verschlüsseltes ‚Bezahl-Fernsehen‘.
PES	<u>P</u> acketized <u>E</u> lementary <u>S</u> tream: paketierter, aufgeteilter → ES
PID	<u>P</u> acket <u>I</u> dentifier: Identifikation der Daten nach Programm, Video und Audio.
Playout	Das Ausspielen.
Playout-Server	Ausspielservers für Streams.
PMT	<u>P</u> rogram <u>M</u> ap <u>T</u> able: Tabelle in der → PSI

PSI	<u>P</u> rogram <u>S</u> pecific <u>I</u> nformation: Durch MPEG-2 definierte Zusatzdaten.
<u>Q</u>	
Quelltext	siehe → Sourcecode
<u>R</u>	
RAW	roh, Rohdatenformat.
RB	<u>R</u> adio <u>B</u> remen: Landesrundfunkanstalt für das Bundesland Bremen.
RBB	<u>R</u> undfunk <u>B</u> erlin- <u>B</u> randenburg: Landesrundfunkanstalt für die Länder Berlin und Brandenburg.
Receiver	Empfangsgerät zur Decodierung digital übertragener Fernseh- und Mehrwertdienste über Kabel, Satellit oder terrestrische Antenne.
RST	<u>R</u> unning <u>S</u> tatus <u>T</u> able: Tabelle in der → SI
<u>S</u>	
SAT	Abkürzung für Satellit
SD	<u>S</u> tandard <u>D</u> efinition: Standard-Auflösung.
SDT	<u>S</u> ervice <u>D</u> escription <u>T</u> able: Tabelle in der → SI
SDTV	<u>S</u> tandard <u>D</u> efinition <u>T</u> elevision, Sammelbegriff für analoge Fernseh- und Videonormen, deren Auflösungen geringer sind als bei -> HDTV
Sections	Daten mit begrenztem Umfang und mit einheitlichem Datenheader, die zyklisch erneut ausgestrahlt werden.
Service	Programmangebot in einem bestimmten Kanal, vgl. Kapitel 3.3.
Settop-Box	Gerät, das, an einen Fernseher angeschlossen, dem Benutzer zusätzliche Nutzungsmöglichkeiten bietet.
SI	<u>S</u> ervice <u>I</u> nformation: Durch DVB definierte Zusatzdaten.
Simulcast	Zeitgleiche Ausstrahlung des gleichen Inhalts auf weiteren, zusätzlichen Kanälen.

Sourcecode	In einer Programmiersprache geschriebener, für den Menschen lesbare Text eines Computerprogramms.
SR	<u>S</u> aarländischer <u>R</u> undfunk: Landesrundfunkanstalt für das Saarland.
State Diagramm	siehe Kapitel 3.7.
STB	siehe Settop-Box.
ST-Box	siehe Settop-Box.
SWR	<u>S</u> üd <u>w</u> est <u>r</u> undfunk: Landesrundfunkanstalt für die Länder Baden-Württemberg und Rheinland-Pfalz.
Syntax	Grammatik, um bei einer Programmiersprache erlaubte Konstruktionen festzulegen und unerlaubte auszuschließen.
I	
Tag	Englisches Wort für Kennzeichnung, Marke, Markierstelle
TCP	<u>T</u> ransmission <u>C</u> ontrol <u>P</u> rotocol: Vereinbarung auf welche Art und Weise Daten zwischen Computern ausgetauscht werden sollen.
TCP/IP	→ TCP-Protokoll, das auf dem → IP-Protokoll aufsetzt.
TDT	<u>T</u> ime & <u>D</u> ate <u>T</u> able: Tabelle in der → SI
Tektronix	Name einer Firma für Test- und Messgeräte.
Thread	Aktivitätsträger: In der Informatik bezeichnet ein Thread einen Ausführungsstrang beziehungsweise eine Ausführungsreihenfolge der Abarbeitung der Software.
Timeout	Zeitspanne, bevor ein Vorgang abgebrochen wird.
Timeslot	Zeitabschnitt innerhalb eines Übertragungsrahmens.
TOT	<u>T</u> ime <u>O</u> ffset <u>T</u> able: Tabelle in der → SI
TR	<u>T</u> echnical <u>R</u> eport: Von der → ETSI veröffentlichte informative Regelungen.
Transponder	Der Begriff Transponder ist zusammengesetzt aus den Begriffen Transmitter und Responder. Ein Fernsehsatellit benutzt den Transponder um das auf

einer Frequenz empfangene Signal auf einer anderen Frequenz wieder zurückzustrahlen.

Transportstrom

→ DVB-Transportstrom

TS

→ Transportstrom

U

Uplink

Satellitenkommunikation von der Erdstation zum Satelliten.

V

W

WDR

Westdeutscher Rundfunk: Landesrundfunkanstalt für das Land Nordrhein-Westfalen.

X

Y

Z

ZDF

Zweites Deutsches Fernsehen: Öffentlich-rechtliche Sendeanstalt in Deutschland

8 Literaturverzeichnis

- [1] HDTV in Deutschland – Statusbericht 2005, Deutsche TV-Plattform, AG: HDTV und Bildqualitätsverbesserung, Release 1.0
- [2] Kuhn, Kelin J.: HDTV Television – An Introduction
Internet: <http://www.ee.washington.edu/conselec/CE/kuhn/hdtv/95x5.htm>, 19.10.2006
- [3] Südwestrundfunk, Das Internetportal der ARD
Internet:
<http://www.ard.de/intern/organisation/rechtsgrundlagen/grundversorgung/-/id=54408/1cm440t/index.html>, 06.11.2006
- [4] Anixe HD Television GmbH & Co. KG
Internet: <http://www.anixehd.tv/index.php?id=sites/about>, 06.11.2006
- [5] European HDTV Media Company,
Internet: <http://www.euro1080.tv>, 06.11.2006
- [6] INFOSAT Euro-Media S.A., Satellite Frequency List, October 2006, Seite 21
- [7] Satfinder-Info, Sat-TV und Radio-Programmtabellen in Europa, Internet:
<http://www.satfinder.info/index.php>, 06.11.2006
- [8] Allgemeine Nachrichten, Fernseh- und Kinotechnische Gesellschaft (FKTG) e.V., Internet: <https://www.fktg.de/news.php?ID=4644>, 22.10.2006
- [9] Autorenteam IRT, Tagungs-CD, 22. FKTG-Jahrestagung, 15.-18. Mai 2006, Potsdam, Session 3: HD in der Einführung, Vortrag 15:
Konversionsprozesse bei der Einführung von HDTV
- [10] Technische Spezifikation ETSI TS 101 154, V1.7.7 (2005-1): Digital Video Broadcasting (DVB); Implementation guidelines for the use of Video and Audio Coding in Broadcasting Applications based on the MPEG-2 Transport Stream
- [11] Norm ISO/IEC 13818-1: 1994 (E):
Information Technology – Generic coding of moving pictures and associated audio: systems

- [12] Meyer-Schwarzenberger, G, srt [Hrsg.]: Lexikon Video- Audio- Netztechnik. – Heidelberg: Hüthig-Verlag, ISBN: 3-7785-2856-4
- [13] srt [Hrsg.]: Ausbildungshandbuch audiovisuelle Medienberufe, Band III. Heidelberg: Hüthig-Verlag, ISBN: 3-7785-2855-6, Seite 231-241, 39, 594
- [14] Schäfer, R.: DVB bei den öffentlich-rechtlichen Rundfunkanstalten – Eine Einführung. , 51. Jahrgang, Nr. 10/1997, Seiten 620 - 630.
- [15] Mäusl, R.: Fernsehtechnik – Vom Studiosignal zum DVB-Sendesignal, 3. Auflage. Heidelberg – Hüthig-Verlag, ISBN: 3-7785-2857-2, Seiten 87-100
- [16] ARD Play-Out-Center, RBB, Potsdam
Internet: <http://www.ard-digital.de/index.php?id=5597&languageid=1>,
20.10.2006
- [17] Transponder-Monitor: INFOSAT Euro-Media S.A., Heft 7, Juli 2006, Nr. 220, 20. Jahrgang, Seite 103.
- [18] Ruelberg, K.: Vorlesung ‚Grundlagen der Film- und Fernsehtechnik‘, 04.05.2004 und 11.05.2004, Fachhochschule Köln
- [19] Wikipedia – Die freie Enzyklopädie:
Internet: <http://de.wikipedia.org/wiki/Hexadezimal>, 06.11.2006
- [20] Norm EN 300 468, V1.3.1 (1998-02): Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB systems
- [21] Wikipedia – Die freie Enzyklopädie:
Internet: http://en.wikipedia.org/wiki/Finite_state_machine, 06.11.2006
- [22] Telemeter Electronic GmbH: Internet: http://www.telemeter.de/index.php/shop/produkte_bereiche/hf_mikrowellentechnik__1/dc_blocks, 10.11.2006
- [23] Wikipedia – Die freie Enzyklopädie:
Internet: <http://de.wikipedia.org/wiki/Fernbedienung>, 06.11.2006
- [24] Institut für Rundfunktechnik, München: Internet:
www.irt.de/IRT/produkte/IRT_DVB_Playout_Server_d.pdf, 20.10.2006
- [25] Müller, Marcus: IRTrans GmbH: IRTrans Benutzerhandbuch, Version 2.8, Zugl. Internet:
http://www.irtrans.de/download/Docs/Installation%20und%20Bedienung%20IRTrans_DE-A5.pdf, 06.11.2006

- [26] Borland Delphi Professional, Version 3.0 (Build 5.53): Delphi-Hilfe,
Hilfsthema: ‚Die Anweisung case‘
- [27] Daniel Tonks, Remote Central.com – Your universal remote control center
Internet: <http://www.remotecentral.com/features/premotes.htm>

9.2 IRT-Proposal

IRT / Klaus Merkel

V 0.1 / 19th September 2005

Linking HDTV events to SDTV services

Proposal for the extension of DVB specifications and guidelines

The proposed signalling allows to link HDTV events to SDTV services. This feature can significantly ease the migration from SDTV services to full HDTV services. It supports the continuation of SDTV services required for SD receiver compatibility reasons and avoids the need to have a full HD simulcast service as long as not 100% of the events is available in HD quality.

The proposed signalling is very "lightweight" and fully inline with the existing SI standard. It can be easily implemented in HD receivers and alternatively also be interpreted by broadcast MHP applications on HD MHP receivers.

The proposed changes of EN 300 468 and TR 101 211 are given below, changes are marked in red.

Extensions to EN 300 468:

„Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB systems,,

6 Descriptors

6.2 Descriptor coding

6.2.17 Linkage descriptor

...

Table 51: Linkage type coding

Linkage_type	Description
...	...
0x0B	HD service with related content
0x0C	SD service with related content
0x0D to 0x7F	reserved for future use
0x80 to 0xFE	user defined
0xFF	reserved for future use

Extensions to TR 101 211:

„Digital Video Broadcasting (DVB); Guidelines on implementation and usage of Service Information (SI)“

4 Rules of Operation

4.2 SI descriptor allocation and usage

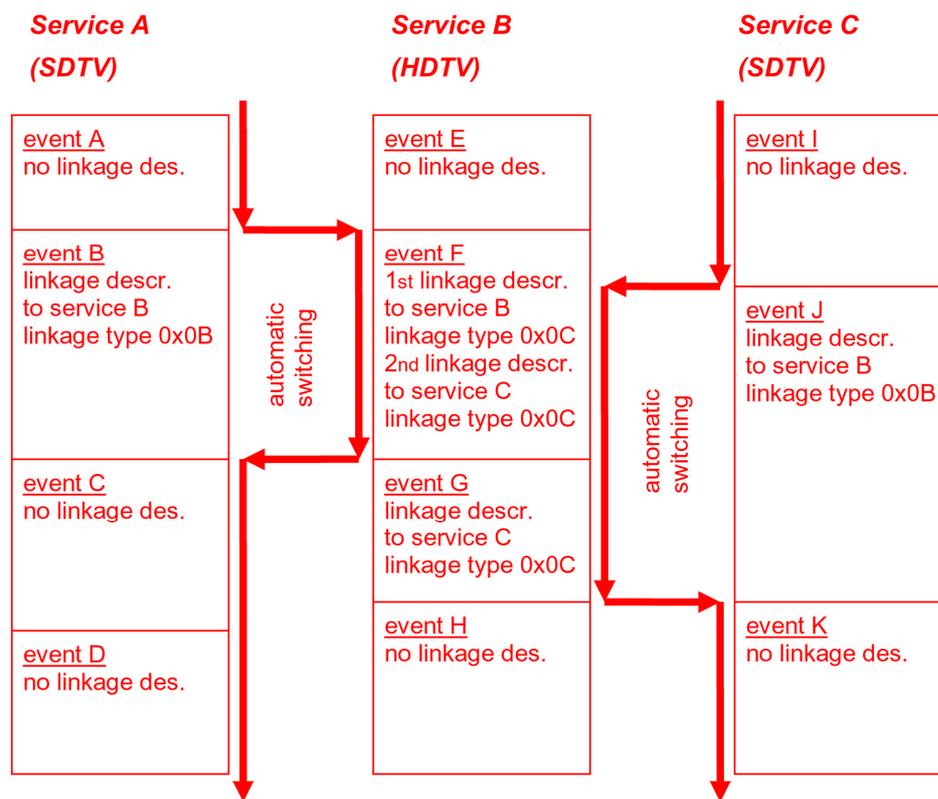
4.2.4 Descriptors of the Event Information Table

4.2.4.6 Linkage descriptor

This descriptor is used to give a link to another service. If it appears in this loop it links to a service that is attached to this event. Transmission of this descriptor is optional. If there is a `time_shifted_event_descriptor`, this descriptor is not allowed. More than one linkage descriptor is allowed in a loop. The meaning of the descriptor, when it occurs here, depends on the value of the `linkage_type`. If the `linkage_type` is:

- 0x01, the descriptor refers to a service that contains information about this event. An example of the intended use is for the IRD to switch to the information service when the user requests additional information about this event.
- 0x0B, the descriptor refers to a HD service that carries the same event as described in the actual SD service. If the corresponding event is indicated as "present", a HD capable receiver shall switch automatically to the indicated HD service but keep the information from which service the switching was initiated. An SD only receiver shall ignore a linkage descriptor with this linkage type.
- 0x0C, the descriptor refers to a SD service which carries the programme related to the current HD event. Interpretation is only relevant when a switching according to linkage type 0x0B was performed before. As long as the EIT present carries a linkage descriptor with linkage type 0x0C, the receiver shall remain on the current HD service. As soon as a new EIT present does not longer contain the reference to the SD service which initiated the switching to the actual service the receiver shall switch automatically back to the indicated SD. If the link back is not present immediately after switching to the HD service, the receiver shall not switch back but keep on the HD service until the link back becomes active and then disappears (this may be the case due to not fully synchronous EIT triggering). An SD only receiver shall ignore a linkage descriptor with this linkage type.

The HD/SD switching procedure according to linkage types 0x0B/0x0C is illustrated below:



The meaning of other values of linkage_type is not defined in this context. Note that the linkage_type does not indicate the service_type of the referenced service. An example of the intended use of the linkage descriptor is that an IRD user interface could include a mechanism like "info about the event" which would make the IRD tune to the linked service after the user initiated the mechanism.

9.3 Config-Datei

configchannel.ini

```
[Global]
ElementCount=10
PaceStartChannel=1
[0] ;Dies ist der HD Kanal
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 556
PaceChannel = 918
Frequenz = 123
[1] ;ARD
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 555
PaceChannel = 1
Frequenz = 123
[2] ;BR
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 22
PaceChannel = 2
Frequenz = 123
[3] ;WDR
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 23
PaceChannel = 3
Frequenz = 123
[4] ;hr
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 24
PaceChannel = 4
Frequenz = 123
[5] ;SWR
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 25
PaceChannel = 5
Frequenz = 123
[6] ;NDR
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 26
PaceChannel = 6
Frequenz = 123
[7] ;mdr
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 27
PaceChannel = 7
Frequenz = 123
[8] ;rbb
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 28
PaceChannel = 8
Frequenz = 123
[9] ;HD_selbstaendig
ChannelNwId = 1
ChannelTsId = 9999
ChannelSvId = 4
PaceChannel = 9
Frequenz = 123
```

Eidesstattliche Erklärung

Ich versichere hiermit die vorgelegte Arbeit in dem gemeldeten Zeitraum ohne fremde Hilfe verfasst und mich keiner anderen als der angegebenen Hilfsmittel und Quellen bedient zu haben.

München, den 30.11.2006

Birgit Schröter

Sperrvermerk

Die vorgelegte Arbeit unterliegt keinem Sperrvermerk.

Weitergabeerklärung

Ich erkläre hiermit mein Einverständnis, dass das vorliegende Exemplar meiner Diplomarbeit oder eine Kopie hiervon für wissenschaftliche Zwecke verwendet werden darf.

München, den 30.11.2006

Birgit Schröter