

Joachim Schwarz

## Die Anwendung des AES-24 ID Entwurfs auf Audiogeräte

November 1994

### Inhaltsverzeichnis

<b>1 Kurze Vorbemerkung zum Thema</b>	<b>1</b>
<b>2 Beschreibung der Geräte-Spezifikation</b>	<b>4</b>
<b>3 Das AES-24 Objektmodell</b>	<b>5</b>
3.1 Das Device-Modell . . . . .	8
3.2 Objekte und Befehle (Objects & Methods) . . . . .	9
<b>4 Das Message Format</b>	<b>13</b>
4.1 Der Network Header . . . . .	13
4.2 Sequence Number . . . . .	13
4.3 Object Method Handle . . . . .	14
4.4 Die Parameter-Werte (Parameter Value) . . . . .	15
4.5 Der Network Trailer . . . . .	15
<b>5 Die Kommunikation mit dem Terzband-Entzerrer</b>	<b>15</b>
5.1 Beispiel 1: Einstellung eines Entzerrer-Bandes . . . . .	15
5.2 Beispiel 2: Einstellung der Verstärkung . . . . .	17
<b>6 Schlußbemerkung</b>	<b>18</b>

### 1 Kurze Vorbemerkung zum Thema

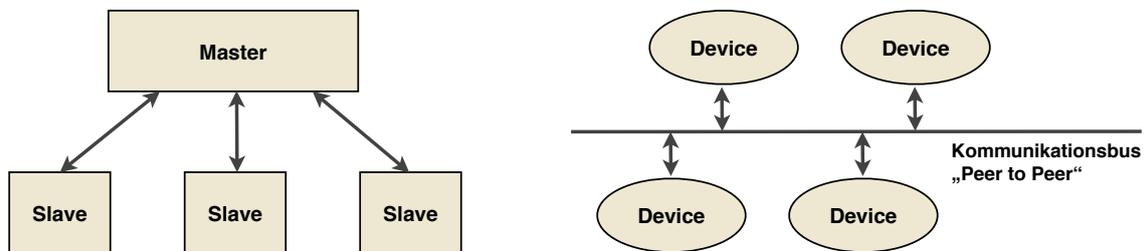
Dieser Bericht beschreibt, wie das vorläufige AES-24 ID-xxxx Protokoll auf einen typischen, graphischen 1/3-Oktav Entzerrer angewendet werden kann. Die Änderungen der sogenannten „Amber“-Revision wurden, wie bei der Sitzung der SC-10-2 Arbeitsgruppe vorgeschlagen, bereits entsprechend berücksichtigt. Das Ziel dieses Vortrages ist es, das Protokoll weitgehend verständlich zu erläutern und eventuelle erste Versuche im Labor zu ermöglichen. Protokolle sollen eine allgemein anzuwendende Form der Datenübertragung definieren. Die Erfordernisse der Industrie wie auch kostengünstige technische Lösungen sollen dabei berücksichtigt werden. Dies verhindert aber leider eine schnelle Vollendung dieses Standards. Wir bitten, diesen Umstand bei der Bewertung des nachstehenden Textes zu berücksichtigen.

Das AES-24 ID-xxxx Protokoll bietet einen neuen Ansatz, verschiedenste Audiogeräte mittels geeigneter Rechner fernsteuerbar zu machen. Ansätze zur Harmonisierung der Bemühungen in der Industrie

hat es in den letzten Jahren viele gegeben. Die kommerziell erfolgreichste, wenn auch leider in ihren Möglichkeiten sehr beschränkte Lösung stellte das MIDI-Protokoll dar. Mittlerweile ist dieses Protokoll gute 15 Jahre alt und basiert immer noch auf den Hardwarebeschränkungen der Zeit, zu der es entworfen wurde. Daher ist man bemüht, zügig einen neuen Standard ins Leben zu rufen, der wiederum herstellerübergreifende Gültigkeit besitzt.

Wie das leider bei der Erstellung eines Protokolls häufig der Fall ist, gibt es mittlerweile viele Ansätze, die aber meist nur proprietäre Insellösungen darstellen und von der Mehrheit - aus rein kommerziellen wie Prestige Gründen - abgelehnt werden. Grundsätzlich sind alle diese Lösungen schon vom Ansatz her nicht kompatibel. Daher ist eine gemeinsame Steuerung zur Zeit nicht möglich.

Schnelle und preiswerte Datenübertragung bis zu 130 MB/sec stehen heute zur Verfügung. Daher ist es nicht notwendig, eine Master-Slave Konfiguration zu implementieren. Vielmehr ist es heute möglich, dezentrale Intelligenz, bzw. das sogenannte Peer-to-Peer Design erfolgreich anzuwenden.



**Abbildung 1:** Vergleich Master-Slave mit Peer-to-Peer

Die AES (Audio Engineering Society) hat sich entschlossen, durch die Bildung von Arbeitsgruppen für eine schnellere Erstellung eines möglichen industrieweiten Protokolls zu sorgen. Dabei wird beabsichtigt, durch eine geeignete Struktur Weiterentwicklungen möglich zu machen und den Herstellern entsprechenden Spielraum zu lassen. Um dies zu erreichen, einigte man sich auf einen Ansatz, der sieben Ebenen innerhalb des Steuerprotokolls vorsieht. Ebene 1 stellt hierbei die physische Plattform, Ebene 7 die Applikationsoberfläche (z.B. Bedienfeld auf dem Computermonitor) dar.

Die Ebenen unterteilen sich wie folgt:

Ebene 1-4: Data Communication (betreut von der SC-10-1 Arbeitsgruppe)

Ebene 5-7: Application Protocol (SC-10-2 Arbeitsgruppe)

Ebene 7: Information (SC-10-3 Arbeitsgruppe)

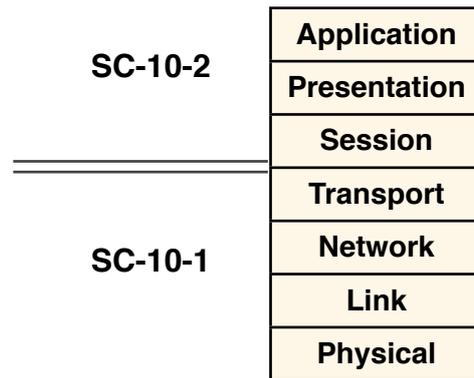


Abbildung 2: Das OSI 7-Layer-Modell

Der Vorteil dieser Aufteilung in 7 Ebenen liegt vor allen Dingen darin, daß jede Ebene im Bedarfsfalle durch neue Erkenntnisse ergänzt bzw. ausgetauscht wird. Hinzu kommt die Eigenschaft, den Kommunikationsaufwand verhältnismäßig gering halten zu können, da jede Ebene nur mit den beiden unmittelbar benachbarten Informationen austauschen muß.

Einen dem AES-Standard übrigens sehr verwandten Vorschlag bildet das MediaLink Protokoll der Firma Lone Wolf. Im Unterschied zum AES-Protokoll verwendet es nur 3 Ebenen. Dieser augenscheinliche Nachteil wird aber durch Kostenersparnis und ein wesentlich einfacheres Steuerprotokoll kompensiert.

Anhand eines Beispieles wollen wir uns den Unterschied des Protokolls zu dem gängigen PA-422 Steuercode betrachten. Die nachstehende Graphik erläutert auf anschauliche Weise die Eigenschaften beider Protokolle und den offensichtlichen Vorteil des AES-24ID:

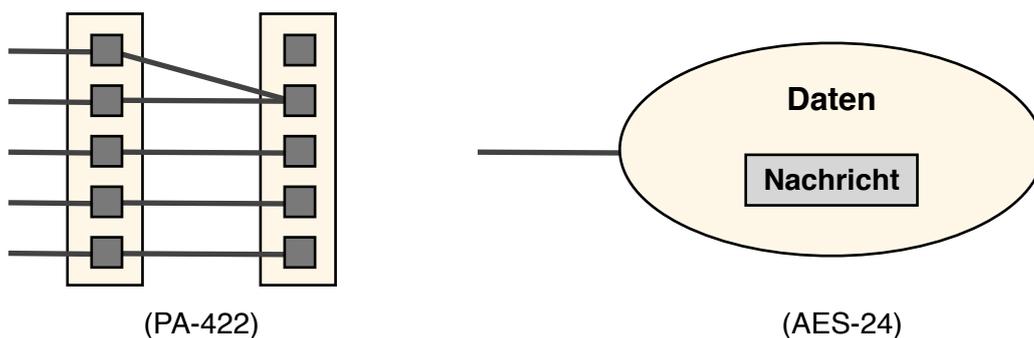


Abbildung 3: Datenaustausch AES-24 und PA-422

Der offensichtliche Nachteil liegt - abgesehen von der antiquierten Master-Slave Konfiguration mit einem Datenaustausch zwischen den Slaves ausschließlich über den Master - vor allen Dingen in der Tatsache, daß konstruktionsbedingt Informationspakete abgesandt werden. Diese Pakete enthalten überflüssigerweise alle Informationen zur Einstellung des jeweilig gemeinten Gerätes, anstatt nur der

zu ändernden Parameter. Hierdurch ergibt sich zwangsläufig eine erheblich größere Datenmenge, die zwar den Leiter (egal ob LWL oder „Twisted Pair“) wahrscheinlich nicht blockiert, wohl aber den Master erheblich in Anspruch nimmt. Hinzu kommt eine größere Fehleranfälligkeit, die bereits bei kleinen Ursachen (z.B. Format-Mismatch) zum Systemstillstand führt.

Das AES-24ID Protokoll hingegen spricht zuerst die gemeinten Objekte an. Erst dann werden die Änderungen, und nur diese (!) durchgegeben. Bei dem angestrebten Peer-to-Peer Design, welches die Vorteile der dezentralen Intelligenz voll ausnützt, wird daher die notwendige Datenrate dramatisch beschränkt. Hinzu kommt die äußerst positive Eigenschaft, daß Fehler einfach ignoriert werden. Wenn also z. B. das 30. Band eines 28-bandigen Entzerrers angesprochen wird, so wird diese fehlerhafte Nachricht einfach ignoriert. Der Vollständigkeit halber sei angefügt, daß das Protokoll eine Untersuchung der angeschlossenen Geräte vornimmt (bidirektionaler Informationsaustausch!) und daher von Anfang an solche fehlerhaften Daten nicht abgesandt werden können.

Leider sprengt der Umfang des Protokolls die Möglichkeiten eines halbstündigen Vortrages. Daher bitte ich Sie, sich bei der AES - Europa-Büro Brüssel - die verfügbare Version (die ja nur als Diskussionsgrundlage dienen kann) anzufordern. Wenn sich auch die Form noch ändern kann und wahrscheinlich wird, so ist es jedem ernsthaft Interessierten dringend zu empfehlen, bereits heute mit entsprechenden Vorbereitungen zu beginnen.

## 2 Beschreibung der Geräte-Spezifikation

Die Schönheit dieses Protokolls liegt in seiner Simplizität, die mit nur wenigen Vorgaben auskommt. Die Schlüsselideen werden wir durch die Anwendung auf den o.g. Entzerrer erarbeiten. Lassen Sie uns nun zuerst die zu steuernden Parameter festlegen. Bei dem zu entwerfenden Gerät handelt es sich um einen 2-kanaligen graphischen Terzband-Entzerrer.

- 28 Terzband ISO-Bandpassfilter, die jeweils um +/- 10dB in 0,5 dB-Schritten justierbar sind
- Ein zwischen 1kHz und 20kHz frei einstellbares Tiefpassfilter mit Ein/Aus-Schalter
- Ein zwischen 10 Hz und 200 Hz frei einstellbares Hochpassfilter mit Ein-/Aus-Schalter
- Eingangspegel zwischen +12 dB und -12 dB in 1 dB Schritten einstellbar
- Ausgangspegel zwischen +12 dB und -12 dB in 1 dB Schritten einstellbar
- Eingangspegel Spitzenwertmesser mit einer Integrationszeit von 1 bzw. 10 ms
- Ausgangspegel Spitzenwertmesser mit einer Integrationszeit von 1 bzw. 10 ms
- Bypass-Schalter zur direkten Leitung des Eingangssignales auf den Ausgang
- 2 unabhängige Kanäle mit jeweils den o.g. Ausstattungsmerkmalen
- 16 Preset-Speicher mit STORE und RECALL. Jedes Preset enthält den Zustand jeder Kontrolle, wie oben beschrieben.
- Taster/Schalter zur Rückführung aller Bänder eines Kanales in die „0 dB-Position“.
- Copy-Funktion, um die Einstellungen eines Kanals auf den anderen zu übertragen.
- „Remote Switch Interface“ (RSI), das mit einem externen vierpoligen Schalter verbunden ist, um Speicherinhalte abzurufen.

- Auswahlvorrichtung zur Festlegung, ob der RSI mit einem 4-Bit Binärschalter (z.B. einem Drehschalter mit 16 Positionen) oder vier Tastern verbunden ist.

Das Gerät wird mit den folgenden Arten von Befehlssätzen ferngesteuert bzw. überwacht:

- **Audio Parameter Access:** Diese Befehle fragen nach dem Status eines bestimmten Audioparameters bzw. bestimmen dessen Änderung. Die Audioparameter dieses Gerätes lauten: EQ-Bänder, Hoch- und Tiefpassfilter, Verstärkung, Anzeigen und „Bypass“.
- **Non-Audio Parameter Access:** Diese Befehle fragen das Gerät nach dem Status der anderen Parameter bzw. deren Änderung. Die „Non-Audio“ Parameter dieses Gerätes sind die RSI-Schalter bzw. deren Schaltzustände.
- **Group Access:** Diese Befehle fordern das Gerät dazu auf, den Status einer Gruppe von Parametern zu melden bzw. diese zu ändern. Im allgemeinen erfolgt die Festlegung von Gruppen willkürlich und umfasst sowohl Audio wie auch „Non-Audio“-Parameter. Das besprochene Gerät enthält folgende Parametergruppen: Die Anzeigen der Kanäle 1 und 2 (sie wurden zusammengefasst, damit sie von einem externen Kontrollfeld aus periodisch als Datenblock abgefragt werden können, um den Datenaustausch des Netzwerkes nicht zu behindern); Entzerrung Kanal 1 und Entzerrung Kanal 2 (zur Gruppe zusammengefasst, um eine komplette Entzerrung als einzigen Datenblock senden zu können); Memory Presets.
- **Utility Functions:** Diese Befehle führen die anwendungsspezifischen Funktionen des Gerätes aus (0 dB Stellung aller Regler und Copy).
- **Device Manager:** Diese Befehle unterstützen die verschiedenen „Manager“ des AES–24 Device-Modells.

In den nachfolgenden Abschnitten werden die Ausstattungsmerkmale und Befehle den Objekten und Methoden des AES–24 ID-xxxx Dokumentes zugeordnet. Leider bedingt die Kürze des Vortrages eine recht unvollständige Ausführung dieses Vorhabens. Wir hoffen jedoch, dem geneigten Leser einige Anhaltspunkte und Anregungen vermitteln zu können. An dieser Stelle machen wir noch einmal darauf aufmerksam, daß dieses Protokoll als Informationsdokument zu verstehen ist. Eine endgültige Version liegt leider noch nicht vor. Die AES SC–10–3 Arbeitsgruppe ist für Anregungen und Verbesserungsvorschläge jeder Art dankbar.

### 3 Das AES–24 Objektmodell

Dieser Abschnitt ordnet die Ausstattungsmerkmale, wie unter 2 beschrieben, dem jeweiligen Objektmodell der AES–24 zu. In diesem Modell wird jedes „Objekt“ des Equalizers durch seinen Namen, seine Klasse und die Objekt-ID (Identifikationsnummer) dargestellt. In unserem Beispiel wurden die Objekt IDs willkürlich gewählt und jedem Objekt zugeordnet:

Name	Klasse	Objekt ID
Chan 1, 31,5 Hz, EQ-Band	KFO	1
Chan 1, 40 Hz, EQ-Band	KFO	2
Chan 1, 50 Hz, EQ-Band	KFO	3
Chan 1, 63 Hz, EQ-Band	KFO	4
Chan 1, 80 Hz, EQ-Band	KFO	5
Chan 1, 100 Hz, EQ-Band	KFO	6
Chan 1, 125 Hz, EQ-Band	KFO	7
Chan 1, 160 Hz, EQ-Band	KFO	8
Chan 1, 200 Hz, EQ-Band	KFO	9
Chan 1, 250 Hz, EQ-Band	KFO	10
Chan 1, 315 Hz, EQ-Band	KFO	11
Chan 1, 400 Hz, EQ-Band	KFO	12
Chan 1, 500 Hz, EQ-Band	KFO	13
Chan 1, 630 Hz, EQ-Band	KFO	14
Chan 1, 800 Hz, EQ-Band	KFO	15
Chan 1, 1 kHz, EQ-Band	KFO	16
Chan 1, 1,25 kHz, EQ-Band	KFO	17
Chan 1, 1,6 kHz, EQ-Band	KFO	18
Chan 1, 2 kHz, EQ-Band	KFO	19
Chan 1, 2,5 kHz, EQ-Band	KFO	20
Chan 1, 3,15 kHz, EQ-Band	KFO	21
Chan 1, 4 kHz, EQ-Band	KFO	22
Chan 1, 5 kHz, EQ-Band	KFO	23
Chan 1, 6,3 kHz, EQ-Band	KFO	24
Chan 1, 8 kHz, EQ-Band	KFO	25
Chan 1, 10 kHz, EQ-Band	KFO	26
Chan 1, 12,5 kHz, EQ-Band	KFO	27
Chan 1, 16 kHz, EQ-Band	KFO	28
Chan 1, LPF	KFO	29
Chan 1, HPF	KFO	30
Chan 1, Input Level	Gain	31
Chan 1, Output Level	Gain	32
Chan 1, Input Meter	Sensor	33
Chan 1, Output Meter	Sensor	34
Chan 1, Bypass	Schalter	35
Chan 1, Meter Cont.	Container	36
Chan 1, Container	Container	37
Chan 1, RSI	frei wählbar	38
Group BDM		508
Device Manager		510

Name	Klasse	Objekt ID
Chan 2, 31,5 Hz, EQ-Band	KFO	43
Chan 2, 40 Hz, EQ-Band	KFO	44
Chan 2, 50 Hz, EQ-Band	KFO	45
Chan 2, 63 Hz, EQ-Band	KFO	46
Chan 2, 80 Hz, EQ-Band	KFO	47
Chan 2, 100 Hz, EQ-Band	KFO	48
Chan 2, 125 Hz, EQ-Band	KFO	49
Chan 2, 160 Hz, EQ-Band	KFO	50
Chan 2, 200 Hz, EQ-Band	KFO	51
Chan 2, 250 Hz, EQ-Band	KFO	52
Chan 2, 315 Hz, EQ-Band	KFO	53
Chan 2, 400 Hz, EQ-Band	KFO	54
Chan 2, 500 Hz, EQ-Band	KFO	55
Chan 2, 630 Hz, EQ-Band	KFO	56
Chan 2, 800 Hz, EQ-Band	KFO	57
Chan 2, 1 kHz, EQ-Band	KFO	58
Chan 2, 1,25 kHz, EQ-Band	KFO	59
Chan 2, 1,6 kHz, EQ-Band	KFO	60
Chan 2, 2 kHz, EQ-Band	KFO	61
Chan 2, 2,5 kHz, EQ-Band	KFO	62
Chan 2, 3,15 kHz, EQ-Band	KFO	63
Chan 2, 4 kHz, EQ-Band	KFO	64
Chan 2, 5 kHz, EQ-Band	KFO	65
Chan 2, 6,3 kHz, EQ-Band	KFO	66
Chan 2, 8 kHz, EQ-Band	KFO	67
Chan 2, 10 kHz, EQ-Band	KFO	68
Chan 2, 12,5 kHz, EQ-Band	KFO	69
Chan 2, 16 kHz, EQ-Band	KFO	70
Chan 2, LPF	KFO	71
Chan 2, HPF	KFO	72
Chan 2, Input Level	Gain	73
Chan 2, Output Level	Gain	74
Chan 2, Input Meter	Sensor	75
Chan 2, Output Meter	Sensor	76
Chan 2, Bypass	Schalter	77
Chan 2, Meter Cont.	Container	78
Chan 2, Container	Container	79
Speicher Manager		507
Signalfluß Manager		509
Device Container		511

BCM = **B**road**C**ast **M**anager

RSI = **R**emote **S**witch **I**nterface

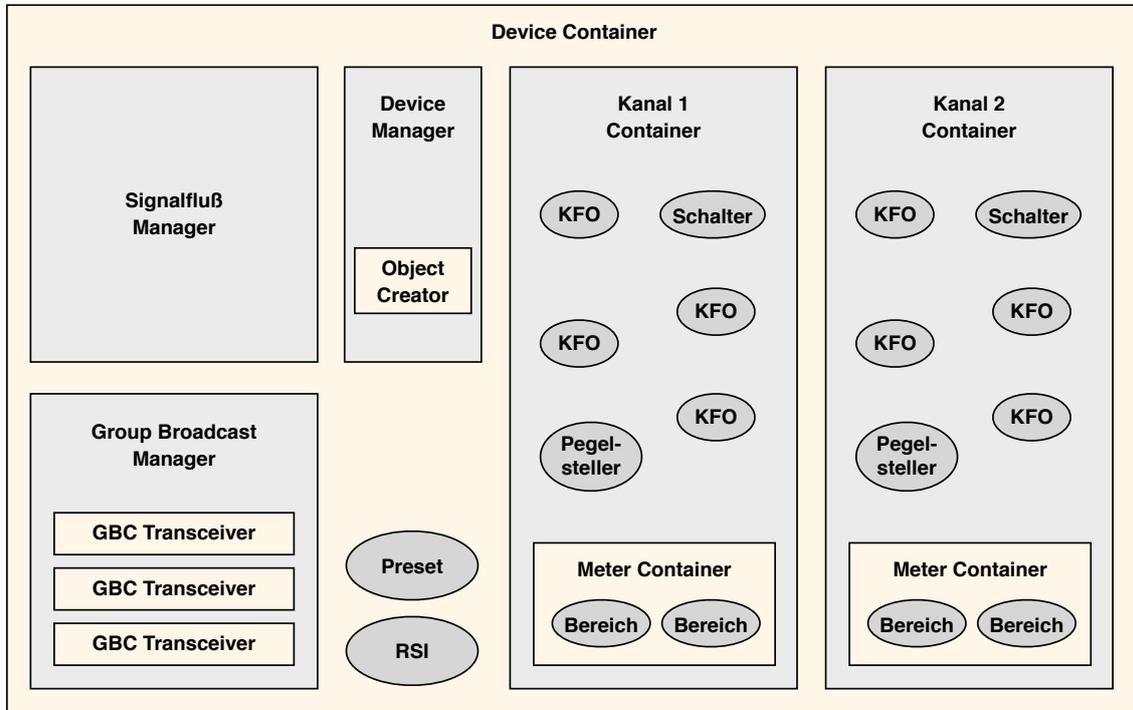
LPF = **L**ow **P**ass **F**ilter

HPF = **H**igh **P**ass **F**ilter

KFO = **K**lassisches **F**ilter **O**bjekt

### 3.1 Das Device-Modell

Das vorgeschlagene AES–24 Device-Modell wird in Abb. 4 weitgehend graphisch veranschaulicht.



**Abbildung 4:** Das AES–24 Device-Modell

Auszug aus dem AES–24 ID-xxxx Dokument:

Ein Device (Gerät) soll durch einen Container dargestellt werden, welcher die folgenden vier Hauptkomponenten enthält:

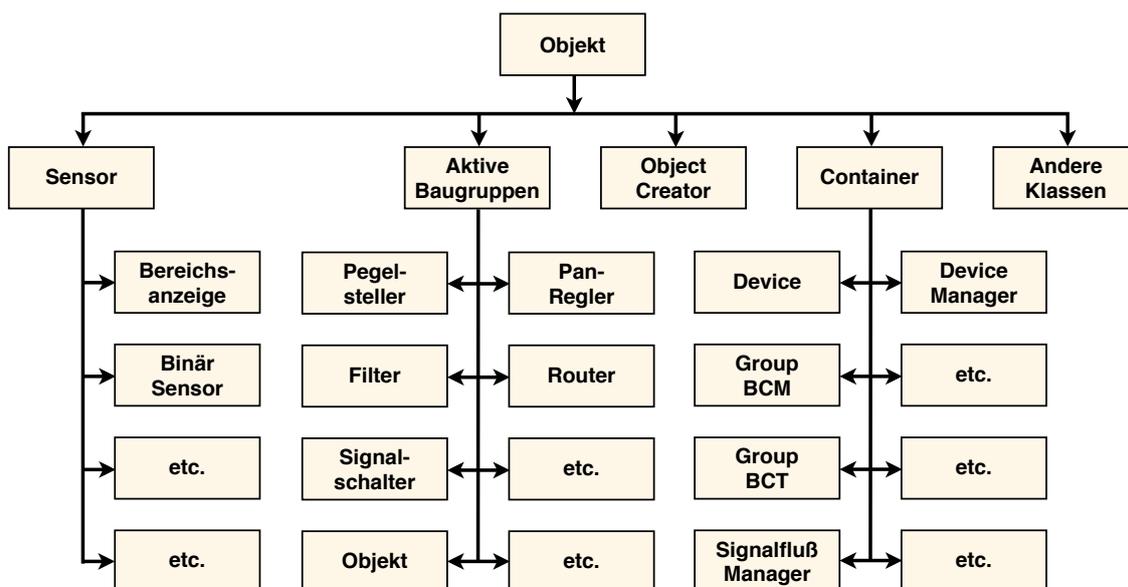
- Einen Device Manager, der eine Anzahl von Object Creators enthält. Ein Object Creator erzeugt Objekte wie z.B. Verstärkung, Schalter, Filter, etc. Einmal erzeugt, erscheinen diese Objekte auch innerhalb des Devices.
- Einen Signalfluß Manager, der bei Abfrage Informationen über die Verbindung von Objekten innerhalb eines Devices zurücksendet. Diesem Objekt kann auch die Möglichkeit gegeben werden, nach Anweisung neue Verbindungen zwischen softwaregestützten Objekten, also z. B. DSP-Algorithmen herzustellen.
- Einen Group Broadcast Manager, der einem Container entspricht, der mit Group Broadcast Transceivern ausgestattet ist. Jeder Group Broadcast Nummer ist genau ein Transceiver zugeordnet. In jedem Device soll mindestens ein Objekt einem Transceiver zugeordnet sein. Jeder Transceiver soll die Liste von Objekten enthalten, die Nachrichten empfangen, welche an ihre Group Broadcast Nummer gesandt wurden.

d) Keine oder mehrere Audio und „Non-Audio“ Objekte, die Verstärker, Schalter, Filter, etc. enthalten. „Non-Audio“ Objekte sollen Speicher, Zeitcode, etc. enthalten.

### 3.1.1 Graphische Darstellung des strukturellen Aufbaus der Objektklassen

Der strukturelle Aufbau der Objektklassen mit seiner streng hierarchischen Ordnung ist von erheblicher Bedeutung für das Verständnis des Steuerungskonzeptes. Leider ist auch hier eine ausführliche Behandlung nicht möglich. Daher sei noch einmal auf das jederzeit verfügbare Informationsdokument AES-24 ID-xxxx verwiesen.

Die graphische Darstellung soll einen Einblick in den Aufbau vermitteln. Wichtig zu wissen ist, daß das Objekt von oben nach unten hierarchisch gegliedert ist und alle nachfolgenden Klassen alles von der vorhergehenden „erben“. Hinweis: Dieser Aufbau ist nur durch den Einsatz verteilter Intelligenz möglich, da das Objekt selbst nicht nur Informationen weitergeben, sondern auch Aktionen bei untergeordneten Klassen starten und anweisen kann.



(BCT = BroadCast Transceiver)

Abbildung 5: Struktur der Objektklassen

## 3.2 Objekte und Befehle (Objects & Methods)

In den folgenden Sektionen werden die Befehlsnummern der Objekte tabelliert. Den Befehls-IDs wurden willkürliche Nummern zugeordnet, um die Erstellung eines Prototypen zu ermöglichen. Da die ersten vier Befehls-ID-Nummern in einem 1-Byte ASN.1 Datenwort untergebracht werden können, wurden sie den gebräuchlichsten Befehlen zugeordnet, um die Bandbreite des Netzwerkes so wenig wie möglich einzuschränken (diese Befehlsnummern wurden schraffiert dargestellt).

Hinweis: Die Ordnung der nachfolgenden Tabellen ist hierarchisch von oben nach unten gegliedert.

### 3.2.1 Konventionelles Filter Objekt (Objekt IDs: 1 - 30, 43 - 72)

Die Anweisungen (Befehle), die von der Eltern-Klasse (Actuator) „geerbt“ werden, lauten:

Befehl	Befehls-ID	Notizen
GetClassID	32	Request
SetClassID	31	Antwort
GetPanelName	30	z. B. Schalter
GetRole	29	Funktion in einem System
SetRole	28	Funktion in einem System
GetLock	27	Kann ich dich kontrollieren?
SetLock	26	Ja/Nein

Ein KFO (Konventionelles Filter Objekt) kennt 16 weitere Befehle:

Befehl	Befehls-ID	Notizen
GetType	16	Die Befehlssequenzen können sowohl an konventionelle, analoge wie auch an digitale Filter gesandt werden
SetType	15	
GetOrder	14	
SetOrder	13	
GetShape	12	
SetShape	11	
GetShapeParam	10	
SetShapeParam	9	
GetQ	8	
SetQ	3	
GetDesignFreq	7	
SetDesignFreq	2	
GetRefGain	6	
SetRefGain	5	
GetRelGain	4	
SetRelGain	1	

**3.2.2 Gain (Objekt IDs: 31, 32, 73, 74)**

Befehl	Befehls-ID	Notizen
GetClassID	32	Request
SetClassID	31	Antwort
GetPanelName	30	z. B. Eingangspegel
Get Role	29	Funktion in einem System
SetRole	28	Funktion in einem System
GetLock	27	Kann ich dich kontrollieren?
SetLock	26	Ja/Nein

Gain kennt 6 weitere Befehle:

Befehl	Befehls-ID	Notizen
GetRefGain	4	
SetRefGain	3	
GetRelGain	2	
SetRelGain	1	
GetResolution	5	
SetResolution	6	

**3.2.3 Pegelanzeige (Objekt IDs: 33, 34, 75, 76)**

Befehl	Befehls-ID	Notizen
GetClassID	32	Request
SetClassID	31	Antwort
GetPanelName	30	z. B. Eingangspegelanzeige
Get Role	29	Funktion in einem System
SetRole	28	Funktion in einem System
GetLock	27	Kann ich dich kontrollieren?
SetLock	26	Ja/Nein

Pegelanzeige kennt 19 weitere Befehle:

Befehl	Befehls-ID	Notizen
GetType	5	Was bist Du?
SetType	6	Du wirst zu. . .
GetDetValue	1	Nenne Deinen Wert!
GetDetMethod	7	Wie lautet Dein Befehlssatz?
SetDetMethod	8	Dein neuer Befehlssatz lautet. . .
GetDetParam	9	
SetDetParam	10	
GetUpThresh	11	
SetUpThresh	3	
GetLowThresh	12	
SetLowTresh	4	
GetRiseHyst	13	
SetRiseHyst	14	
GetFallHyst	15	
SetFallHyst	16	
GetReportRate	17	Messungen/Nachrichten pro Sekunde
SetReportRate	2	Messungen/Nachrichten pro Sekunde
GetRptMsgCnt	18	Get Report Message Count
SetRptMsgCnt	19	Set Report Message Count

### 3.2.4 Schalter, Meter Container, RSIs, Memory Manager, Group Broadcast Manager, Signalfuß Manager und Device Manager

Die Tabellen für o. g. Befehlssätze sind entsprechend den bereits gelisteten strukturiert. Da es sich bei diesem Ansatz um willkürlich festgelegte Befehls-IDs handelt, ersparen wir uns, den noch verbleibenden Rest hier aufzulisten.

## 4 Das Message Format

Das allgemeine Aussehen eines AES–24 Befehls lautet wie folgt:

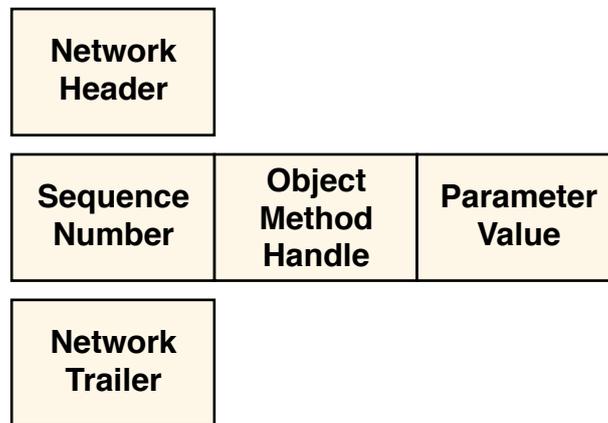


Abbildung 6: AES–24 Befehl

(Dieses Format entspricht der sogenannten „AmberVersion der SC–10–2, vereinbart auf der letzten AES-Sitzung in Dallas, April 1994)

NB.: Nachstehend werden wir versuchen, die englischen Begriffe weitgehend einzudeutschen. Allerdings gelingt dies heutzutage nur noch in eingeschränktem Maße. Wir bitten Sie, uns die manchmal gleichzeitige Verwendung beider Sprachen nachzusehen.

### 4.1 Der Network Header

Dieser Datenblock ist den Datensequenzen angehängt, die den Netzwerkebenen der AES–24 Applikation zu Grunde liegen. Die Daten variieren von Netzwerk zu Netzwerk und auf Grund des Umfangs soll hier nicht weiter darauf eingegangen werden.

### 4.2 Sequence Number

Jede Nachricht wird mit dieser Nummer zu Identifikationszwecken gekennzeichnet. Sie ist besonders hilfreich um korrespondierende Abfragen (Query-messages = „Get...“) und Antworten („Set...“) zusammenzufügen. Die Sequence Numbers müssen nicht notwendigerweise in der entsprechenden Reihenfolge übertragen werden (da sie „willkürlich“ sind). Es ist aber darauf zu achten, daß jedem „Set...“ Befehl/Nachricht, welcher auf einen entsprechenden „Get...“ Befehl folgt, die entsprechende „Get...“ Sequence Number beigeordnet ist, um beide miteinander zu verknüpfen bzw. richtig zu interpretieren.

### 4.3 Object Method Handle

Das sogenannte Object Method Handle (OMH) spezifiziert das Ziel der Nachricht. Es enthält sowohl den Objekt Index als auch die Befehls-ID (Method ID). Dem OMH entspricht ein 1, 2 oder 3-Byte Datenwort, welche im sogenannten ASN.1 Format dargestellt wird.

#### 4.3.1 Das allgemeine ASN.1 Format

Nachstehend finden Sie das allgemeine ASN.1 Format für 1-, 2- und 3-Byte Datenwörter:

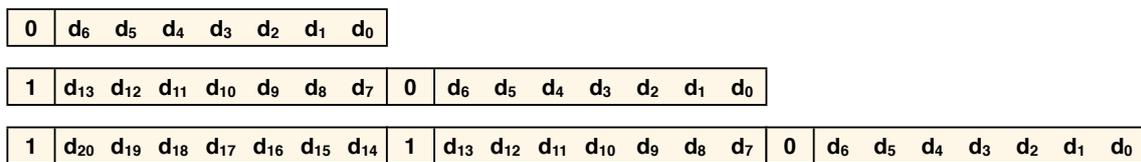


Abbildung 7: ASN.1 Format

Grundsätzlich legt das MSB (Most Significant Bit) fest, ob das nachfolgende Byte das letzte des Datenblocks darstellt. D. h., wurde das MSB auf eine logische „0“ gesetzt, so handelt es sich um das letzte Byte des Datenblocks. Ist das MSB auf „1“ gesetzt, folgen noch weitere Datenbytes. Um ein vollständiges Datenwort zu bilden, werden alle Bytes innerhalb eines Block miteinander verkettet, nachdem die MSB Flags eliminiert wurden.

#### 4.3.2 Die ASN.1 Kodierung des Object Method Handle (OMH)

Das Object Method Handle beschreibt, um welche Art Nachricht es sich handelt und welches Objekt gemeint ist. Es handelt sich dabei um einen einfachen Block mit zwei Feldern: Dem Objekt Index des angesprochenen Objektes und der Befehls-ID (Method ED) der entsprechenden Anweisung. Die Aufteilung der Bits zwischen Objekt Index und Befehls-ID lauten für die jeweiligen Formate wie folgt:

1-Byte-Format:

Objekt Index	Befehls-ID
d <sub>6</sub> - d <sub>2</sub>	d <sub>1</sub> - d <sub>0</sub>
Werte: 0–31	Werte: 0–3

2-Byte-Format:

Objekt Index	Befehls-ID
d <sub>13</sub> - d <sub>5</sub>	d <sub>4</sub> - d <sub>0</sub>
Werte: 0–511	Werte: 0–31

3-Byte-Format:

Objekt Index	Befehls-ID
$d_{20} - d_8$	$d_7 - d_0$
Werte: 0–8191	Werte: 0–255

Für kleine Werte ist das 1-Byte-Format das effizienteste, da es als kürzeste Version auch den geringsten „Overhead“ mit sich führt. Daher ist es angebracht, den niedrigsten Werten die am häufigsten gebrauchten Objektindizes und Befehls-IDs zuzuweisen. Diese Maßnahme verringert den notwendigen Datenfluß im Netzwerk außerordentlich.

#### 4.4 Die Parameter-Werte (Parameter Value)

Das Parameter-Wertefeld beinhaltet die Werte des Befehls. Der Datentyp des Wertefeldes variiert mit dem beigeordneten Befehl (im vorhergehenden Byte enthalten). Das AES–24 ID-xxxx Informationsdokument spezifiziert die entsprechenden Datentypen.

#### 4.5 Der Network Trailer

Dieser Datenblock ist den Datensequenzen angehängt, die den Netzwerkebenen der AES–24 Applikation zu Grunde liegen. Die Daten variieren von Netzwerk zu Netzwerk und auf Grund des Umfangs soll hier nicht weiter darauf eingegangen werden.

### 5 Die Kommunikation mit dem Terzband-Entzerrer

In dieser Sektion wollen wir versuchen, mit Hilfe einiger Beispiele die Befehlsstruktur zu dokumentieren. Auch hier sind wir durch die Kürze der Zeit gezwungen, auf Vollständigkeit zu verzichten.

#### 5.1 Beispiel 1: Einstellung eines Entzerrer-Bandes

**Befehl:** „Stelle im Kanal 1 das 1 kHz-Band auf +6 dB“

Ein Entzerrer-Band wird dadurch eingestellt, daß dem KFO ein „SetRelGain“-Befehl gesandt wird. Die Kodierung dieser Befehlssequenz lautet:

**OMH** (Object Method Handle)

Objekt Index des 1 kHz Bandes im Kanal 1: 16

Befehls-ID (Method ID) von „SetRelGain“: 1

Wir können nun das ASN.1 1-Byte-Format verwenden, um das OMH zu kodieren, da der Objekt-Index dieses Entzerrer-Bandes kleiner als 32 ist und die Befehls-ID (Method ID) kleiner als 4. Daher lautet das OMH:

$$\text{OMH} = (\text{Objekt Index}) * 4 + \text{Befehls-ID} = (16 * 4) + 1 = 65 \text{ (hex \$41)}$$

### Parameter-Value

Das Parameter-Value für den „SetRelGain“ ist vom Typ „Dezibel“ (siehe auch Sektion 10.5.6 des AES–24 ID-xxxx Entwurfes). Der dezimale Datentyp kann durch 8- oder 16-Bit Wörter dargestellt werden. Die Werte bewegen sich dabei zwischen +28/–100 dB (8 Bit) oder +/- 256 dB (16-Bit). Für unser Beispiel soll das 8-Bit Format genügen.

+6 dB im 8-Bit Dezimaltyp entspricht Hexadezimal: \$0C (siehe auch Sektion D.4 im AES–24 ID-xxxx Entwurf).

### Sequence Number

Diese Nummer kann willkürlich zugeteilt werden. In unserem Beispiel wollen wir Hexadezimal 255 = hex \$FF benutzen.

### Die vollständige Befehlssequenz

Die vollständige Befehlssequenz wird durch Aneinanderreihung von Network Header, Sequence Number, OMH, Parameter Value und Network Trailer gebildet.

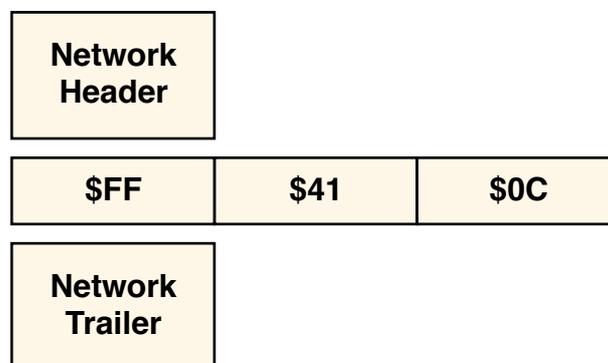


Abbildung 8: Vollständige Befehlssequenz

**Befehl:** „Stelle im Kanal 2 das 1 kHz-Band auf +6 dB“

Dieser Befehl gleicht dem vorhergehenden mit der Ausnahme, daß diesmal der Objekt Index größer als 31 ist, und daher das ASN.1 2-Byte-Format zur richtigen Kodierung notwendig ist. Die Kodierung gestaltet sich wie folgt:

### OMH

Objekt Index des 1 kHz Bandes im Kanal 2: 58

Befehls-ID (Method ID) von „SetRelGain“: 1

Der Objekt Index liegt innerhalb der Bits  $d_{13} - d_5$

Die Befehls-ID (Method ID) liegt innerhalb der Bits  $d_4 - d_0$

Das komplette, 2-Byte OMH lautet: \$8E \$41

### Parameter Value

Der Parameter-Wert ist der gleiche wie im vorhergehenden Beispiel: \$0C

### Sequence Number

Diese Nummer darf willkürlich festgelegt werden. In unserem Beispiel wiederum: \$FF

### Die vollständige Befehlssequenz

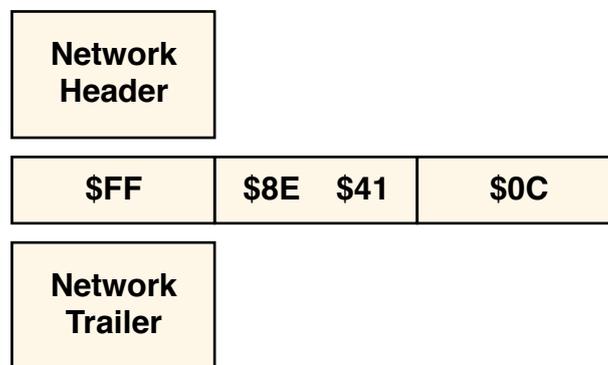


Abbildung 9: Vollständige Befehlssequenz

## 5.2 Beispiel 2: Einstellung der Verstärkung

**Befehl:** Stelle im Kanal 1 die Verstärkung auf  $-10$  dB

Die Verstärkung wird eingestellt, indem man dem Verstärkungsobjekt (Gain Object) einen „SetRelGain“ Befehl sendet. Die Kodierung gestaltet sich wie folgt:

### OMH

Objekt Index der Verstärkungseinstellung im Kanal 1: 31

Befehls-ID (Method ID) von „SetRelGain“: 1

Wir können das ASN. 1 1-Byte-Format verwenden:

$$\text{OMH} = (\text{Objekt Index}) * 4 + \text{Befehls-ID} = (31 * 4) + 1 = 125 \text{ (hex } \$7D)$$

### Parameter Value

Das Parameter-Value für den „SetRelGain“ ist vom Typ „Dezibel“ (siehe auch Sektion 10.5.6 des AES–24 ID-xxxx Entwurfes). Der dezimale Datentyp kann durch 8- oder 16-Bit Wörter dargestellt werden. Die Werte bewegen sich dabei zwischen  $+28/-100$  dB (8-Bit) oder  $\pm 256$  dB (16-Bit). Für unser Beispiel soll das 8-Bit Format genügen.

–10 dB im 8-Bit Dezimaltyp entspricht Hexadezimal: \$EC (siehe auch Sektion D.4 im AES–24 ID-xxxx Entwurf).

### Sequence Number

Diese Nummer darf willkürlich festgelegt werden. Hier: \$FF

### Die vollständige Befehlssequenz

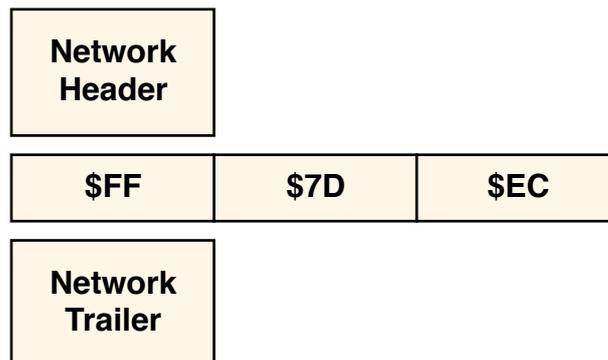


Abbildung 10: Vollständige Befehlssequenz

## 6 *Schlußbemerkung*

Wie bereits eingangs erwähnt, handelt es sich bei dem AES–24 ID-xxxx Informationsprotokoll um einen Entwurf. Dieser Entwurf ist längst noch nicht in allen Punkten geklärt. Sinn dieses Vortrages ist es, diesen lobenswerten Ansatz einem möglichst breiten Fachpublikum zu präsentieren, um so viele Verbesserungsvorschläge wie möglich zu erhalten. Hier bietet sich die Gelegenheit, anders als in vielen bekannten Beispielen, einen Standard von Anfang an kennenzulernen und eigene aus der Erfahrung gewachsene Vorschläge zu unterbreiten.

Dieser Vortrag soll und kann keine vollständige Information über das sehr umfangreiche Protokoll geben. Vielmehr erhoffen wir uns eine fruchtbare Zusammenarbeit, bei der es vielleicht endlich gelingen könnte, proprietären Insellösungen aus dem Weg zu gehen.