

---

# **Protokollanalyse an digitalen Schnittstellen mit den Audio Analysatoren UPL oder UPD**

---

Application Note 1GA15\_1D

Tilman Betz, 9/94  
Neufassung 8/96

Änderungen vorbehalten

Produkte:

**Audio Analyzer UPL**

**Audio Analyzer UPD**



**ROHDE & SCHWARZ**

## 1. Zusammenfassung

Für die Übertragung digitaler Audio-Daten werden verschiedene Datenformate verwendet. Sowohl das im professionellen Bereich benutzte AES/EBU-Format, als auch das im Konsumerbereich übliche S/P DIF-Format übertragen die Audio-Daten für die beiden Kanäle im Zeitmultiplex. Zusätzliche Bits im Datenstrom tragen ergänzende Informationen bzw. dienen der Fehlerprüfung. Die Erzeugung bzw. die Analyse dieser Protokolldaten mit den Audio Analysatoren UPD oder UPL ist das Thema dieser Applikationsschrift. Ein Auszug aus der AES3 informiert über die grundlegenden Definitionen.

## 2. Protokoll-Analyse und Generierung

In der digitalen Audio-Technik sind verschiedene Datenformate für die Signalübertragung genormt. Der Standard für die professionelle Audio-Technik ist die sogenannte AES/EBU-Schnittstelle über die Stereosignale mit bis zu 24 bit Auflösung übertragen werden können. Ein ähnliches Format, jedoch mit abweichenden Stecksystemen und elektrischen Anschlußwerten hat das in der Consumer-Industrie verwendete S/P DIF-Format (Sony / Philips digital interface format).

Die eigentliche Übertragung des Audio Signals erfolgt im Zeitmultiplex für beide Stereokanäle zusammen mit allen notwendigen Zusatzinformationen zur Synchronisation usw.. Das Datenformat bietet aber noch Platz für darüber hinausgehende Informationen und Fehlerprüfungen. Der grundsätzliche Aufbau des genormten Datenformates (AES3, siehe Auszug aus der Norm im Anhang) wird hier kurz beschrieben:

Das digitalisierte Audiosignal wird mit einer festgelegten Abtastrate, der sogenannten sample rate übertragen, genormt sind die Abtastraten 32 kHz, 44.1 kHz und 48 kHz je nach gewünschter Qualitätsforderung an den übertragbaren Tonfrequenzbereich. Mit dieser Abtastrate wird im Multiplex jeweils der linke und der rechte Audiokanal übertragen, ein sogenanntes Frame besteht jeweils aus dem Subframe für den linken Kanal und einem Subframe für den rechten Kanal, jedes dieser Subframes hat eine Länge von 32 bits. Bis zu 24 bits werden davon für das eigentliche Audiosignal verwendet, 4 bits bilden die sogenannte Präambel und dienen der Synchronisierung, 1 bit kennzeichnet die Gültigkeit des Audiosignals und 1 bit dient der Paritätsprüfung des gesendeten Wortes. Es verbleiben noch zwei freie bits in jedem Subframe, das sogenannte Channel-Status-Bit und das User-Bit. In diesen bits kann eine Zusatzinformation übertragen werden, die für das Channel-Status-Bit weitgehend genormt und für das User-Bit freigestellt ist.

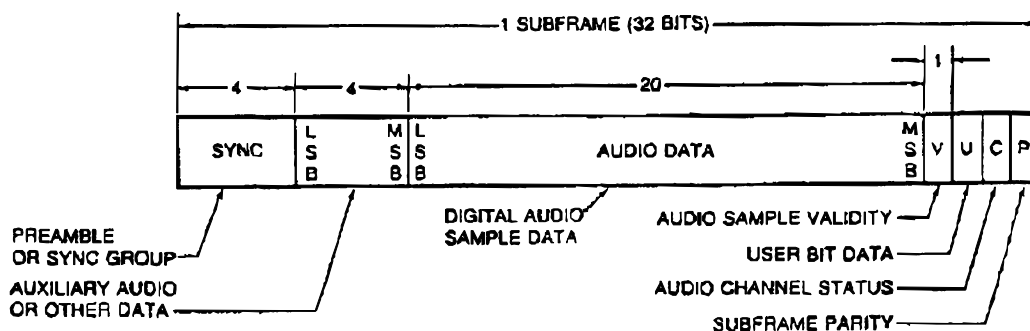


Bild 1: Subframe format

Diese Informationen werden also bitweise mit jedem Audiosample übertragen, getrennt für den linken und den rechten Kanal, die Bitrate entspricht der Abtastrate des Audiosignals also z.B. 48 kbit/s bei 48 kHz Abtastrate.

Für die Channel-Statusinformation werden jeweils 192 aufeinanderfolgende Channel-Status-Bits zusammengefaßt und als 24 Worte zu je 8 bit dargestellt, die genaue Definition der einzelnen Channel Statusworte ist aus dem Auszug aus der Norm AES3 im Anhang zu entnehmen. Die gesamte Channel Statusinformation hat damit eine Wiederholrate von Abtastrate / 192, also z.B. bei einer Abtastrate von 48 kHz eine Wiederholrate von 250 Hz. Es stehen parallel immer die Channel Statusinformation für den linken und den rechten Kanal zur Verfügung, diese Informationen können, müssen aber nicht identisch sein.

Die einzelnen Bytes (Worte) der Channel Statusinformation haben folgende Bedeutung (Auszug aus der AES3):

Byte	Bit 0	1	2	3	4	5	6	7
0	Use of channel status channel	Audio / nonaudio use	Audio signal emphasis			Locking of source sample frequency	Sampling frequency	
1	Channel mode				User bit management			
2	Use of auxiliary sample bits			Source word length and source encoding history		Reserved		
3	Future multichannel function description							
4	Digital audio reference signal		Reserved					
5	Reserved							
6	Alphanumeric channel origin data							
7								
8								
9								
10	Alphanumeric channel destination data							
11								
12								
13								
14	Local sample address code (32-bit binary)							
15								
16								
17								
18	Time-of-day sample address code (32-bit binary)							
19								
20								
21								
22	Reliability flags							
23	Cyclic redundancy check character							

Bild 2: Channel Status Datenformat

Die Audio Analytoren UPL und UPD können neben dem eigentlichen Audiosignal auch diese Zusatzinformationen generieren und analysieren. Außerdem können - allerdings nur mit dem UPD - gezielt fehlerhafte Frames erzeugt werden, um die Verträglichkeit angeschlossener Komponenten zu testen: Es ist möglich, Parityfehler in beliebiger Folge zu erzeugen, genauso wie Block-Folgefehler, Sequenz-Fehler oder auch Checksummenfehler (CRC).

Da die Erzeugung spezieller Bitfolgen mit der in der Norm festgelegten Bedeutung auf der Bit-Ebene relativ aufwendig und unübersichtlich ist, bieten UPL und UPD auch die Möglichkeit, diese Bitfolgen bereits codiert nach ihrer Bedeutung zu generieren und zu analysieren. Diese Codierung bzw. Decodierung wird mit sogenannten Panelfiles erreicht d.h. auf Wunsch wird das Bedienpanel im Generator bzw. des Analysators im UPL/UPD um die Channel-Daten-Darstellung erweitert. Für die genormten Formate sind diese Panelfiles den Audio Analysatoren verfügbar, für beliebige Formate kann der Anwender ein von ihm frei definierbares Bedienpanel für die Channel-Status-Daten zusammenstellen. Außerdem sind alle Daten natürlich auch direkt in binärer oder hexadezimaler Form darstellbar.

Im Folgenden wird die Generierung und die Analyse von Channel Statusdaten bzw. die Erzeugung der zugehörigen Bedienpanels beschrieben:

### 3. Generierung der Protokoll-Daten

#### AES/EBU Protokoll Definition

Bei Wahl der standardisierten Digitalausgänge erscheint im Generatorpanel ein zusätzlicher Abschnitt PROTOCOL. In diesem Abschnitt sind die Befehle zur Definition der Protokollinformation (Channel-Status-Daten, User-Daten, Validity, Parity) zusammengefasst.

Hinweis: Im Vergleich zum Audio Analyzer UPL stellt der Audio Analyzer UPD umfangreichere Möglichkeiten zur Erzeugung von Fehlersignalen zur Verfügung. Im folgenden wird auf diese Unterschiede hingewiesen.

<b>Valid Chan</b>	<p>stellt die Gültigkeitskennung innerhalb des AES -EBU Datenstromes ein.  <b>NONE:</b> Validity-Bit nie gesetzt  <b>1:</b> Validity-Bit nur im Kanal 1 (= links, A) gesetzt (erscheint nur bei UPD)  <b>2:</b> Validity-Bit nur im Kanal 2 (= rechts, B) gesetzt (erscheint nur bei UPD)  <b>1&amp;2:</b> Validity-Bit in beiden Kanälen gesetzt</p>
<b>Parity (nur bei UPD)</b>	<p>legt fest, ob das Protokoll mit Parityfehlern erzeugt wird .  <b>TRUE:</b> Alle Samples werden mit korrekter Parity erzeugt.  <b>WITH ERR:</b> Einstellung der Fehlerrate mit den Befehlen No. Trues, No. Falses und Offset.  <b>No. True:</b> gibt die Zahl der Samples mit korrektem Paritybit an.          Zahleingabe von 100... 1000000          Erläuterung zur Definition von Parityfehlern:          Es werden zuerst eine Anzahl (Offset) Samples mit korrektem Paritybit ausgegeben, danach beginnt eine zyklische Sequenz bestehend aus jeweils einer Anzahl (No. False) Samples mit falschem Paritybit gefolgt von einer Anzahl (No. True) Samples mit korrektem Paritybit. "True" und "False" wiederholen sich dann abwechselnd.          Die Zahlen beziehen sich auf die Summe aus L und R Samples.          Anwendungen:  <ul style="list-style-type: none"> <li>• Wenn die Summe aus No. True und No. False 384 ist, dann kann mit Offset jeweils die Position der Parityfehler innerhalb eines Blocks bestimmt werden.</li> <li>• Ein sehr großes Offset kann benutzt werden, um eine Testhardware sicher einschwingen zu lassen.</li> </ul> <b>No. False:</b> gibt die Zahl der Samples mit falschem Paritybit an.          Zahleingabe von 0 ..1000          Bei einer Einstellung von 0 werden nur richtige Paritybits ausgegeben.  <b>Offset:</b> gibt die Zahl der Samples mit korrektem Paritybit bis zum ersten Parityfehler an.          Zahleingabe von 0 ... 1000000          Erläuterung siehe bei No. True.</p>
<b>Block Err (nur bei UPD)</b>	<p>gibt das Intervall (in Blöcken) an, mit dem Fehler in der Blockbeginn-Präambel-Folge erzeugt werden.          Zahleingabe von 0 ... 100000          0: kein Fehler, alle Blockbeginn -Präambel sind richtig.          1 ... 100000: Es wird nach der angegebenen Zahl von Samples jeweils eine Blockbeginn -Präambel durch eine normale "Links"-Präambel ersetzt. (fehlender Blockbeginn)</p>

<b>Sequ. Err (nur bei UPD)</b>	<p>gibt das Intervall (in Blöcken) an, mit dem Fehler in der Präambelreihenfolge erzeugt werden. Zahleneingabe von 0 ... 10000</p> <p>0: kein Fehler, alle Präambels richtig.</p> <p>1 ... 10000: Es wird mit dem angegebenen Abstand von von Blöcken jeweils ein Block mit Sequenz-Fehler erzeugt. Dabei wird die 48. linke Präambel innerhalb dieses Blockes durch eine rechte ersetzt. Statt der Sequenz " L R L R L " wird die Sequenz " L R R R L " erzeugt. Dieser umgedrehte Sample hat aber sonst die Statusinformation des entsprechenden linken Kanals, nur die Präambel ist invertiert.</p>
<b>Ch Stat L</b>	<p>legt die Art der Erzeugung von Channel Statusdaten LINKS fest.</p> <p><b>ZERO:</b> Alle Channel Status Datenbits sind 0. (Der Betriebsmode (AES3, CRC, RAW) wird vom Befehl Ch Stat. R festgelegt).</p> <p><b>FILE + AES3:</b> UPL/UPD erzeugt selbst local timecode und CRC, die restlichen Channel Status Daten werden mit der folgenden Datei festgelegt.</p> <p><b>PANEL + AES3:</b> UPL/UPD erzeugt selbst local timecode und CRC, restliche Channel Daten werden mit dem durch die Datei "Panelfile" definiertem Panel eingestellt.</p> <p><b>FILE+CRC:</b></p> <p><b>PANEL + CRC:</b> wie FILE + AES3 bzw. PANEL + AES3, jedoch local timecode wird nicht vom UPL/UPD erzeugt, sondern als fester Wert eingestellt.</p> <p><b>FILE:</b></p> <p><b>PANEL:</b> wie FILE + AES3 bzw. PANEL + AES3, jedoch weder local timecode noch CRC werden vom UPL/UPD erzeugt. (Betriebsart RAW)</p> <p>Die hier mögliche Einstellung beeinflusst Ch Stat. R. Sollte dort etwas eingestellt sein, das nicht zugleich mit der hier getroffenen Wahl möglich ist, dann erscheint eine entsprechende Fehlermeldung und die Einstellung wird zurückgewiesen. Das PANEL kann nur einmal vergeben werden und die Betriebsart (AES3, CRC, RAW) muß in beiden Kanälen gleich sein.</p> <p><b>BINARY:</b> hier erscheint ein Window, in dem die einzelnen Channel-Status-Bits binär eingegeben werden können. Die Taste SELECT wechselt das jeweilige Bit von 0 auf 1 oder umgekehrt. BINARY kann nur einmal bei CH Stat L oder Ch Stat R verwendet werden.</p>
<b>Filename</b>	<p>Angabe einer Datei, die Channel Status Daten für LINKS enthält.</p> <p>voreingestellter Dateityp: * pgc</p> <p>Die Daten werden jeweils mit einer Zeile definiert. Durch Schlüsselworte in der Datei wird angegeben, für welche Bits und für welche Seite (links, rechts oder beide) die Daten bestimmt sind. Bei diesem Befehl werden jedoch nur diejenigen Definitionen berücksichtigt, die Angaben für die linke oder für beide Seiten erhalten. Damit kann die gleiche Datei für beide Seiten verwendet werden.</p> <p>Syntax:</p> <p>Seite:</p> <p>Mit AES _ CHAN _ STAT oder AES _ CHAN _ STAT _ BOTH wird der Beginn von Daten für beide Seiten angezeigt. Entsprechend zeigen AES _ CHAN _ STAT _ RIGHT und AES _ CHAN _ STAT _ LEFT den Beginn der Daten für eine Seite an.</p> <p>Diese Schlüsselworte dürfen in beliebiger Reihenfolge, auch wiederholt, auftreten.</p> <p>Daten:</p> <p>In der Datei werden Werte mit folgender Zeile eingegeben: Schlüsselwort BIT, gefolgt von einem Zielbereich und dem jeweiligen Wert.</p> <p><b>Beispiel:</b> BIT: 12-15, (BITS 12-15 werden mit NOOOI~ b elegt)</p> <p>Falls die Generatorstatusdaten aus dem Analysator genommen werden sollen, kann dies mit dem Schlüsselwort "TRACK" statt eines Wertes angegeben werden.</p> <p><b>Beispiel:</b> BIT: 12-15, TRACK</p> <p>Die Uhrzeit des UPL/UPD kann mit dem Schlüsselwort "TIME" statt eines Wertes angegeben werden. Dabei wird die Uhrzeit mit der momentan gewählten Abtastrate (siehe Kapitel 2.5.3) multipliziert, um die Zahl der Samples seit Mitternacht zu erhalten.</p> <p><b>Beispiel:</b> BIT: 112- 143, TIME <b>Beispieldatei</b> R&amp;S_AES3.PGC</p>
<b>Ch Stat. R</b>	<p>legt die Art der Erzeugung von Channel Status Daten RECHTS fest. Die hier mögliche Einstellung ist von Ch Stat. L abhängig. Das PANEL kann nur einmal vergeben werden und die Betriebsart (AES3, CRC, RAW) muß in beiden Kanälen gleich sein.</p> <p>Erläuterungen zu den einzelnen Befehlen siehe bei Ch. Stat L.</p> <p><b>ZERO:</b> Alle Channel Status Datenbits sind 0. (Betriebsart (AES,CRC,RAW) wird durch Ch. Stat L festgelegt. Sollte Links auch auf ZERO eingestellt sein, so ist die Betriebsart RAW).</p> <p><b>EQUAL L:</b> Beide Seiten sind gleich, alle für links getroffenen Definitionen werden nach rechts kopiert. Die Betriebsart wird durch Ch. Stat L festgelegt.</p>

**FILE+AES3:** Diese Auswahl erscheint nur, falls bei Ch.Stat L ZERO, FILE+AES3 oder PANEL I AES3 gewählt wurde.

**PANEL + AES3:** Diese Auswahl erscheint nur, falls bei Ch.Stat L ZERO oder FILE + AES3 gewählt wurde.

**FILE + CRC:** Diese Auswahl erscheint nur, falls bei Ch.Stat L ZERO, FILE + CRC oder PANEL + CRC gewählt wurde.

**PANEL + CRC:** Diese Auswahl erscheint nur, falls bei Ch.Stat L ZERO oder FILE + CRC gewählt wurde.

**FILE:** Diese Auswahl erscheint nur, falls bei Ch.Stat L ZERO, PANEL oder FILE gewählt wurde.

**PANEL:** Diese Auswahl erscheint nur, falls bei Ch.Stat L ZERO oder FILE gewählt wurde.

**BINARY:** erscheint nur, falls bei Ch Stat L nicht bereits BINARY gewählt wurde.

**Filename** Angabe einer Datei, die Channel Status Daten für RECHTS enthält.  
voreingestellter Dateityp: \*.pgc  
Die Daten werden jeweils mit einer Zeile definiert. Durch Schlüsselworte in der Datei wird angegeben, für welche Bits und für welche Seite (links, rechts oder beide) die Daten bestimmt sind. (siehe hierzu die Angaben für die linke Seite)  
Bei diesem Befehl werden jedoch nur diejenigen Definitionen berücksichtigt, die Angaben für die rechte oder für beide Seiten enthalten. Damit kann die gleiche Datei für beide Seiten verwendet werden.  
Format: siehe bei Ch. Stat. L

**CRC Error (nur bei UPD)** gibt das Intervall (in Blöcken) an, mit dem Fehler im CRC der Statusdaten erzeugt werden.  
Zahleingabe von 0... 100000  
0: kein Fehler, alle CRCs richtig.  
1 ... 100000: Es wird nach der angegebenen Zahl von Blöcken jeweils ein CRC-Fehler in beiden Kanälen gleichzeitig erzeugt. (CRC-Fehler: der richtige CRC wird durch das inverse Bitmuster ersetzt).

Dieser Befehl erscheint nur, falls die Betriebsart der Channel Statusdaten CRC oder AES3 ist.

**User Mode** legt die Art der Erzeugung von User-Daten fest.  
**ZERO:** alle Userbits sind zu 0 initialisiert.  
**FILE DEF:** Userbits werden gemäß den Definitionen in der nachfolgenden Datei ausgegeben.  
**Hinweis:**  
Ein Wechsel der Userbits hält sowohl den Generator als auch den Analysator kurzfristig an.

**Filename** Angabe einer Datei, die User Daten enthält.  
voreingestellter Dateityp: \*.pgu  
  
In der Datei sind sowohl Userdaten für die linke als auch rechte Seite enthalten. Durch das Schlüsselwort AES \_ USER \_ DATA \_ LEFT wird der Beginn von Daten für die linke Seite angezeigt, entsprechend wirkt AES \_ USER \_ DATA \_ RIGHT.  
  
Werte müssen jeweils ohne weitere Bezeichnung in den folgenden Zeilen als Hexadezimalzahl angegeben werden, wobei jede Zeile 32 Bit enthält.  
  
Die eingelesenen Userbits werden vom UPL/UPD zyklisch wiederholt, wobei der Zyklus für beide Seiten gleich lang ist. Wenn auf einer Seite weniger Bits als auf der anderen Seite eingelesen werden, so wird mit 0 aufgefüllt. 6 (oder ein Vielfaches davon) ergeben mit dem Blockbeginn korrigierte Userdaten.  
Die maximal zulässige Zahl von Userbits sind 4096 Worte = 16384 Bytes = 131072 Bits je Seite.

Beispiel: AES\_USER\_DATA\_RIGHT  
0 x 55504420 # = ' U PD '

**Panelfile** Angabe einer Datei, welche die Definitionen des frei definierbaren Panels enthält. Dieser Menüpunkt erscheint nur, wenn bei Ch.Stat oder CH.Stat R ein Panel angegeben wurde.

voreingestellter Dateityp: \*.PPC

Mit dieser Datei kann ein auf die jeweilige Anforderungen zugeschnittenes Eingabefeld definiert werden. Dazu werden in der Datei jeweils die Befehlsbezeichnungen, die erlaubte Wertauswahl und die jeweilige Zielposition der Einstellung angegeben. Es sind dabei 4Arten von Befehlen zulässig:

- **Auswahlbefehle** Durch den Eintrag des Schlüsselwortes SELECTION, oder EXTSELECT gefolgt von Parametern, wird ein Auswahlbefehl definiert.

**Beispiele:**

  - SELECTION " Use", BIT:0, 0 = "CONS", 1 = "PROF"
  - (Bit 0 der Channel Status Daten kann mit der Menüzeile ' Use' zwischen ANLR TRACK, CONS und PROF umgeschaltet werden.)
  - SELECTION " Usermod", BIT:12 -15,0= "notind", 3= "USER"
  - (Die 4 Bits 12-15 der Channel Status Daten können mit der Menüzeile " Usermod" zwischen ANLR TRACK, not ind und USER umgeschaltet werden.)

**Hinweise:**

  - Es sind max. 12 normale Auswahlbefehle und 3 erweiterte zulässig.
  - Die Auswahl ANLR TRACK wird automatisch vom UPL/UPD eingefügt. Ihre Wahl bedeutet das Kopieren der entsprechenden Bits des Analysators.
  - Das BIT-Intervall darf nicht größer als 32 sein.
  - Es sind max. 8 Auswahlmöglichkeiten je Befehl ( = SELE - Bei sich überlappenden Bitbereichen überschreibt die jeweils letzte Einstellung die bisher definierten Bits.
  - Bei EXTSELECT sind bis zu 24 Auswahlmöglichkeiten zulässig.
- **Textbefehle** Durch den Eintrag des Schlüsselwortes TEXT, gefolgt von Parametern, wird ein Textbefehl definiert.

**Beispiel:**

  - TEXT " Origin", BIT:48 -79  
(Die 32 Bits 48-79 der Channel Status Daten werden mit den hier einzugebenden Textzeichen gefüllt.)

**Hinweise:**

  - Es sind max. 4 Auswahlbefehle zulässig.
  - Das BIT-Intervall darf nicht größer als 32 sein, d.h. es sind nur maximal 4 Textzeichen (ASCII) je Befehl zulässig.
  - Bei einem leerem Eingabetext wird vom UPL/UPD der entsprechende Inhalt des Analysators kopiert.
- **Wertbefehle (normal)** Durch den Eintrag des Schlüsselwortes VALUE, gefolgt von Parametern, wird ein Wertbefehl definiert. In diesem können Status bits als Zahlen angegeben werden. Diese Zahlen können noch mit einem in der Datei angegebenen Multiplikator multipliziert werden .

**Beispiel:**

  - VALUE " Abs.Hour", BIT:144 -175, MULT:SET\_RATE M U LT: 3600
  - (In den 32 Bits 144-175 der Channel Status Daten wird der hier eingegebene Wert eingetragen, wobei vor der Eintragung die Zahl hinter MULT ( 3600 \* und SET \_ RATE = eingestellte Abtastrate) mit dem Wert multipliziert wird )

**Hinweise:**

  - Es sind max.12 Wertbefehle zulässig.
  - Das BIT-Intervall darf nicht größer als 32 sein, die Eingabe ist auf 31 Bit beschränkt.
  - Bei einer Paneleingabe -1 wird der entsprechende Bit-Bereich des Analysators kopiert.
  - Die 2 Schlüsselworte MULT sind optional; Als Multiplikationszahl ist auch das Schlüsselwort SET\_RATE zulässig.
- **Wertbefehle (additiv)** Durch den Eintrag des Schlüsselwortes ADDVALUE, gefolgt von Parametern, wird ein Wertbefehl definiert, wobei dieser Wert auf bereits vorhandene Werte addiert wird.

**Beispiel:**

  - ADDVALUE " Abs.Min", BIT:144 -175, MULT:2880000
  - (Auf die 32 Bits 144-175 der Channel Status Daten wird der hier eingegebene Wert addiert, wobei vor der Addition die Zahl hinter MULT (288000 = 60 \* 48000) mit dem Wert multipliziert wird.)

**Hinweise:** - Siehe beim normalen Wertbefehl.

**Beispiele:** R&S \_AES3.PPC für Panel nach AES3-Format  
R&S \_CONS.PPC für Panel nach Consumer-Format

Die Befehle werden vom UPL/UPD in der Reihenfolge

  1. SELECTION/EXTSELECT
  2. VALUE / ADD VALUE
  - 3.TEXT

im Panel dargestellt. In der Datei ist jedoch eine beliebige Reihenfolge erlaubt.

## 4. Protokoll-Analyse

### Bildschirmaufbau:

Der Bildschirm (bestehend aus 16 Zeilen zu je 50 Zeichen) wird in 2 Bereiche aufgeteilt:

#### OBEN: Feste Protokollanzeigen:

- Validity:** zeigt an, wie das Validitybit im jeweiligen Kanal steht.
- Parityerrors:** zeigt beim UPD die Summe der bisherigen Parityfehler an; Diese Zahl wird bei Neuwahl des Analysators oder beim Druck auf die Starttaste wieder auf Null zurückgesetzt. Im UPL wird angezeigt, ob Parity-Fehler aufgetreten sind oder nicht.
- Change:** zeigt an, ob sich in den Channel Status Daten irgendwas verändert hat.  
**NO:** keine Änderung  
**LTC:** nur Änderungen in den Feldern 'local time code' (Bits 112 bis 143) und CRC (Bits 184 -191).  
**YES:** Änderungen in einer sonstigen Bitposition.
- Other:** zeigt an, ob der jeweils andere Kanal gleich oder unterschiedlich ist
- Errors:** zeigt die aufgetretenen Fehler an:  
**beim UPL:**  
**LOCK:** PLL nicht eingerastet  
**CONFIDEN:** nicht decodierbar (eye open less than half a bit)  
**CODING:** Codier-Fehler erkannt  
**NONE:** kein Fehler  
**beim UPD:**  
**UNEXP BB:** unerwartete Präambel für Blockbeginn (zu früh)  
**SQ BLOCK:** fehlende (Lücke) Präambel für Blockbeginn  
**NOT LOCK:** fehlende Präambel für Blockbeginn PREAMBLE: ungültige Präambel  
**SQ L/R:** Fehler in der Kanalfolge (L/R)  
**RATE ERR:** gemessene Taktrate und eingestellte Rate weichen um mehr als 200 ppm voneinander ab.  
**NONE:** kein Fehler

**UNTEN:** Diese 13 Zeilen können mit einer Protokollsteuerdatei dem jeweils verwendeten Protokoll angepaßt werden .

Die Fehlerzähler Parityerror, CRC \_ links und CRC \_ rechts können beim UPD durch Drücken auf die Starttaste oder Neuwahl des Analysators zurückgesetzt werden.

In der Bilddarstellung werden sich ändernde Statusbits bei der Ausgabe rot dargestellt. Seit der letzten Ausgabe unveränderter Bits werden in grün dargestellt. Kommentartexte ("PRINT"-Befehl) werden in gelb dargestellt.

#### Beispiel: Protokoll Analyse: Channel Status Links

vom UPD  
eingetragener Text  
ab hier; user defined

Validity:	R:0	L:1	Parityerrors: -
Change:	YES	Other: EQUAL	Errors: NONE
Byte:	=====	AES3	=====
0:	Format: prof	Mode: audio	
	Emph: J.17	Source: locked	
	Rate: 48kHz		
1:	Chanmod: stereo	Usermod: AES18	
2:	Auxmod: 20 no	Length: 23	R:0
3-5:	Vector: 12	Grade: 2	R:0
6-13:	Origin: UPD_	Destin: R&S_	
14-21:	Local: 01234567	Time: 12:45:00	
22:	Relia: 0-5:1 6-13:0	14-17:0	18-21:0
23:	CRC L:	CRC R:	
Measured sample rate: 48000.2			

dieses Beispiel wird mit der Protokolldatei R&S \_AES3.PAC erzeugt.



**Source** wählt die darzustellenden Daten aus.  
**CHAN STAT L** Channel Status Daten links werden dargestellt.  
**CHAN STAT R** Channel Status Daten rechts werden dargestellt.  
**USER L** User Daten links werden dargestellt.  
**USER R** User Daten rechts werden dargestellt.

bei CHAN STAT L oder R erscheint folgende weitere Zeile:

**Proto File** Proto File wählt die Interpretationsdatei für Channel Status Daten aus.  
Auswahl aus: Dateinamen, voreingestellter Dateityp: \*.pac

Beschreibung:

Die Protokollinformation wird in zwei Teilen dargestellt; die beiden oberen Zeilen werden fest formatiert dargestellt und enthalten Informationen über Validity, Parity, sonstige Fehler sowie über Unterschiede der beiden Kanäle. Die restlichen Zeilen werden an Hand der Angaben in der Datei formatiert, wobei den einzelnen Bits jeweils ein beliebiger Ausgabertext zugeordnet werden kann. Diese Datei kann vom Anwender an beliebige Protokolle angepaßt werden.

Beispiel: siehe R&S\_AES3.PAC

Dateiformat: gleich wie bei USER DAT, siehe unten

bei USER DAT erscheinen folgende weitere Zeilen:

**FORMAT** Format wählt den Interpretationsmode.  
Auswahl aus: BINARY, HEX, ASCII, AES18  
**BINARY** Userdaten werden als 01010-Folge dargestellt. Darstellung erfolgt Block aligned.  
**HEX** Userdaten werden als hexadezimaler Zahlen-Folge dargestellt. Darstellung erfolgt Block aligned.  
**ASCII** Userdaten werden als Text dargestellt. Darstellung erfolgt Block aligned.  
**FILE** Interpretation, Block aligned

bei FILE erscheint noch folgende Zeile:

**Proto File** Proto File wählt die Interpretationsdatei für User-Daten aus.  
Auswahl aus: Dateinamen, voreingestellter Dateityp: \* PAU

Dateiformat:

Die Protokolldatei hat folgenden Aufbau:

Je Zeile eine Operation, gefolgt von den Parametern, jeweils durch Kommata getrennt. Die Parameter erlauben die Darstellungen von beliebigen Bits (jeweils 1 bis zu 32) aus den gewählten Daten, wobei die Ausgabeposition frei gewählt werden kann. Jede Zeile darf nicht mehr 255 Zeichen enthalten.

Operationen:

**PRINT:** Textausgabe (unabhängig von Daten)

Beispiel:

PRINT 26, 5, "Usermod:"  
(in Spalte 26, Zeile 5 wird der Text Usermod: eingetragen)

**VALUE:** Wertausgabe, entweder als HEX-Zahl (default) oder als Text, sofern eine Zuweisung vorhanden ist.  
Beispiel:

VALUE 17, 3, BIT:2-4, 0 = "not ind", 4 = "no empf", 6= "50/15 "  
(in Spalte 17, Zeile 3 wird der Inhalt der 3 Bits (2,3,4) als Hex (oder Text, falls der Wert 0, 4 oder 6 ist) dargestellt)

VALUE 17,11, CRC\_L\_ERR  
(in Spalte 17, Zeile 11 wird der Inhalt des CRC Links Fehlerzählers dargestellt)

Hinweis zur Ausgabebreite:  
Falls Textzuweisungen enthalten sind, bestimmt der längste Text die Ausgabebreite; bei fehlenden Textzuweisungen ergibt sich die Ausgabebreite aus der Anzahl der darzustellenden Bits.

Als Datenquelle sind folgende Angaben zulässig:

- 'BIT' gefolgt von ':' und der Angabe der Bits:
  - eine Zahl von 0 bis 191: Einzelbit
  - ein Intervall (z.B. 4-9): zusammengefaßte Folge von Bits, max. 32 Bits zulässig.
- 'CRC\_L\_ERR': interner Fehlerzähler der CRC\_Fehler links.
- 'CRC\_R\_ERR': interner Fehlerzähler der CRC\_Fehler rechts.
- 'MEASURED\_RATE': gemessene Abtastrate (dargestellt als 5.1-stellige Floatzahl (z. B. '48001.2')

**BINARY:** wie VALUE, jedoch Default-Ausgabe als Bitmuster.

**TEXT:** Textausgabe (in ASCII) in der jeweiligen durch die Anzahl der ausgewählten Bits Zahl von Buchstaben; Je TEXT-Operation können nur jeweils 32 Bit dargestellt werden, für längere Textausgabe sind mehrere Textzeilen hintereinander zu verwenden.

Beispiel:

TEXT 17,8, BIT:48-79  
(in Spalte 17, Zeile 8 wird der Inhalt der 32 Bits als Text dargestellt)  
Nicht druckbare Zeichen werden als '?' bzw. '.' falls '0' dargestellt.

**TIME:** Zeitausgabe (als 12:45:56) der ausgewählten Zahl dividiert durch die angegebene Rate.  
Zahl/Rate wird als Sekunden seit Mitternacht interpretiert.

Beispiele:

TIME35,9, BIT:144-175, RATE:48000.0 TIME35,9, BIT: 144 -175, RATE:SET\_RATE  
(in Spalte 35, Zeile 9 wird der Inhalt der 32 Bits als Zeit (z.B.12:34:45) dargestellt)

**RATE:** Sollte der Abtastrate entsprechen, kann als Floatzahl angegeben werden. Alternativ sind hier auch folgende Angaben möglich:

MEASURED\_RATE: gemessene Taktrate  
SET\_RATE: im Panel eingestellte Taktrate

## 5. Auszug aus der Norm AES3

### **AES Recommended practice for digital audio engineering-- Serial transmission format for two-channel linearly represented digital audio data**

#### **1 Scope**

This document specifies a recommended interface for the serial digital transmission of two channels of periodically sampled and linearly represented digital audio data from one transmitter to one receiver.

It is expected that the format will be used to convey audio data that have been sampled at any of the sampling **frequencies recognized** by the AES5 *Recommended Practice for Professional Digital Audio Applications Employing Pulse-Code Modulation--Preferred Sampling Frequencies*.

The format is intended for use with shielded twisted pair cable of conventional design over distances of up to 100 m without transmission equalization or any special equalization at the receiver. Longer cable lengths may be used, but with a rapidly increasing requirement for care in cable selection and possible receiver equalization or the use of active repeaters, or both.

The document does not cover connection to any common carrier equipment, nor does it specifically address any questions about the synchronizing of large systems, although by its nature the format permits easy synchronization of receiving devices to the transmitting devices.

Specific synchronization issues are covered in AES 11.

In this interface specification, mention is made of an interface for consumer use. The two interfaces are not identical.

An engineering guideline document to accompany this interface specification is in preparation.

#### **2 Interface format**

##### **2.1 Terminology**

**2.1.1 sampling frequency:** Frequency of the samples representing an audio signal. When more than one audio signal is transmitted through the same interface, the sampling frequencies are identical.

**2.1.2 audio sample word:** Amplitude of a digital audio sample. Representation is linear in 2's complement binary form. Positive numbers correspond to positive analog voltages at the input of the analog-to-digital converter (ADC). The number of bits per word can be specified from 16 to 24 in two coding ranges (less than or equal to 20 bits and less than or equal to 24 bits).

**2.1.3 auxiliary sample bits:** 4 least significant bits (LSBs) which can be assigned as auxiliary sample bits and used for auxiliary information when the number of audio sample bits is less than or equal to 20.

**2.1.4 validity bit:** Bit indicating whether the audio sample bits in the subframe (time slots 4 to 27 or 8 to 27, depending on the audio word length as described in 2.2.1 ) are suitable for conversion to an analog audio signal.

**2.1.5 channel status:** Bits carrying, in a fixed format derived from the block (see 2.1.11), information associated with each audio channel which is decodable by any interface user.

**2.1.6 user data:** Channel provided to carry any other information.

**2.1.7 parity bit:** Bit provided to permit the detection of an odd number of errors resulting from malfunctions in the interface.

**2.1.8 preambles:** Specific patterns used for synchronization. There are three different preambles (see 2.4).

**2.1.9 subframe:** Fixed structure used to carry the information described in 2.2.2).

**2.1.10 frame:** Sequence of two successive and associated subframes.

**2.1.11 block:** Group of 192 consecutive frames. The start of a block is designated by a special subframe preamble (see 2.4).

**2.1.12 channel coding:** Coding describing the method by which the binary digits are represented for transmission through the interface.

## 2.2 Structure of format

**2.2.1 Subframe format** Each subframe is divided into 32 time slots, numbered from 0 to 31 (see figure 1).

Time slots 0 to 3 (preambles) carry one of the three permitted preambles (see 2.2.2 and 2.4; also see figure 2) .

Time slots 4 to 27 (audio sample word) carry the audio sample word in linear 2's complement representation. The most significant bit (MSB) is carried by time slot 27.

When a 24-bit coding range is used, the LSB is in time slot 4 [see figure 1(a)].

When a 20-bit coding range is sufficient, time slots 8 to 27 carry the audio sample word with the LSB in time slot 8. Time slots 4 to 7 may be used for other applications. Under these circumstances, the bits in time slots 4 to 7 are designated auxiliary sample bits [see figure 1(b)].

If the source provides fewer bits than the interface allows (either 20 or 24), the unused LSBs are set to logic 0.

Time slot 28 (validity bit) carries the validity bit associated with the audio sample word (see 2.5).

Time slot 29 (user data bit) carries 1 bit of the user data channel associated with the audio channel transmitted in the same subframe (see Section 3).

Time slot 30 (channel status bit) carries 1 bit of the channel status information associated with the audio channel transmitted in the same subframe (see Section 4) .

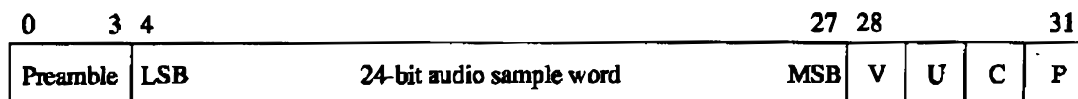
Time slot 31 (parity bit) carries a parity bit such that time slots 4 to 31 inclusive will carry an even number of ones and an even number of zeros (even parity).

NOTE The preambles have even parity as an explicit property.

### 2.2.2 Frame format

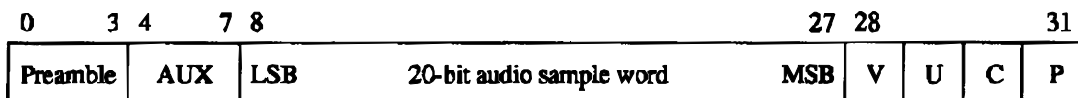
A frame is uniquely composed of two subframes (see figure 2). The rate of transmission of frames corresponds exactly to the source sampling frequency.

The first subframe normally starts with preamble "X." However, the preamble changes to preamble "Z" once every 192 frames. This defines the block structure used to organize the channel status information. The second subframe always starts with preamble "Y."



(a)

**V**    **Validity bit**  
**U**    **User data bit**  
**C**    **Channel status bit**  
**P**    **Parity bit**  
**AUX** **Auxiliary sample bits**



(b)

Figure 1. Subframe format.

The modes of transmission are signaled by setting bits 0 to 3 of byte1 of channel status.

**Two-channel mode:** In two-channel mode, the samples from both channels are transmitted in consecutive subframes. Channel 1 is in subframe 1, and channel 2 is in subframe 2.

**Stereophonic mode:** In stereophonic mode, the interface is used to transmit stereophonic audio in which the two channels are presumed to have been simultaneously sampled. The left, or "A," channel is in subframe 1, and the right, or "B," channel is in subframe 2.

Each bit to be transmitted is represented by a symbol comprising two consecutive binary states. The first state of a symbol is always different from the second state of the previous symbol. The second state of the symbol is identical to the first if the bit to be transmitted is logic 0. However, it is different if the bit is logic 1 (see figure 3).

**Single-channel mode (monophonic):** In monophonic mode, the transmitted bit rate remains at the normal two-channel rate and the audio sample word is placed in subframe 1. Time slots 4 to 31 of subframe 2 either carry the bits identical to subframe 1 or are set to logic 0. A receiver normally defaults to channel 1 unless manual override is provided.

**Primary / secondary mode:** In some applications requiring two channels where one of the channels is the main or primary channel while the other is a secondary channel, the primary channel is in subframe 1, and the secondary channel is in subframe 2.

To minimize the direct-current (dc) component on the transmission line, to facilitate clock recovery from the data stream, and to make the interface insensitive to the polarity of connections, time slots 4 to 31 are encoded in biphase-mark.

## 2.4 Preambles

Preambles are specific patterns providing synchronization and identification of the subframes and blocks.

To achieve synchronization within one sampling period and to make this process completely reliable, these patterns violate the biphase-mark code rules, thereby avoiding the possibility of data imitating the preambles.

A set of three preambles is used. These preambles are transmitted in the time allocated to four time slots at the start of each subframe (time slots 0 to 3), and are represented by eight successive states. The first state of the preamble is always different from the second state of the previous symbol (representing the parity bit).

Depending on this state the preambles are:

Preceding state	0	1
-----------------	---	---

	Channel coding		
"X"	11100010	00011101	Subframe 1
"Y"	11100100	00011011	Subframe 2
"Z"	11101000	00010111	Subframe 1 and block start

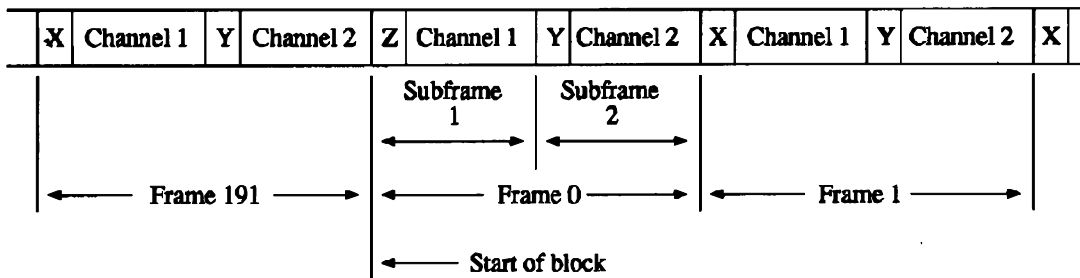


Figure 2. Frame format

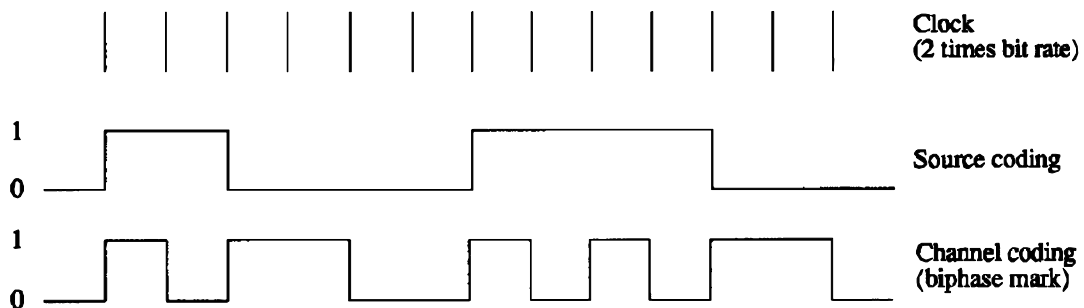


Figure 3. Channel coding

Like biphase code, these preambles are dc free and provide clock recovery. They differ in at least two states from any valid biphase sequence.

Figure 4 represents preamble "X."

NOTE: Owing to the even-parity bit in time slot 31, all preambles will start with a transition in the same direction (see 2.2.1). Thus only one of these sets of preambles will, in practice, be transmitted through the interface. However, it is necessary for either set to be decodable because a polarity reversal might occur in the connection.

### 2.5 Validity bit

The validity bit is logic 0 if the audio sample word is suitable for conversion to an analog audio signal, and it is logic 1 if it is not.

There is no default state for the validity bit.

### 3 User data format

User data bits may be used in any way desired by the user.

Possible formats for the user data channel are indicated by the channel status byte 1, bits 4-7.

The default value of the user data bit is logic 0.

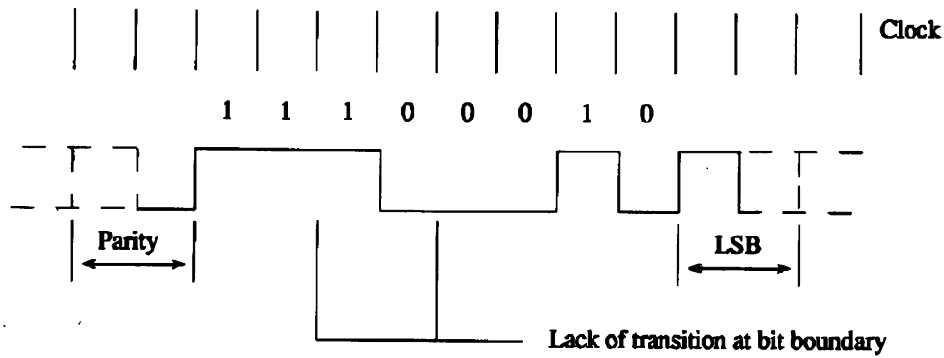


Figure 4 Preamble X (11100010).

#### 4 Channel status format

The channel status for each audio signal carries information associated with that audio signal, and thus it is possible for different channel status data to be carried in the two subframes of the digital audio signal. Examples of information to be carried in the channel status are: length of audio sample words, number of audio channels, sampling frequency, sample address code, alphanumeric source and destination codes, and emphasis.

Channel status information is organized in 192-bit blocks, subdivided into 24 bytes (see figure 5). The first bit of each block is carried in the frame with preamble "Z. "

Byte	Bit 0	1	2	3	4	5	6	7
0	Use of channel status channel	Audio / nonaudio use	Audio signal emphasis			Locking of source sample frequency	Sampling frequency	
1	Channel mode				User bit management			
2	Use of auxiliary sample bits			Source word length and source encoding history		Reserved		
3	Future multichannel function description							
4	Digital audio reference signal		Reserved					
5	Reserved							
6	Alphanumeric channel origin data							
7								
8								
9								
10	Alphanumeric channel destination data							
11								
12								
13								
14	Local sample address code (32-bit binary)							
15								
16								
17								
18	Time-of-day sample address code (32-bit binary)							
19								
20								
21								
22	Reliability flags							
23	Cyclic redundancy check character							

Figure 5. Channel status data format.

The specific organization follows, wherein the suffix 0 designates the first byte or bit.

## Byte 0

bit 0	0	Consumer use of channel status block (see note).
	1	Professional use of channel status block.
bit 1	0	Normal audio mode.
	1	Nonaudio mode.
bits 2-4 bit state	Encoded audio signal emphasis.	
	234	
	000	Emphasis not indicated. Receiver defaults to no emphasis with manual override enabled.
	100	No emphasis. Receiver manual override is disabled
	110	50/15- $\mu$ s emphasis. Receiver manual override is disabled
111	International Telegraph and Telephone Consultative Committee (CCITT) J. 17 emphasis (with 6.5 dB insertion loss at 800 Hz). Receiver manual override is disabled.	
All other states of bits 2-4 are reserved and are not to be used until further defined .		
bit 5	1	Source sampling frequency unlocked.
	0	Default and source sampling frequency locked.
bits 6-7 bit state	Encoded sampling frequency.	
	67	
	00	Sampling frequency not indicated. Receiver default to 48 kHz and manual override or auto set is enabled
	01	48-kHz sampling frequency. Manual override or auto set is disabled.
	10	44. 1-kHz sampling frequency. Manual override or auto set is disabled.
11	32-kHz sampling frequency. Manual override or auto set is disabled.	

NOTE: The significance of byte 0, bit 0 is such that a transmission from an interface conforming to IEC 958 "consumer use" can be identified, and a receiver conforming only to IEC 958 "consumer use" will correctly identify a transmission from a "professional-use" interface as defined in this standard. Connection of a "professional-use" transmitter with a "consumer-use" receiver or vice versa might result in unpredictable operation. Thus the following byte definitions only apply when bit 0 = logic 1 (professional use of the channel status block) .

## Byte 1

bits 0-3 bit state	Encoded channel mode.	
	0123	
	0000	Mode not indicated. Receiver default to two channel mode. Manual override is enabled.
	0001	Two-channel mode. Manual override is disabled .
	0010	Single-channel mode (monophonic). Manual override is disabled.
	0011	Primary/secondary mode (subframe I is primary). Manual override is disabled .
	0100	Stereophonic mode (channel 1 is left channel). Manual override is disabled .
	0101	Reserved for user-defined applications.
	0110	Reserved for user-defined applications.
	1111	Vector to byte 3 . Reserved for future applications.
All other states of bits 0- 3 are reserved and are not to be used until further defined .		
bits 4-7 bit state	Encoded user bits management.	
	4567	
	0000	Default, no user information is indicated.
	0001	192-bit block structure. Preamble "Z" indicates the start of block.
	0010	Reserved for the AES18 standard
0011	User defined	
All other states of bits 4 7 are reserved to be used until further defined		



## Byte 2

bits 0-2 bits state	Encoded use of auxiliary sample bits		
	012		
	000	Maximum audio sample word length is 20 bits (default). Use of auxiliary sample bits is not defined	
	001	Maximum audio sample word length is 24 bits. Auxiliary sample bits are used for main audio sample data.	
	010	Maximum audio sample word length is 20 bits. Auxiliary sample bits in this channel are used to carry a single coordination signal (see note 1)	
	011	Reserved for user-defined applications .	
	All other states of bits 0-2 are reserved to be used until further defined		
	NOTE 1-The signal coding used for the coordination channel is described in Annex A		
bits 3-5 bit state	Encoded audio sample word length of transmitted signal (see notes 2, 3, and 4).		
	345	Audio sample word length if maximum length is 24 bits as indicated by bits 0-2 above.	Audio sample word length if maximum length is 20 bits as indicated by bits 0-2 above.
	000	Word length not indicated (default)	Word length not indicated (default)
	001	23 bits	19 bits
	010	22 bits	18 bits
	011	21 bits	17 bits
	100	20 bits	16 bits
	101	24 bits	20 bits
	All other states of bits 3-5 are reserved to be used until further defined		
bits 6-7	Reserved and are set to logic 0 until further defined		
	NOTE 2 The default state of bits 3-5 indicates that the number of active bits within the 20- or 24-bit coding range is not specified by the transmitter. The receiver should default to the maximum number of bits specified by the coding range and enable manual override or auto set.		
	NOTE 3 The nondefault state of bits 3-5 indicates the number of bits within the 20- or 24-bit coding range which might be active. This is also an indirect expression of the number of LSBs that are certain to be inactive, which is equal to 20 or 24 minus the number corresponding to the bit state. The receiver should disable manual override and auto set for these bit states.		
	NOTE 4 Irrespective of the audio sample word length as indicated by any of the states of bits 3-5, the MSB is in time slot 27 of the transmitted as specified in 2.2.1.		

## Byte 3

bits 0-7 bit state	Vectored target byte from byte 1 Reserved for future use as multichannel function description. These bits are set to logic 0 at present
--------------------------	--

## Byte 4

bits 0-1 bits state	Digital audio reference signal (per AES11)	
	01	
	00	Not a reference signal (default)
	01	Grade 1 reference signal
	10	Grade 2 reference signal
	11	Reserved and not used until further defined
bits 2-7	Reserved and are set to logic 0 until further defined	

## Byte 5

bits 0-7	Reserved and are set to logic 0 until further defined
----------	---

## Bytes 6-9

bits 0-7 (each byte)	Alphanumeric channel origin data. First character in message is byte 6. 7-bit International Organization for Standardization (ISO) 646, American Standard Code for Information Interchange (ASCII), data with no parity bit. LSBs are transmitted first with logic 0 in bit 7. Nonprinted control characters (codes 01 to 1F hex and 7F hex) are not permitted. Default value is logic 0 (code 00 hex, ASCII null).
-------------------------	--

## Bytes 10-13

bits 0-7 (each byte)	Alphanumeric channel destination data. First character in message is byte 6. 7-bit International Organization for Standardization (ISO) 646, American Standard Code for Information Interchange (ASCII), data with no parity bit. LSBs are transmitted first with logic 0 in bit 7. Nonprinted control characters (codes 01 to 1F hex and 7F hex) are not permitted. Default value is logic 0 (code 00 hex, ASCII null).
-------------------------	---

## Bytes 14-17

bits 0-7 (each byte)	Local sample address code (32-bit binary with LSBs first). Value is of first sample of current block LSBs are transmitted first. Default value is logic 0.  NOTE-This has the same function as a recording index counter.
-------------------------	--

## Bytes 18-21

bits 0-7 (each byte)	Time-of-day sample address code (32-bit binary with LSBs first). Value is of first sample of current block. LSBs are transmitted first. Default value is logic 0.  NOTE This is the time of day laid down during the source encoding of the signal and remains unchanged during subsequent operations. A value of all zeros for the binary sample address code is, for transcoding to real time, or to time codes in particular, to be taken as midnight (i.e., 00 h, 00 min, 00 s, 00 frame). Transcoding of the binary number to any conventional time code requires accurate sample frequency information to provide a sample accurate time.
-------------------------	--

## Byte 22

bits 0-3	Flag used to identify whether the information carried by the channel status data is reliable. According to the following table, if reliable, the appropriate bits are set to logic 0 (default); if unreliable, the bits are set to logic 1 Reserved and are set to logic 0 until further defined
bit 4	Bytes 0 to 5
bit 5	Bytes 6 to 13
bit 6	Bytes 14 to 17
bit 7	Bytes 18 to 21

## Byte 23

bits 0-7	Channel status data cyclic redundancy check character (CRCC) Generating polynomial is $G(x) = x^8 + x^4 + x^3 + x^2 + 1$ The CRCC conveys information to test valid reception of the entire channel status data block (bytes 0 to 22 inclusive). For serial implementations the initial condition of all ones should be used in generating the check bits with the LSB transmitted first. Default value is logic 0 for "minimum" implementation of channel status only (see 5.2.1) NOTE Annex B includes a diagram of the shift register circuit used to generate the code, two examples of channel status data, and the corresponding CRCC
----------	---

## 5. Interface format implementation

### 5.1 General

To promote compatible operation between items of equipment built to this specification it is necessary to establish which information bits and operational bits need to be encoded and sent by a transmitter and decoded by an interface receiver.

Documentation shall be provided describing the channel status features supported by the interface transmitters and receivers.

### 5.2 Transmitter

Transmitters shall follow all the formatting and channel coding rules established in earlier sections of this specification including all notes therein. Along with the audio sample word, all transmitters shall correctly encode and transmit the validity bit, user bit, parity bit, and the three preambles. The channel status shall be encoded to one of the implementations given in 5.2.1, 5.2.2, and 5.2.3.

The following three implementations are defined: "minimum," "standard," and "enhanced." These terms are used to communicate in a simple manner the level of implementation of the interface transmitter involving the many features of channel status. Irrespective of the level of implementation, all reserved states of bits defined in Section 4 shall remain unchanged.

### **5.2.1 "Minimum" implementation of channel status**

The "minimum" implementation represents the lowest level of implementation of the interface that meets the requirements of this specification document. In the "minimum" implementation, transmitters shall encode and transmit channel status byte 0 bit 0 with a state of logic 1 signifying "professional use of channel status block." All other channel status bits of byte 0 to byte 23 inclusive shall be transmitted with the default state of all logic 0-5. In this circumstance, the receiver will adopt the default conditions specified in bytes 0 to 2.

If additional bytes of channel status (which do not fully comply with the "standard" implementation, see 5.2.2) are implemented as required by an application, the interface transmitter shall be classified as a "minimum" implementation of channel status.

It should be noted that the "minimum" implementation imposes severe operational restrictions on some receiving devices which may be connected to it. For example, receivers implementing byte 23 will normally show a cyclic redundancy check error when the default value of logic 0 is received as the CRCC. Also, reception of the default value for byte 0 bits 6-7 might cause improper operation in receiving devices not supporting manual override or auto set capabilities.

### **5.2.2 "Standard" implementation of channel status**

The "standard" implementation provides a fundamental level of implementation which should prove sufficient for general applications in professional audio or broadcasting. In addition to conforming to the requirements described in 5.2.1 for the "minimum" implementation, a "standard" implementation interface transmitter shall correctly encode and transmit all channel status bits in byte 0, byte 1, byte 2, and byte 23 (CRCC) in the manner specified in this document.

### **5.2.3 "Enhanced" implementation of channel status**

In addition to conforming to the requirements described in 5.2.2 for the "standard" implementation, the "enhanced" implementation shall provide further capabilities .

## **5.3 Receivers**

Implementation in receivers is highly dependent on the application. Proper documentation shall be provided on the level of implementation of the interface receiver for decoding the transmitted information (validity, user, channel status, parity) and on whatever subsequent action is taken by the equipment of which it is a part.

## **6 Electrical requirements**

### **6.1 General characteristics**

The electrical parameters of the interface are based on those defined in CCITT V. 11 which allow transmission of balanced-voltage digital signals up to a few hundred meters distance.

In order to improve the balance of the transmitter or the receiver, or both, beyond that recommended by the CCITT, a circuit conforming to the general configuration shown in figure 6 may be used.

In this circuit, the series capacitors  $C_g$  and  $C_g$  isolate the transformers and prevent damage from connection to a source containing a dc voltage. In addition to achieving higher rejection of common-mode signals, the transformers reduce grounding and electromagnetic interference (EMI) problems. Although equalization may be used at the receiver, no equalization before transmission shall be permitted.

The interconnecting cable shall be balanced and screened (shielded) with a nominal characteristic impedance of  $110 \Omega$  at frequencies from 0.1 to 6.0 MHz .

## 6.2 Line driver characteristics

### 6.2.1 Output impedance

The line driver shall have a balanced output with an internal impedance of  $110\ \Omega \pm 20\%$ , at frequencies from 0.1 to 6.0 MHz, when measured at the output terminals.

### 6.2.2 Signal amplitude

The signal amplitude shall be between 2 and 7 V peak to peak, when measured across a  $110\text{-}\Omega$  resistor connected to the output terminals, without any interconnecting cable present.

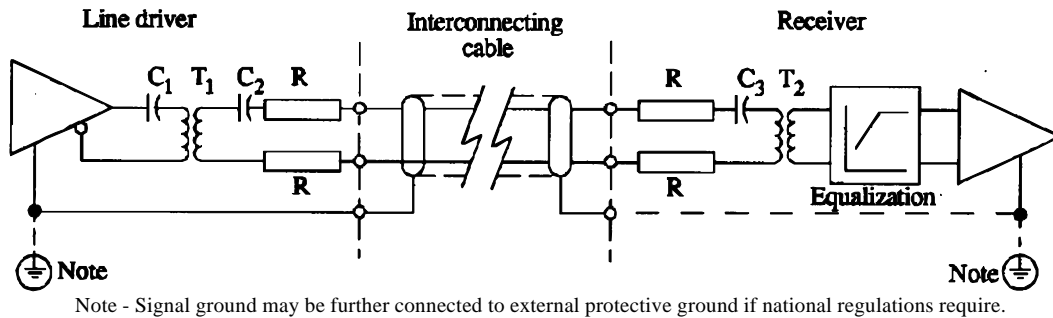


Figure 6 General circuit configuration.

### 6.2.3 Balance

Any common-mode component at the output terminals shall be more than 30 dB below the signal at frequencies from dc to 6 MHz.

### 6.2.4 Rise and fall times

The rise and fall times, determined between the 10% and 90% amplitude points, shall be between 5 ns and 30 ns when measured across a  $110\text{-}\Omega$  resistor connected to the output terminals, without any interconnecting cable present.

NOTE: Operation toward the lower limit of 5 ns may improve the received signal eye pattern, but may increase EMI at the transmitter. Equipment must meet local regulations regarding EMI.

Data transitions shall occur within  $\pm 20$  ns of an ideal jitter-free clock measured at the half-voltage points.

NOTE: This specification applies only to the signal after channel coding. Tighter specifications apply to the audio sample clock.

### 6.3.1 Terminating impedance

The receiver shall present an essentially resistive impedance of  $110\ \Omega \pm 20\%$  to the interconnecting cable over the frequency band from 0.1 to 6.0 MHz when measured across the input terminals. The application of more than one receiver to any one line might create transmission errors due to the resulting impedance mismatch.

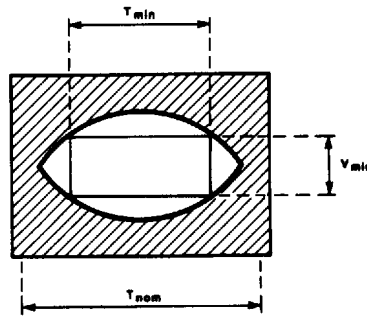
### 6.3.2 Maximum input signals

The receiver shall correctly interpret the data when connected directly to a line driver working between the extreme voltage limits specified in 6.2.2.

NOTE: The AES3-1985 specification for line driver signal amplitude was 10 V peak to peak maximum.

### 6.3.3 Minimum input signals

The receiver shall correctly sense the data when a random input signal produces the eye diagram characterized by a  $V_{\min}$  of 200 mV and  $T_{\min}$  of 50% of  $T_{\text{nom}}$  (see figure 7).



$T_{min}: 0.5 \times T_{nom}$   
 $V_{min}: 200 \text{ mV}$   
 $T_{nom}$  One-half the biphas symbol period

Figure 7, Eye diagram

### 6.3.4 Receiver equalization

Optional equalization can be applied in the receiver to enable an interconnecting cable longer than 100 m to be used. A suggested frequency equalization characteristic is shown in figure 8. The receiver shall meet the requirements specified in 6.3.2 and 6.3.3.

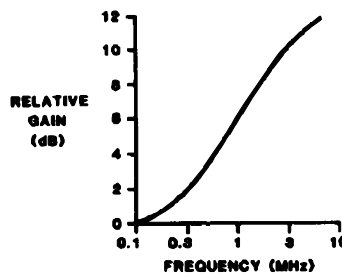


Figure 8. Suggested equalization characteristic for receiver

### 6.3.5 Common-mode rejection

There shall be no data errors introduced by the presence of a common-mode signal of up to 7 V peak at frequencies from dc to 20 kHz.

## 6.4 Connectors

The standard connector for both outputs and inputs shall be the circular latching three-pin connector described in IEC 268- 12. (This type of connector is normally called XLR.)

An output connector fixed on an item of equipment shall use male pins with a female shell. The corresponding cable connector shall thus have female pins with a male shell.

An input connector fixed on an item of equipment shall use female pins with a male shell. The corresponding cable connector shall thus have male pins with a female shell. The pin usage shall be:

- Pin 1 Cable shield or signal earth;
- Pin 2 Signal;
- Pin 3 Signal.

(Note that the relative polarity of pins 2 and 3 is not important in the digital case.)

Equipment manufacturers should clearly label digital audio inputs and outputs as such, including the terms "digital audio input" or "digital audio output" as appropriate .

In such cases where panel space is limited and the function of the connector might be confused with an analog signal connector, the abbreviation DI or DO should be used to designate digital audio inputs and outputs, respectively.

## 7 Normative References

[The following Standards contain provisions which, through reference in this text, constitute provisions of this Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and the most recent editions of the Standards listed below should be obtained.]

AES5-1984, AES Recommended Practice for Professional Digital Audio Applications Employing Pulse Code Modulation Preferred Sampling Frequencies, Audio Engineering Society, New York, NY, USA (1984).

AES10-1991, AES Recommended Practice for Digital Audio Engineering--Serial Multichannel Audio Digital Interface (MADI), Audio Engineering Society, New York, NY, USA (1991).

AES 11 - 1991, AES Recommended Practice for Digital Audio Engineering--Synchronization of Digital Audio Equipment in Studio Operations, Audio Engineering Society, New York, NY, USA (1991).

AES18-1992, AES Recommended practice for digital audio engineering--Format for the user data channel of the AES digital audio interface, Audio Engineering Society, New York, NY, USA (1992).

CCIR Recommendation 647, A digital audio interface for broadcasting studios, Green Book, vol. 10 part 1, International Radio Consultative Committee (CCIR), Dubrovnik, Yugoslavia (1986).

CCITT Recommendation J.17, Pre-emphasis used on sound program circuits, International Telegraph and Telephone Consultative Committee (1972).

CCITT Recommendation V.11, Electrical characteristics for balanced double-current interchange circuits for general use with integrated circuit equipment in the field of data communications, International Telegraph and Telephone Consultative Committee (1976, 1980).

IEC 268-11, Sound system equipment, part 11: Application of connectors for the interconnection of sound system components, International Electrotechnical Commission, Geneva, Switzerland (1987).

IEC 268-12, Sound system equipment, part 12: Application of connectors for broadcast and similar use, International Electrotechnical Commission, Geneva, Switzerland (1987).

IEC 841, Audio recording--PCM encoder / decoder system, International Electrotechnical Commission, Geneva, Switzerland (1988).

IEC 908, Compact disc digital audio system, International Electrotechnical Commission, Geneva, Switzerland (1987).

IEC 958, Digital audio interface, International Electrotechnical Commission, Geneva, Switzerland (1989).

ISO 646: 1983, Information processing--ISO 7-bit coded character set for information interchange, International Organization for Standardization, Geneva, Switzerland (1983).

Aus:

J. Audio Eng. Soc, Vol. 40, No. 3, 1992 March