

EBU Technical Statement D84-1999

Use of 50 Mbit/s MPEG compression in television programme production

<i>EBU Committee</i>	<i>First Issued</i>	<i>Revised</i>	<i>Re-issued</i>
TC	1999		

Keywords: Video compression, Video recording

In 1996, EBU Statement D80 [1] identified the need for a single, open video compression system algorithm for storage and data transfer in television programme production. The EBU considers that a compression algorithm operating in an I-frame only format at about 50 Mbit/s is likely to offer sufficient picture quality and headroom for post-processing in all but the most demanding applications.

Based on the recent findings of the EBU/SMPTE Task Force [2] and experimental evidence, the EBU has accepted that two compression families, DV and MPEG, have been developed that are suitable for this purpose. Both use intraframe coding at a bit rate of about 50 Mbit/s.

The EBU wishes to call the attention of manufacturers to the fact that some of the operating requirements of a television production environment may entail constraints in the detailed specifications of MPEG bit streams, particularly when recording MPEG bit streams on tape.

These requirements include the following:

- Bit-by-bit transparent recording, editing and playback of bit streams,
- Insert editing of an arbitrary number of frames,
- Ability to perform pre-read edits,
- Slow-motion and still frame playback from suitable machines,
- recording and replay in faster than real time should not be precluded.

The EBU asks manufacturers of television production equipment using MPEG video compression operating close to an upper bound of 50 Mbit/s to ensure that all equipment, particularly VTRs, can inter-operate with other equipment operating at the same nominal bit rate.

Bibliography

- [1] EBU Statement D80-1996: **Compression in television programme production**
 - [2] EBU Tech Review Special Supplement 1998: **Final Report: Analyses and Results, EBU/SMPTE Task force for Harmonised Standards for the Exchange of Programme Material as Bit-streams**
-