INTERNATIONAL STANDARD ISO/IEC 23003-3:2012
TECHNICAL CORRIGENDUM 2

Published 2013-09-01

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — MPEG audio technologies —

Part 3:
Unified speech and audio coding

TECHNICAL CORRIGENDUM 2

Technologies de l'information — Technologies audio MPEG —

Partie 3: Discours unifié et codage audio

RECTIFICATIF TECHNIQUE 2

Technical Corrigendum 2 to ISO/IEC 23003-3:2012 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

At the end of 4.5.3 add:

"

Furthermore, restrictions on the sampling rate apply for the USAC Baseline profile. The sampling rate signaled as part of the UsacConfig() shall be one out of those listed in Table 3. These sampling rates are chosen such that they can conveniently be resampled to 44100 Hz and 48000 Hz, respectively.

ICS 35.040 **Ref. No. ISO/IEC 23003-3:2012/Cor.2:2013(E)**

Published in Switzerland

Table 3 — Allowed sampling rates for the Baseline USAC profile

SamplingRates [Hz] and usacSamplingFrequencyIndex			
88200	0x01	96000	0x00
70560	n/a	76800	n/a
58800	n/a	64000	0x02
44100	0x04	48000	0x03
35280	n/a	38400	0x12
29400	n/a	32000	0x05
22050	0x07	24000	0x06
17640	n/a	19200	0x17
14700	n/a	16000	0x08
11760	n/a	12800	0x1a
11025	0x0a	12000	0x09
8820	n/a	9600	0x1b
7350	0x0c	8000	0x0b

"

In 5.3.2 replace:

“

Table 23 – Syntax of UsacCoreCoderData()

Syntax	No. of bits	Mnemonic

UsacCoreCoderData(nrChannels, indepFlag)
{
  for (ch=0; ch < nrChannels; ch++) {
  core_mode[ch]**;**
	1	**uimsbf**

}
 	
  if (nrChannels == 2) {
  StereoCoreToolInfo(core_mode);
  }
 	
  for (ch=0; ch<nrChannels; ch++) {
  if (core_mode[ch] == 1) {
  lpd_channel_stream(indepFlag);
  }
  else {
  if ((nrChannels == 1) || (core_mode[0] != core_mode[1])) {		

tns_data_present[ch];	**1**	**uimsbf**

}
  fd_channel_stream(common_window, common_tw,
  tns_data_present[ch], noiseFilling, indepFlag);
  }
  }
}		

“

With:

“

Table 23 – Syntax of UsacCoreCoderData()

Syntax	No. of bits	Mnemonic

UsacCoreCoderData(nrChannels, indepFlag)
{
  for (ch=0; ch < nrChannels; ch++) {
  core_mode[ch]**;**
	1	**uimsbf**

}
 	
  if (nrChannels == 2) {
  StereoCoreToolInfo(core_mode);
  }
 	
  for (ch=0; ch<nrChannels; ch++) {
  if (core_mode[ch] == 1) {
  lpd_channel_stream(indepFlag);
		NOTE 1

}
  else {
  if ((nrChannels == 1) || (core_mode[0] != core_mode[1])) {
  tns_data_present[ch];
	1	**uimsbf**

}
  fd_channel_stream(common_window, common_tw,
  tns_data_present[ch], noiseFilling, indepFlag);
		NOTE 2

} } }		
Note 1: Each channel shall have its own instance of lpd_channel_stream Note 2: Each channel shall have its own instance of fd_channel_stream		

“

In Table 23, replace

”
```
if (nrChannels == 2) {
  StereoCoreToolInfo(core_mode);
}
```
”

With

”
```
if (nrChannels == 2) {
  StereoCoreToolInfo(core_mode, stereoConfigIndex);
}
```
”

In Table 24, replace

”
```
if (ms_mask_present == 3) {
  cplx_pred_data();
}
```
”

With

″
```
if ((ms_mask_present == 3)  && (stereoConfigIndex == 0)){
  cplx_pred_data();
}
```
″

In Table 35, replace

″
```
for (k=0; k<no_qn; k++) {
   qn_data(nk_mode, no_qn)
}
```
″

With

″
```
qn_data(nk_mode, no_qn);
```
″

In Table 38, replace

″
```
arith_finish(x_ac_dec, i,N);
```
″

With

″
```
arith_finish(x_ac_dec, i,N,lg);
```
″

In 7.4.3, replace

″
```
arith_finish(x_ace_dec,offset,N)
{
  arith_rewind_bitstream(14);
  for (i=offset ;i<N/4;i++) {
    x_ac_dec[2*i] = 0;
    x_ac_dec[2*i+1] = 0;
    q[1][i] = 1;
  }
}
```
″

With

″
```
arith_finish(x_ace_dec,offset,N,lg)
{
  if(lg>0) arith_rewind_bitstream(14);
  for (i=offset ;i<N/4;i++) {
    x_ac_dec[2*i] = 0;
    x_ac_dec[2*i+1] = 0;
```



```
    q[1][i] = 1;
  }
}
"
```

In 7.7.2.3.4, replace

"
```
if (ms_mask_present == 3) {
```
"

With

"
```
if ((ms_mask_present == 3)  && (stereoConfigIndex == 0)){
```
"

In subclause 7.11.2.3.3, replace

"
$$\theta_1^{l,m} = OPD_{left}^{l,m}$$
$$\theta_2^{l,m} = OPD_{left}^{l,m} - IPD^{l,m}$$
"

With

"
$$\theta_1^{l,m} = OPD_{left}^{l,m}$$
$$\theta_2^{l,m} = OPD_{left}^{l,m} - IPD^{l,m}$$

It is noted that due to the wrapping property of phase values, the correction angles $\theta_1^{l,m}$ and $\theta_2^{l,m}$ are calculated using a modulo 2π operation.
"

In 7.11.2.7, replace

"
The RES input to the MPS decoder is fed by the 32 QMF band analysis of the RES output from the core decoder, with the upper 32 QMF bands set to zero (as described in ISO/IEC 23003-1:2007, 6.3.3 for downsampled MPS decoder operation).
"

With

"
The RES input to the MPS decoder is fed by the 32 (16, 24, depending on sbrRatioIndex) QMF band analysis of the RES output from the core decoder, with the upper 32 (resp. 48, 40) QMF bands set to zero (as described in ISO/IEC 23003-1:2007, 6.3.3 for downsampled MPS decoder operation).
"

In 7.11.2.7, replace

"
The MPS decoder is fed by the 32 QMF band analysis of the output of the core decoder, with the upper 32 QMF bands set to zero (as described in ISO/IEC 23003-1:2007, 6.3.3 for downsampled MPS decoder operation).
"

With

"
The MPS decoder is fed by the 32 (16, 24, depending on sbrRatioIndex) QMF band analysis of the output of the core decoder, with the upper 32 (resp. 48, 40) QMF bands set to zero (as described in ISO/IEC 23003-1:2007, 6.3.3 for downsampled MPS decoder operation).
"