Technical Specification Group Services and System Aspects ***TSGS#16(02)0228***
Meeting #16, Marco Island, Florida, USA, 10-13 June 2002

Source: **TSG-SA WG4**

Title: CRs to TS 28.062 on Corrections to Clauses 4 to 10 and to Annexes C and H, Additional TFO_Message Elements for Immediate Codec Type Optimisation, TFO Version Handling (Release 5)

Document for: **Approval**

Agenda Item: **7.4.3**

The following CRs, agreed at the TSG-SA WG4 meeting #21, are presented to TSG SA #16 for approval.

Spec	CR	Rev	Phase	Subject	Cat	Vers	WG	Meeting	S4 doc
28.062	017	1	REL-5	Editorial corrections and additions	F	5.0.0	S4	TSG-SA WG4#21	S4-020311
28.862	019	2	REL-5	Additional TFO_Message Elements for Immediate Codec Type Optimisation	F	5.0.0	S4	TSG-SA WG4#21	S4-020355
28.862	020	2	REL-5	Corrections to TS 28.062, sections 4 to 8	F	5.0.0	S4	TSG-SA WG4#21	S4-020352
28.862	021	3	REL-5	Corrections to TS 28.062, Annex C	F	5.0.0	S4	TSG-SA WG4#21	S4-020364
28.862	022	2	REL-5	TFO Version Handling	F	5.0.0	S4	TSG-SA WG4#21	S4-020357
28.862	023	2	REL-5	Configuration Exchange in Annex C	F	5.0.0	S4	TSG-SA WG4#21	S4-020358
28.862	024	2	REL-5	Corrections to Annex H	F	5.0.0	S4	TSG-SA WG4#21	S4-020359
28.862	026	1	REL-5	Corrections to sections 9 and 10	F	5.0.0	S4	TSG-SA WG4#21	S4-020353
28.862	027	1	REL-5	Immediate Codec Type Optimization	F	5.0.0	S4	TSG-SA WG4#21	S4-020356

3GPP TSG-S4 #21
Rennes, France, 13-17 May 2002

S4-020311

CR-Form-v6.1

CHANGE REQUEST

⌘	TS 28.062	CR	017	⌘ rev	1	⌘	Current version:	5.0.0	⌘
	Spec Title:	Inband Tandem Free Operation (TFO) of speech codecs							⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network ☐ Core Network **X**

Title:	⌘	Editorial Corrections and Additions to 28.062 V5.0.0
Source:	⌘	TSG SA WG4
Work item code:	⌘	AMRWB
Date:	⌘	2002-06-11
Category:	⌘	**F**
Release:	⌘	**REL-5**

Use one of the following categories:
- ***F*** *(correction)*
- ***A*** *(corresponds to a correction in an earlier release)*
- ***B*** *(addition of feature),*
- ***C*** *(functional modification of feature)*
- ***D*** *(editorial modification)*

Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
REL-4	*(Release 4)*
REL-5	*(Release 5)*

Reason for change:	⌘	Inconsistency, incompleteness
Summary of change:	⌘	Correct used abbreviation In/Output example of TFO decision rule C-Code for AMR-WB
Consequences if not approved:	⌘	Specification incorrect or difficult to use

Clauses affected:	⌘	11.2.3, Annex E		
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ O&M Specifications	⌘	
Other comments:	⌘			

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

.

11.2.3 Mandatory Minimum set of Modes (MaMiMo) for AMR-WB TFO

If for one or both sides of a possible TFO connection an AMR-WB codec type offers a Supported Codec Set which is not a superset or is not equal to the Mandatory Minimum set of Modes, then no AMR-WB TFO shall be done involving this AMR-WB codec type.

The MaMiMo for AMR-WB TFO comprises the AMR-WB modes 8,85 and 12,65 (kbit/s).

Annex E (normative): TFO Decision Algorithm C-Code

E.1 Brief Description of the Program 'tfo_decision'

The program 'tfo_decision' implements the TFO decision algorithm described in clauses 11 and 12. With the help of this program, the TFO decision algorithm can be run for different codec configurations in order to check and illustrate the TFO decision algorithm.

To perform the whole TFO decision algorithm it is needed to run the C-Code for all combinations of local and distant supported codec types. The output of the programm tells if TFO would be possible for a single combination and in which way. The ranking of the TFO candidates is not done by this C-Code. For that, it has to be checked the preference list in §11.6.2 (and for AMR-WB in some cases the OACS evaluation in §12.3.2.3).

The necessary files for compiling the program 'tfo_decision' are: tfo.cpp, tfo_decision.cpp, extensionsForAMRWB.cpp, tfo_decision.h, oacs.cpp, oacs.h, extensionsForAMRWB.h.

The files oacs.h, oacs.cpp, tfo_decision.h, tfo_decision.cpp, extensionsForAMRWB.cpp and extensionsForAMRWB.h serve as reference implementation of the TFO decision algorithm.

The C-Code is available in a separate file AMR_TFO_C-Code(version_number of 28.062).zip.

In case of inconsistencies between the TFO decision C-Code and clauses 11 and 12 the C-Code shall take precedence.

E.1.1 Input

The program tfo_decision reads from stdin. Each line is separated by spaces into 10 fields that contain the input data for a TFO decision. For example:

```
XXXXXXXX -X--XX-X 4 FR_AMR y --XXXXXX ---X-X-X 3 HR_AMR y
```

1. field:	LSCS	XXXXXXXX	all modes supported
2. field:	LACS	-X--XX-X	modes 10,2, 6,7, 5,9, 4,75
3. field:	LMACS	4	local MACS 4
4. field:	LUC	FR_AMR	local used codec type FR_AMR
5. field:	LOM	y	('y' or 'n') local optimization mode yes
6. field:	DSCS	--XXXXXX	modes 7,95, 7,4, 6,7, 5,9, 5,15, 4,75
7. field:	DACS	---X-X-X	modes 7,4, 6,7, 5,9, 4,75
8. field:	DMACS	3	distant MACS 3
9. field:	DUC	HR_AMR	distant used codec type HR_AMR

10. field: DOM y ('y' or 'n') distant optimization mode yes

The fields LSCS, LACS, DSCS, DACS must consist of 8 characters ‘x’ or 'X' or '-' in case of AMR-NB codec types or of 9 characters for AMR-WB types. They are indicating the 8 AMR or the 9 AMR-WB modes. An ‘x’ or ‘X’ stands for ‘mode is present’.

The LMACS and DMACS field must be numbers. LUC and DUC may be FR_AMR, HR_AMR, UMTS_AMR, UMTS_AMR_2, GSM_EFR, GSM_FR, GSM_HR, OHR_AMR, FR_AMR-WB, UMTS_AMR-WB, OHR_AMR-WB, OFR_AMR-WB. The LOM and DOM fields must be 'y' or 'n'.

E.1.2 Output

The program tfo_decision prints directly to stdout. The output is self-explaining, e.g.:

E.1.2.1 Output for AMR-NB

```
    FR_AMR                      HR_AMR
   MACS = 4                    MACS = 3
   OM = yes                    OM = yes


      SCS    ACS    IACS   OACS   CSCS   ACS    SCS
12,2  X      -      -      -      -      -      -
10,2  X      X      -      -      -      -      -
7,95  X      -      -      X      X      -      X
7,40  X      -      -      -      X      X      X
6,70  X      X      -      X      X      -      X
5,90  X      X      X      -      X      X      X
5,15  X      -      -      -      X      -      X
4,75  X      X      X      X      X      X      X


Change ACS to OACS and establish TFO.
```

OACS: In this example the IACS consists of the modes 5,9 and 4,75. The OACS consists of three modes (7,95, 6,7, 4,75). The TFO Decision Algorithm states that the ACSs on both sides have to be changed to the OACS in order to establish TFO. Immediate TFO is not possible in this example.

E.1.2.2 Output for AMR-WB

```
  UMTS_AMR_WB                    UMTS _AMR_WB
    MACS = 4                       MACS = 4
    OM = yes                       OM = yes


      SCS    ACS    IACS   OACS  DefACS  CSCS   ACS    SCS
23.85 X      X      -      X     X       X      -      X
23.05 -      -      -      -     -       -      -      -
19.85 -      -      -      -     -       -      -      -
18.25 -      -      -      -     -       -      -      -
15.85 X      X      -      X     X       X      X      X
14.25 -      -      -      -     -       -      -      -
12.65 X      X      -      X     X       X      X      X
 8.85 X      X      -      X     X       X      X      X
 6.60 X      -      -      -     -       X      X      X


Change ACS to OACS and establish TFO on OACS.
OACS evaluation step 1 (*1000): 1000
OACS evaluation step 2 (*1000): 878.571
Preference value: 3
```

In this example the IACS is empty, because immediate TFO is not possible (reason: the contiguity rule). The TFO Decision Algorithm states that the ACSs on both sides would have to be changed to the OACS in order to establish TFO. The OACS is identical to the Default ACS : 23.85, 15.85, 12.65 and 8.80.
The values of the OACS evaluation steps and the preference value are given to allow a comparison with possibly present alternative AMR-WB TFO candidate configurations.

Remark: The reference C-Code is capable of handling all 9 modes of AMR-WB, not only those 5 (6,60, 8,85, 12,65m 15,85, 23,85) for speech telephony service. Reason is, that the C-Code was designed before the reduction of AMR-WB modes.

3GPP TSG-SA4 #21
Rennes, France, 13-17 May 2002

S4-020355

CR-Form-v6.1

CHANGE REQUEST

⌘	**TS 28.062**	**CR**	**019**	⌘ **rev**	**2**	⌘	Current version:	**5.0.0**	⌘
	Spec Title:	Inband Tandem Free Operation (TFO) of speech codecs							⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network **X** Core Network **X**

Title:	⌘	**TFO_Message Elements for Immediate Codec Type Optimisation**
Source:	⌘	**TSG SA WG4**
Work item code:	⌘	AMRWB
Date:	⌘	2002-06-11
Category:	⌘	**F**
Release:	⌘	**REL-5**

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
REL-4 (Release 4)
REL-5 (Release 5)

Reason for change:	⌘	Optimised TFO establishment for alternative codecs, Version Handling. The procedures defined in TS 28.062-500 are not general enough, but work only for AMR-NB. The corrections here extend this to all codec types. This was in principle already agreed in the TFO subgroup in Lulea (SA4 #20).
Summary of change:	⌘	New elements in TFO Messages
Consequences if not approved:	⌘	The default Codec Type Optimisation has to be used causing increased delay and reduced speech quality during TFO establishment.

Clauses affected:	⌘	7		
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ O&M Specifications	⌘	
Other comments:	⌘	see accompanying document on " Immediate Codec Type Optimisation"		

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

7 TFO Messages

The TFO Messages, introduced in clause 6, follow the generic IS_Message principle defined in annex A.
The following definitions are provided for the ***Sender*** side:

TFO_REQ (): Identifies the source of the message as a TFO capable device, using a defined Codec_Type.
TFO_REQ contains the following parameters ():

- the System_Identification of the sender;
- the specific Local_Signature of the sender;
- the Local_Used_Codec_Type at sender side;
- possibly additional attributes for the Local_Used_Codec_Type.
- possibly additionally the TFO_Version
- possibly additionally alternative Codec_Types (short form of Codec_List)
- possibly additionally a future TFO_Extension.

TFO_ACK (): Is the response to a TFO_REQ Message.
TFO_ACK contains the corresponding parameters as TFO_REQ, except for the Local_Signature replaced by the Reflected_Signature, copied from the received TFO_REQ Message.

TFO_REQ_L (): Is sent in case of Codec Mismatch or for sporadic updates of information.
TFO_REQ_L contains the following parameters ():

- the System_Identification of the sender;
- the specific Local_Signature of the sender;
- the Local_Used_Codec_Type at sender side;
- the Local_Codec_List of alternative Codec_Types;
- possibly additional attributes for the used and the alternative Codec_Types.
- possibly additionally the TFO_Version
- possibly additionally a future TFO_Extension.

TFO_ACK_L (): Is the response to a TFO_REQ_L Message.
TFO_ACK_L contains the corresponding parameters as TFO_REQ_L, except for the Local_Signature replaced by the Reflected_Signature, copied from the received TFO_REQ_L Message.

TFO_TRANS (): Commands possible IPEs to let the TFO Frames pass transparently within the LSB (8 kbit/s) or the two LSBs (16 kbit/s) or the four LSBs (32kbit/s). TFO_TRANS contains the following parameter ():

- the Local_Channel_Type (8 kbit/s or 16 kbit/s or 32 kbit/s).

TFO_NORMAL: Commands possible IPEs to revert to normal operation.
TFO_NORMAL has no parameters.

TFO_DUP: Informs the distant partner that TFO Frames are received, while still transmitting PCM samples.
TFO_DUP has no parameters.

TFO_SYL: Informs the distant partner (if still possible) that TFO Frames are no longer received.
TFO_SYL has no parameters.

TFO_FILL: Message without specific meaning, used to pre-synchronise IPEs or to bridge over gaps in TFO protocols. TFO_FILL has no parameters.

7.1 Extendibility

A mechanism for future extensions is defined in a way that existing implementations in the field shall be able to ignore future, for them unknown Codec_Types and their potential attributes. The existing implementations shall be able to decode the remainder of the messages (which is known to them) uncompromised. This mechanism allows to extent:

- the number of Local_Used_Codec_Types from 15 (short form) up to 255 (long form) for one System_Identification;
- the Codec_List;
- the Codec_Attributes (if needed).

In case of the TFO_REQ or TFO_ACK messages the attributes of the Local_Used_Codec_Type shall be sent in the codec specific way, without a preceding Codec_Attribute_Head Extension_Block. Existing equipment, that do not know a future Codec_Type and therefore do not know if and how many attribute Extension_Blocks do follow, shall skip these Extension_Blocks, until they find a TFO Message Header again. Similarly, if future Extension_Blocks to a known Codec_Type are detected, existing equipment shall skip these Extension_Blocks, until they find a TFO Message Header again.

In case of the TFO_REQ_L or TFO_ACK_L Messages the simple Codec_List shall be sent immediately after the SIG_LUC and possible Codec_x Extension_Blocks. Then the attributes of all alternative Codec_Types shall follow. Each set of codec attributes shall be preceded by the Codec_Attribute_Head Extension_Block (with Codec_Type Identifier and Length Indicator) followed by the Codec specific attributes.

7.2 Regular and Embedded TFO Messages

A TFO Message is called "**regular**", if it is sent inserted into the PCM sample stream. A TFO Message is called "**embedded**", if it is embedded into a TFO Frame. The bit stealing scheme, as defined in Annex A, is identical for regular and embedded TFO Messages. The EMBED bit of the TFO Frames (see clause 5) indicates if the TFO Frame contains an embedded TFO Message. Due to the specific construction of the TFO Messages, they replace some of the synchronisation bits of the TFO Frames. Consequently, the TFO Frame synchronisation pattern will be affected by the presence of an embedded TFO Message, without compromising the synchronisation performances. Data and other control bits of the TFO Frames are not affected by embedded TFO Messages.

7.3 Cyclic Redundancy Check

The Extension_Blocks, defined in the following clauses, shall be protected by three CRC parity bits. These shall be generated as defined in the 3GPP TS 48.060 for the Enhanced Full Rate. For simplicity the present document is reprinted here:
"These parity bits are added to the bits of the subset, according to a degenerate (shortened) cyclic code using the generator polynomial:

$$g(D) = D^3 + D + 1$$

The encoding of the cyclic code is performed in a systematic form which means that, in GF(2), the polynomial:

$$d(m)D^n + d(m+1)D^{n-1} + ...+ d(m + n-3)D^3 + p(0)D^2 + p(1)D + p(2)$$

where p(0), p(1), p(2) are the parity bits, when divided by g(D), yields a remainder equal to:

$$1 + D + D^2$$

For every CRC, the transmission order is p(0) first followed by p(1) and p(2) successively."
In case of Extension_Blocks, p(0)..p(2) are mapped to bits 16..18.

7.4 TFO_REQ Messages

Symbolic Notation:
TFO_REQ (Sys_Id, LSig, Local_Used_Codec_Type[, Used_Codec_Attributes])
TFO_REQ_L (Sys_Id, LSig, Local_Used_Codec_Type, Codec_List [, Alternative_Codec_Attributes])

The TFO_REQ Messages conform to the IS_REQ Message format, defined in the Annex A, with IS_System_Identification, followed by the SIG_LUC Extension_Block, optionally the Codec_x Extension_Block, the Codec_List Extension_Block(s) and the Codec_Attribute Extension_Blocks.
The shortest TFO_REQ takes 140 ms for transmission, see Figure 7.4-1.
The shortest TFO_REQ_L takes 180 ms (Figure 7.4-2).

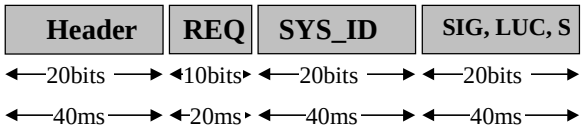


Figure 7.4-1: Construction of the shortest possible TFO_REQ Message

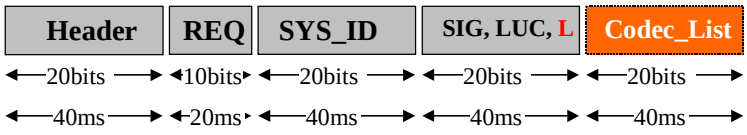


Figure 7.4-2: Construction of the shortest possible TFO_REQ_L Message

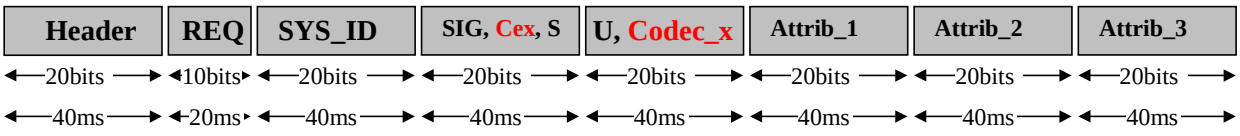


Figure 7.4-3: Example of a TFO_REQ Message with a Codec with an index higher than 15 and with three Attribute Extension_Blocks (300 ms length)

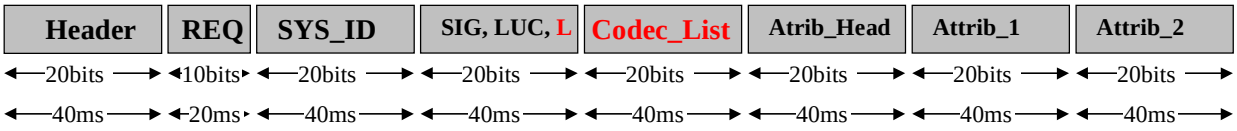


Figure 7.4-4: Example of a TFO_REQ_L Message with Codec_List and one alternative Codec with two Attribute Extension_Blocks (300 ms length)

A TFO_REQ (TFO_ACK) may have an additional TFO_Version Extension_Block that contains the TFO_Version.Subversion and a Selector. This Selector may indicate future extensions to TFO_REQ (TFO_ACK), which may require further additional Extension_Blocks following the TFO_Version, see figure 7.4-5.

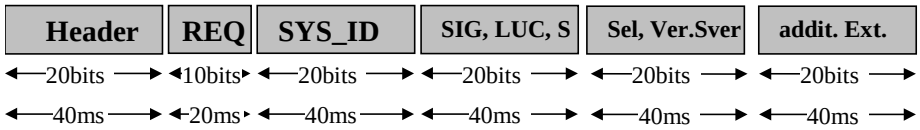


Figure 7.4-5: Construction of a TFO_REQ Message with Selector, TFO_Version.Subversion and one additional Extension_Block

7.4.1 Definition of the SIG_LUC Extension_Block

The SIG_LUC Extension_Block consists of 20 bits, as defined in Table 7.4.1-1. It shall always follow immediately after the SYS_ID Extension_Block. It differentiates a TFO_REQ from a TFO_REQ_L message and a TFO_ACK from a TFO_ACK_L message.
The Codec_x Extension_Block shall also be used in TFO_REQ or TFO_REQ_L messages if the Local_Used_Codec_Type has a CoID higher than 14.

Table 7.4.1-1: SIG_LUC Extension_Block

Bit	Description	Comment
Bit 1	"**0**"	normal IS-Message Sync Bit, constant.
Bit 2	**List_Ind**	Indicates, whether the Codec_List is included in the TFO Message or not **0: S: TFO_REQ** or **TFO_ACK**: Codec_List is not included (short) **1: L: TFO_REQ_L** or **TFO_ACK_L**: Codec_List is included (long)
Bit 3..10	**Sig**	An 8-bit random number to facilitate the detection of circuit loop back conditions and to identify the message source
Bit 11	"**0**"	normal IS-Message Sync Bit, constant
Bit 12.. 15:	**Codec_Type** **CoID_s** (short form)	Identifies the Local_Used_Codec_Type, which is currently used by the sender **0000…1110: reserved for 15 Codec_Types** **1111: Codec_x Extension_Block** follows immediately
Bit 16..18:	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX** EX == "0.0" EX == "1.1"	The normal 2 bits for IS_Message Extension. No other extension block follows An other extension block follows

7.4.2 Definition of the Codec_x Extension_Block

The Codec_x Extension_Block, if present, always follows the SIG_LUC Extension_Block. It consists of 20 bits, as defined in Table 7.4.2-1. It shall follow always immediately after the SIG_LUC Extension_Block, if the Codec_Type field is set to "1111".

Table 7.4.2-1: Codec_x Extension_Block

Bit	Description	Comment
Bit 1	"**0**"	normal IS-Message Sync Bit, constant.
Bit 2	**Codec_Sel**	Differentiates the Codec_x Extension_Block **0: U: Used_Codec_Type** is defined in Codec_Type_x field **1: Reserved**
Bit 3..10	**Codec_Type_x CoID** (long form)	Identifies the Local_Used_Codec_Type, which is currently used by the sender **0000.0000 … 1111.1111 reserved for 255 Codec_Types** 0000.1111 is undefined and shall not be used.
Bit 11	"**0**"	normal IS-Message Sync Bit, constant
Bit 12.. 15:	"1010"	Reserved for future use, set to "1010" to minimise audible effects
Bit 16..18:	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX**	The normal 2 bits for IS_Message Extension. 00: No other extension block follows 11: An other extension block follows

7.4.3 Definition of the Codec_List_Extension_Block

The Codec_List Extension_Block is used in a TFO_REQ_L, TFO_ACK_L messages to list the supported Codec_Types. It consists of 20 bits, as defined in Table 7.4.3-1. The Codec_List must at least contain the Local_Used_Codec_Type. If a system supports more than 12 Codec_Types, then other Codec_List Extension_Blocks (Table 7.4.3-2) may follow.

Table 7.4.3-1: Codec_List Extension Block

Bit	Description	Comment
Bit 1	"**0**"	Normal IS-Message Sync Bit, constant.
Bit 2..10	**Codec_List_1**	First part of Codec_List. For each Codec_Type one bit is reserved. If the bit is set to "0" then the specific Codec_Type is not supported; if the bit is set to "1" then the specific Codec_Type could be used.
Bit 11	"**0**"	Normal IS-Message Sync Bit, constant
Bit 12.. 14:	**Codec_List_2**	Second part of the Codec_List; All three bits are reserved for future Codec_Types (up to Codec_Type 12)
Bit 15	**Codec_List_x**	If set to "1" a further Codec_List Extension_Block follows; otherwise set to "0"
Bit 16..18:	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows

Table 7.4.3-2: Further Codec_List Extension Block(s)

Bit	Description	Comment
Bit 1	"**0**"	normal IS-Message Sync Bit, constant.
Bit 2..10	**Codec_List_1x**	First part of Codec_List. For each Codec_Type one bit is reserved. If the bit is set to "0" then the specific Codec_Type is not supported; if the bit is set to "1" then the specific Codec_Type could be used. Bit 2: Codec_Type 13 (+ x*12; x=1..2..3) Bit 4: Codec_Type 14 (+ x*12; x=1..2..3) and so on
Bit 11	"**0**"	normal IS-Message Sync Bit, constant
Bit 12.. 14:	**Codec_List_2x**	Second part of the Codec_List; All three bits are reserved for future Codec_Types (up to Codec_Type 24 (+x*12; x=1..2..3)
Bit 15	**Codec_List_xx**	If set to "1" a further Codec_List Extension_Block follows; otherwise set to "0"
Bit 16..18:	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows

7.4.4 Definition of the Codec_Attribute_Head Extension_Block

The Codec_Attribute_Head Extension_Block (Table 7.4.4-1) shall precede the Codec Attribute Extension_Blocks of a Codec_Type, if this Codec_Type needs additional attributes. This Codec_Attribute_Head identifies the Codec_Type and the number of additional Extension_Blocks to follow.

Table 7.4.4-1: Codec_Attribute_Head Extension_Block

Bit	Description	Comment
Bit 1	"**0**"	normal IS-Message Sync Bit, constant.
Bit 2	**PAR_Sel**	Differentiates this Extension_Block **0:** Parameters included in **PAR** field: Simple Codec_List_Extension **1:** Length Indicator (**LI**) included: Parameters follow in subsequent Extension_Blocks
Bit 3..10	**CoID**	This field identifies the Codec_Type for which the subsequent attributes are valid. The same coding as in the Codec_x Extension_Block is used (long form)
Bit 11	"**0**"	normal IS-Message Sync Bit, constant
Bit 12.. 15:	**LI / PAR**	If Par_Sel==1: LI: Length Indicator: 0000: reserved; 0001: one other Extension_Block follows, etc. If Par_Sel==0: PAR: Codec specific definition of these four bits
Bit 16..18:	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows

NOTE: This Extension_Block shall be used for the codecs introduced in the future that need attributes. It shall precede the Attribute Extension_Blocks. This allows earlier versions to skip the blocks they do not understand. It shall not be used for the GSM_FR, GSM_HR and GSM_EFR Codec_Types.

7.4.5 Definition of the TFO_Version Extension_Block

The TFO_Version Extension_Block (Table 7.4.5-1) contains the "TFO_Version" (4 bit), the "TFO_Subversion" (4 bit) and a "Selector" (5bit). The TFO_Version Extension Block (and the additional Extension_Blocks indicated by the Selector, if any, see below) shall always be the last of Extension_Blocks of a TFO_REQ or TFO_REQ_L (or TFO_ACK or TFO_ACK_L) message. This is necessary to provide compatibility with older versions, which must be able to skip these Extension_Blocks without being effected negatively.

The TFO_Version and TFO_Subversion are specified in Annex H. A TFO implementation of Release 5 or onwards shall send this TFO_Version. If it is omitted then a TFO_Version lower than 5 shall be assumed by the receiving side.

The Selector is used to indicate the type of extension and the number of additional extension blocks (if any). The Selector code "00000" indicates that no further extension is followig. The Selector code "10101" is not allowed to provide improved distinction against the TFO_Header.

7.4.5.1 Selector for Alternative Codecs

If the Selector is set to "00001" then this indicates that alternative codec types are supported, which are specified in additional Extension_Blocks following the TFO_Version Extension_Block. This Selector shall not be used in TFO_REQ_L or TFO_ACK_L messages, since equivalent information would then already be provided in the Codec_List

Extension_Block. It shall only be used in TFO_REQ or TFO_ACK messages to provide information on alternative codec types in an early stage of the TFO protocol, i.e., before TFO is established. For each alternative Codec_Type that is offered during TFO negotiation, one Codec_Attribute_Head Extension_Block shall be included. If the specified Codec_Type requires additional attributes then the required number of Codec_Attribute Extension_Blocks follow after the Codec_Attribute_Head Extension_Block. The list of alternative Codec_Types is terminated when the EX bits indicate no further Extension_Blocks (00) and the next TFO Message Header is following.

Table 7.4.5-1: TFO_Version Extension_Block

Bit	**Description**	**Comment**
Bit 1	"**0**"	normal IS-Message Sync Bit, constant.
Bit 2..6	**Selector**	Indicates if and which further extension_blocks are following. Coding for bits 2.3.4.5.6: 00000: nothing is following after this TFO_Version 00001: One (or more) alternative Codec Type(s) is (are) following, 10101: reserved (used by the IS_Header) all other codes: reserved for future use.
Bit 7..10	**Ver**	This field contains the TFO_Version number as specified in Annex H
Bit 11	"**0**"	normal IS-Message Sync Bit, constant
Bit 12.. 15:	**Sver**	This field contains the TFO_Subversion number as specified in Annex H
Bit 16..18:	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows

7.5 TFO_ACK Messages

Symbolic Notation:
TFO_ACK (Sys_Id, RSig, Local_Used_Codec_Type [, Used_Codec_Attributes])
TFO_ACK_L (Sys_Id, RSig, Local_Used_Codec_Type, Codec_List [, Alternative_Codec_Attributes]).
The TFO_ACK Messages conform to the IS_ACK Message, defined in the Annex A, with IS_System_Identification, followed by the SIG_LUC Extension_Block, and optionally the Codec_x Extension_Block, the Codec_List Extension_Block(s) and the Codec_Attribute Extension_Blocks.
TFO_ACK and TFO_REQ Messages differ only in the ACK / REQ Command block and the construction of the Signature: Local_Signature in case of TFO_REQ, Reflected_Signature in case of TFO_ACK. All extension blocks defined for the TFO_REQ are valid as well for TFO_ACK.
The shortest TFO_ACK takes 140 ms for transmission.
The shortest TFO_ACK_L takes 180 ms.

7.6 TFO_TRANS Messages

Symbolic Notation: TFO_TRANS (Channel_Type).
Two TFO_TRANS Messages are defined in conformity to the IS_TRANS Messages in Annex A.
For 8 kbit/s submultiplexing the "**TFO_TRANS (8k)**" is used and is identical to "IS_TRANS_1_u".
For 16 kbit/s submultiplexing the "**TFO_TRANS (16k)**" is used and is identical to "IS_TRANS_2_u".
For 32 kbit/s submultiplexing the ”**TFO_TRANS (32k)**” is used and is identical to ”IS_TRANS_4_u”.

TFO_TRANS() takes 100 ms for transmission.
In most cases the respective TFO_TRANS Message shall be sent twice: once as a regular TFO Message, exactly before any series of TFO Frames, and once embedded into the first TFO Frames, see clause 10.

7.7 TFO_NORMAL Message

Symbolic Notation: TFO_NORMAL.
The TFO_NORMAL Message is identical to the IS_NORMAL Message defined in the Annex A.
It shall be sent at least once whenever an established Tandem Free Operation needs to be terminated in a controlled way.
TFO_NORMAL takes 100 ms for transmission.

7.8 TFO_FILL Message

Symbolic Notation: TFO_FILL.
The TFO_FILL Message is identical to the IS_FILL Message, defined in the Annex A.
TFO_FILL may be used to pre-synchronise IPEs. Since IS_FILL is one of the shortest IS Messages, this is the fastest way to synchronise IPEs, without IPEs swallowing other protocol elements. By default three TFO_FILL messages shall be sent at the beginning; this number may be, however, configuration dependent.
One TFO_FILL takes 60 ms for transmission.

7.9 TFO_DUP Message

Symbolic Notation: TFO_DUP
The TFO_DUP Message is identical to the IS_DUP Message, defined in Annex A.
TFO_DUP informs the distant TFO Partner, that TFO Frames have been received unexpected, e.g. during Establishment. This enables a fast re-establishment of TFO after a ***local*** handover.
TFO_DUP takes 60 ms for transmission.

7.10 TFO_SYL Message

Symbolic Notation: TFO_SYL
The TFO_SYL Message is identical to the IS_SYL Message, defined in Annex A.
TFO_SYL informs the distant TFO Partner, that tandem free operation has existed, but suddenly no TFO Frames were received anymore. This enables a fast re-establishment of TFO after a ***distant*** handover.
TFO_SYL takes 60 ms for transmission.

7.11 Specification of the TFO Messages

7.11.1 Codec_Types

The Codec_Types are defined according to 3GPP TS 26.103, table 6.3-1.
The short form (CoID_s) exists for all Codec_Types with indices below 15 and consists of the last four bits (LSBs) of the long form (CoID).

7.11.2 Codec_List

The Codec_List is defined according to 3GPP TS 26.103. The mapping into the Codec_List Extension block shall be as follows: bit 1 of octet 1 shall be placed into Bit 2 of the Codec_List Extension block, and so on until bit 4 of octet 2 shall be placed into Bit 14.

If more than 12 Codec Types are contained in the Codec_List, then Bit 15 of the first Codec_List Extension block shall be set to "1" and an further Codec_List Extension block shall be added for the next 12 Codec Types.

7.11.3 Codec_Type Attributes

The Codec_Types GSM Full Rate, GSM Half Rate and GSM Enhanced Full Rate do not need additional attributes. They are fully defined by the System_Identification (see Annex A.5) and the Codec_Type.

7.11.3.1 AMR Codec_Type Attributes

The Adaptive Multi-Rate Codec_Types (FR_AMR, HR_AMR, UMTS_AMR, UMTS_AMR_2) and the Adaptive Multi-Rate Wideband Codec_Types (FR_AMR-WB and UMTS_AMR-WB) need several attributes within the TFO_REQ and TFO_ACK as well as in the TFO_REQ_L and TFO_ACK_L Messages. For Con_Req and Con_Ack frames see Annex C.
There are two major kinds of attributes: the ACS (Active Codec Set) and potentially the SCS (Supported Codec Set).
The ACS is related to the Local_Used_Codec_Type and is part of the Used_Codec_Attributes. One and exactly one ACS shall be sent in all cases where the Local_Used_Codec_Type is FR_AMR, HR_AMR, UMTS_AMR, or UMTS_AMR_2, FR_AMR-WB or UMTS_AMR-WB within one ACS_Extension_Block. This ACS_Extension_Block carries some more parameters, as defined in the next clause, the most important one is the "Full_Sub" flag, indicating whether or not the full set or a sub-set of the AMR (AMR-WB) is supported. In TFO_REQ and TFO_ACK Messages the ACS shall follow immediately after the SIG_LUC_Extension_Block. In TFO_REQ_L and TFO_ACK_L Messages an Attribute_Head_Extension_Block shall follow after the Local_Codec_List, indicating the Codec_Type it specifies, followed by the corresponding ACS_Extension_Block.
The SCS shall be sent in TFO_REQ or TFO_ACK only if the ACS_Extension_Block indicates that the sending side does not support the full set of AMR codec modes, but a subset (Full_Sub flag). In this case the SCS_Extension_Block shall follow immediately after the ACS_Extension_Block.

NOTE 1: Hence, the TFO_Protocol can decide immediately after the reception of TFO_REQ or TFO_ACK whether TFO is possible or not, and can report the distant TFO parameters to the Control Entity in the Network.

One and only one ACS_Extension_Block is included in TFO_REQ_L and TFO_ACK_L, if the Local_Used_Codec_Type is FR_AMR, HR_AMR, UMTS_AMR or UMTS_AMR_2, FR_AMR-WB or UMTS_AMR-WB. In addition, one SCS_Extension_Block is needed for each AMR Codec_Type flagged in the Local_Codec_List. In that case an Attribute_Head_Extension_Block shall follow after the Local_Codec_List, indicating the Codec_Type it specifies, followed by the corresponding SCS_Extension_Block. If multiple AMR_Codec_Types are flagged, then multiple Attribute_Heads and SCS_Extension_Blocks may be needed. If the full set of AMR Codec Modes is supported, then neither the Attribute_Head nor the SCS_Extension_Block shall be sent for the alternative Codec_Type(s).
The following figures give the examples for the full-set AMR TFO Messages.

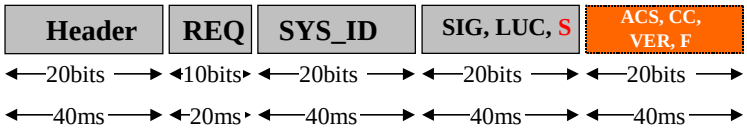


Figure 7.11.3.1-1: Construction of the shortest possible TFO_REQ Message for any AMR Codec Type

TFO_ACK follows the same construction. Both have a length of 180ms.

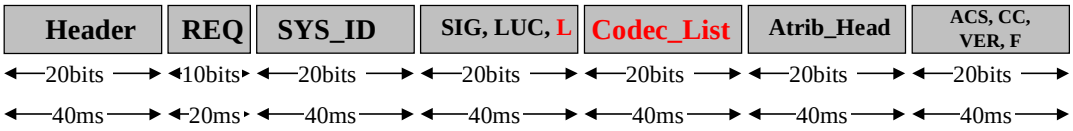


Figure 7.11.3.1-2: Construction of the shortest possible TFO_REQ_L Message listing an AMR Codec_Type in the Codec_List

TFO_ACK_L follows the same construction. Both have a length of 260ms.

NOTE 2: In TFO_REQ_L (TFO_ACK_L) at least one Attribute_Head is needed, if the Local_Used_Codec_Type is AMR or AMR-WB, because otherwise a TFO partner that does not know the Local_Used_Codec_Type cannot know how many attributes are needed – if any. Since these longer messages are used only when mismatch is identified or in other situations, where protocol speed is not important, this additional 40ms message length is not important.

In the worst case in GSM, when both AMR Codec_Types and the FR_AMR-WB are flagged in the Codec_List, but none supports the full set, then seven Extention_Blocks need to follow after the Codec_List.
Example: FR_AMR == Local_Used_Codec_Type: Attribute_Head(FR_AMR) – ACS(FR_AMR) – SCS(FR_AMR) – Attribute_Head(HR_AMR) – SCS(HR_AMR) – Attribute_Head(FR_AMR-WB) – SCS(FR_AMR-WB)

7.11.3.1.1 AMR Active_Codec_Set Attributes

One AMR_ACS (AMR-WB_ACS) Extension_Block shall be added in the TFO_REQ and TFO_ACK messages after the SIG_LUC Extension_Block if an AMR (AMR-WB) Codec_Type is used as the Local_Used_Codec_Type.

Table 7.11.3.1.1-1: AMR_ACS Extension_Block

Bit	Description	Comment
Bit 1	"**0**"	Normal IS-Message Sync Bit, constant.
Bit 2..9	**Active Codec Set (NB_ACS)**	Active Codec Set: For each Codec_Mode of the AMR one bit is reserved. If the bit is set to "0" then the specific Codec_Mode is not in the ACS, otherwise it is in and may be used by the adaptation algorithm. Bit 2: AMR_Mode 12,2 kbit/s (undefined for HR_AMR) Bit 3: AMR_Mode 10,2 kbit/s (undefined for HR_AMR) Bit 4: AMR_Mode 7,95 kbit/s Bit 5: AMR_Mode 7,40 kbit/s Bit 6: AMR_Mode 6,70 kbit/s Bit 7: AMR_Mode 5,90 kbit/s Bit 8: AMR_Mode 5,15 kbit/s Bit 9: AMR_Mode 4,75 kbit/s
Bit 10	**Full_Sub (NB_F/S)**	0: **F**ull Set supported, NB_SCS is not following 1: **S**ubset only supported, NB_SCS is following immediately
Bit 11	"**0**"	Normal IS-Message Sync Bit, constant
Bit 12	spare ~~AMR-WB+~~	set to "1" ~~if this bit is set to "1" (like "spare"), then AMR-WB is not supported.~~ ~~if this bit is set to "0", then AMR-WB is supported. The AMR-WB SCS Extension block is following. If NB_SCS is also following (NB_F/S=1)), then WB_SCS is following after that.~~ ~~Note: a REL-4 TFO Protocol does not understand this and ignores the last extension block with WB_SCS.~~
Bit 13	**Optimisation Mode (NB_OM)**	ACS Optimisation Mode 0 No ACS Change supported 1 ACS change supported
Bit 14 & 15	**NB_Ver**	**Ver**sion Number of the AMR-NB TFO Scheme Bit 15 is equivalent to the ATVN in Configuration Frames, see Annex C
Bit 16..18	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows (i.e. SCS)

Table 7.11.3.1.1-2: AMR-WB_ACS Extension_Block

Bit	Description	Comment
Bit 1	”**0**”	Normal IS-Message Sync Bit, constant.
Bit 2..10	**Active Codec Set (WB_ACS)**	Active Codec Set: For each Codec_Mode of the AMR-WB one bit is reserved. If the bit is set to ”0” then the specific Codec_Mode is not in the ACS, otherwise it is in and may be used by the adaptation algorithm. Bit 2: AMR-WB_Mode 23.85 kbit/s Bit 3: AMR-WB_Mode 23.05 kbit/s Bit 4: AMR-WB_Mode 19.85 kbit/s Bit 5: AMR-WB_Mode 18.25 kbit/s Bit 6: AMR-WB_Mode 15.85 kbit/s Bit 7: AMR-WB_Mode 14.25 kbit/s Bit 8: AMR-WB_Mode 12.65 kbit/s Bit 9: AMR-WB_Mode 8.85 kbit/s Bit 10: AMR-WB_Mode 6.60 kbit/s
Bit 11	”**0**”	Normal IS-Message Sync Bit, constant
Bit 12	**Full_Sub (WB_F/S)**	0: **F**ull Set supported, WB_SCS is not following. 1: **S**ubset only supported, WB_SCS is following immediately
Bit 13	**Optimisation Mode (WB_OM)**	ACS Optimisation Mode 0: No ACS Change supported 1: ACS Change supported
Bit 14	spare ~~**AMR-NB+**~~	set to "1" ~~1: AMR-NB is not supported~~ ~~0: AMR-NB is supported, NB_SCS is following. If WB_SCS is also following (WB_F/S=1), then NB_SCS is following after that.~~
Bit 15	spare ~~**WB_Ver**~~	set to "1" ~~**Ver**sion Number of the AMR-WB TFO Scheme~~ ~~Bit 15 is equivalent to the ATVN in Configuration Frames, see Annex C~~
Bit 16..18	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows (i.e. SCS)

7.11.3.1.2 AMR Supported_Codec_Set Attributes

The AMR_SCS (AMR-WB_SCS) Extension_Block contains the information on the AMR (AMR-WB) Supported Codec Set. It shall be omitted, if the full set is supported. Table 7.11.3.1.2-1 gives the description of the SCS Extension_Block.
For the Local_Used_Codec_Type the SCS Extension_Block shall follow immediately after the corresponding ACS Extension_Block. In that case the Full_Sub flag shall be set within the ACS Extension_Block. For alternative Codec_Types, as flagged in the Local_Codec_List, the SCS shall follow immediately after the corresponding Attribute_Head Extension_Block.

NOTE: The VERsion numbers in ACS and SCS Extension_Blocks shall be identical for one Codec_Type, but may be different for different Codec_Types (e.g. FR_AMR and HR_AMR or FR_AMR-WB).

Table 7.11.3.1.2-1: AMR_SCS Extension_Block

Bit	Description	Comment
Bit 1	"**0**"	Normal IS-Message Sync Bit, constant.
Bit 2…9	**Supported Codec Set (NB_SCS)**	Supported Codec Set: For each Codec_Mode of the AMR one bit is reserved. If the bit is set to "0" then the specific Codec_Mode is not supported; if the bit is set to "1" then the specific Codec_Mode is supported and may be considered for the optimisation of the common ACS. Bit 2: AMR_Mode 12,2 kbit/s (undefined in SCS(H)) Bit 3: AMR_Mode 10,2 kbit/s (undefined in SCS(H)) Bit 4: AMR_Mode 7,95 kbit/s Bit 5: AMR_Mode 7,4 kbit/s Bit 6: AMR_Mode 6,7 kbit/s Bit 7: AMR_Mode 5,9 kbit/s Bit 8: AMR_Mode 5,15 kbit/s Bit 9: AMR_Mode 4,75 kbit/s
Bit 10	**NB_MACS MSB**	See comment for Bit 12…13
Bit 11	"**0**"	normal IS-Message Sync Bit, constant
Bit 12…13	**NB_MACS LSBs**	The maximally supported number of Codec_Modes in this radio leg. Coding for bits 10.12.13: "0.0.1" 1 Mode "0.1.0" 2 Modes "0.1.1" 3 Modes "1.0.0" 4 Modes "1.0.1" 5 Modes "1.1.0" 6 Modes "1.1.1" 7 Modes "0.0.0" 8 Modes
Bit 14…15	**NB_Ver**	Version Number of the AMR TFO Scheme for that Codec_Type Bit 15 is equivalent to the ATVN in Configuration Frames, see Annex C
Bit 16..18	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19 20	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows

Table 7.11.3.1.2-2: AMR-WB_SCS Extension_Block

Bit	Description	Comment
Bit 1	”**0**”	Normal IS-Message Sync Bit, constant.
Bit 2…10	**Supported Codec Set (WB_SCS)**	Supported Codec Set: For each Codec_Mode of the AMR-WB one bit is reserved. If the bit is set to ”0” then the specific Codec_Mode is not supported; if the bit is set to ”1” then the specific Codec_Mode is supported and may be considered for the optimisation of the common WB_ACS. Bit 2: AMR-WB_Mode 23.85 kbit/s Bit 3: AMR-WB_Mode 23.05 kbit/s Bit 4: AMR-WB_Mode 19.85 kbit/s Bit 5: AMR-WB_Mode 18.25 kbit/s Bit 6: AMR-WB_Mode 15.85 kbit/s Bit 7: AMR-WB_Mode 14.25 kbit/s Bit 8: AMR-WB_Mode 12.65 kbit/s Bit 9: AMR-WB_Mode 8.85 kbit/s Bit 10: AMR-WB_Mode 6.60 kbit/s
Bit 11	”**0**”	normal IS-Message Sync Bit, constant
Bit 12…14	**WB_MACS**	The maximally supported number of Codec_Modes in this radio leg. Coding: "0.0.1" 1 Mode "0.1.0" 2 Modes "0.1.1" 3 Modes “1.0.0” 4 Modes “1.0.1” 5 Modes ”1.1.0” 6 Modes ”1.1.1” 7 Modes "0.0.0" 8 Modes
Bit 15	spare ~~WB_Ver~~	set to "1" ~~Version Number of the AMR-WB TFO Scheme.~~ ~~Bit 15 is equivalent to the ATVN in Configuration Frames, see Annex C~~
Bit 16..18	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19 20	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows

7.11.3.1.3 AMR specific Codec_Attribute_Head Extension_Block

The AMR specific Codec_Attribute_Head Extension_Block (Table 7.11.3.1.3-1) shall precede the Codec Attribute Extension_Blocks of any AMR Codec_Type.

Table 7.11.3.1.3-1: AMR specific Codec_Attribute_Head Extension_Block

Bit	Description	Comment
Bit 1	"0"	normal IS-Message Sync Bit, constant.
Bit 2	**PAR_Sel**	Differentiates this Extension_Block **0:** Parameters included in **PAR** field: Simple Codec_List_Extension **1:** Length Indicator (**LI**) included: Parameters follow in subsequent Extension_Blocks
Bit 3..10	**CoID = HR_AMR or FR_AMR or UMTS_AMR or UMTS_AMR2 or OHR_AMR**	This field identifies the AMR Codec_Type for which the subsequent attributes are valid. The same coding as in the Codec_x Extension_Block is used (long form)
Bit 11	"0"	normal IS-Message Sync Bit, constant
Bit 12.. 15:	**LI / PAR**	If Par_Sel==1: LI: Length Indicator: 0000: reserved; 0001: one other Extension_Block follows, etc. If Par_Sel==0: PAR: Codec specific definition of these four bits
Bit 16..18:	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows

If PAR_Sel is set to "1" then the AMR_ACS and potentially AMR_SCS is/are following.

If PAR_Sel is set to "0", then one of 16 possible AMR Configurations is indicated in the PAR field and no additional Codec Attribute Extension_Blocks do follow.
Coding for PAR (bits 12.13.14.15):
0000: to
1111: for further study.

The option" Par_Sel=0" and the corresponding configuration codes can only be used in TFO Version 5 and onwards. A Pre-REL-5 implementation does not understand it and ignores it.

7.11.3.1.4 AMR-WB specific Codec_Attribute_Head Extension_Block

The AMR-WB specific Codec_Attribute_Head Extension_Block (Table 7.11.3.1.4-1) shall precede the Codec Attribute Extension_Blocks of any AMR-WB Codec_Type.

Table 7.11.3.1.4-1: AMR-WB specific Codec_Attribute_Head Extension_Block

Bit	**Description**	**Comment**
Bit 1	**"0"**	normal IS-Message Sync Bit, constant.
Bit 2	**PAR_Sel**	Differentiates this Extension_Block **0:** Parameters included in **PAR** field: Simple Codec_List_Extension **1:** Length Indicator (**LI**) included: Parameters follow in subsequent Extension_Blocks
Bit 3..10	**CoID = FR_AMR-WB or UMTS_AMR-WB or OHR_AMR-WB or OFR_AMR-WB**	This field identifies the AMR-WB Codec_Type for which the subsequent attributes are valid. The same coding as in the Codec_x Extension_Block is used (long form)
Bit 11	**"0"**	normal IS-Message Sync Bit, constant
Bit 12.. 15:	**LI / PAR**	If Par_Sel==1: LI: Length Indicator: 0000: reserved; 0001: one other Extension_Block follows, etc. If Par_Sel==0: PAR: Codec specific definition of these four bits
Bit 16..18:	**CRC**	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	**EX**	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows

If PAR_Sel is set to "1" then the AMR-WB_ACS and potentially AMR-WB_SCS is/are following.

If PAR_Sel is set to "0" then one of 16 possible AMR-WB Configurations is indicated in the PAR field and no additional Codec Attribute Extension_Blocks do follow.
Coding for PAR (bits 12.13.14.15):
0000: to
1111: for further study.

3GPP TSG-SA4 #21
Rennes, France, 13-17 May 2002

S4-020352

CR-Form-v6.1

CHANGE REQUEST

⌘	TS 28.062	CR	020	⌘ rev	2	⌘	Current version:	5.0.0	⌘
	Spec Title:	Inband Tandem Free Operation (TFO) of speech codecs							⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network X Core Network X

Title:	⌘	**Corrections to TS 28.062, sections 4 to 8**
Source:	⌘	**TSG SA WG4**
Work item code:	⌘	**AMRWB**
Date:	⌘	2002-06-11
Category:	⌘	**F**
Release:	⌘	**REL-5**

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
REL-4	*(Release 4)*
REL-5	*(Release 5)*

Reason for change:	⌘	Inconsistencies
Summary of change:	⌘	Smaller details, but more than editorial
Consequences if not approved:	⌘	Spec. is wrong or potentially misleading

Clauses affected:	⌘	4 to 8		
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ O&M Specifications	⌘	
Other comments:	⌘			

4.2.2 Principle for TFO Operation for Wide-Band speech codec types (i.e. AMR-WB)

In case of AMR-WB the TRAU/TC performs in uplink direction the wideband decoding and a successive lowpass-filtering, downsampling to 8kHz sampling rate and PCM (G.711) encoding, before its sends the narrowband version of the speech signal towards its destination. This downsampled speech signal in PCM (G.711) representation allows interworking with the narrowband world (PSTN etc.). If a 64kbit/s channel is used, then a transcoded wideband signal (7 kHz speech bandwidth and 16kHz sampling rate) would anyway not fit into it. An efficient way to transport the wideband signal via such a channel is to use TFO (or TrFO) which delivers the compressed (encoded) speech. The encoded speech has a bandwidth significantly lower than 64kbit/s. In TFO_State OPERATION the TRAU/TC sends the AMR-WB TFO Frames within the LSBs of this PCM signal.

In the other, downlink direction the TRAU/TC performs G.711 decoding, upsampling to 16 kHz sampling rate, lowpass- filtering and wideband encoding before it sends the AMR-WB parameters down to the A/Iu interface. In TFO_State OPERATION the TRAU sends the AMR-WB parameters as received via the TFO Frames downlink.

A listener on the A/Iu interface will always hear the narrowband version of the speech conversation, while both ends send and receive the wideband version.

The basic principle for TFO operation for WB speec codec tpyes is the same as for narrow-band speech codec types (see section before). The following items must additionally be considered:

- A new size of ~~:~~640 bits for the 32 kbit/s TFO Frames format is needed in case the highest AMR-WB modes shall be used (the related TRAU format is defined in 48.060).

- The scenario in figure 4.2.2-1 shows the situation when AMR-WB TFO has not yet been established while the call started with a narrowband codec. This is a likely starting scenario, because it it not desirable to occupy radio ressources unnecessarily with wide-band signals, until TFO is operational.

Nex modification

5.2.2.1 TFO Frame Format AMR_TFO_16k

TFO Frames with format AMR_TFO_16k are derived from the TRAU Frames for Adaptive Multi Rate as defined in the 3GPP TS 48.060. The AMR_TFO_16k Frame structure is illustrated in Figure 5.2.2.1-1, using the same notations as in 3GPP TS 48.060. Table 5.2.2-1 defines the coding of the Control Bits for AMR TFO Frames. Note that additional TFO Configuration Parameters may be carried by the Data Bits of the TFO Frames, as defined in annex C.

	Bit number							
Octet no.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	1	C1	C2	C3	C4	C5	C6	C7
3	C8	C9	C10	C11	C12	C13	C14	C15
4	1	C16	C17	C18	C19	C20	C21	C22
5	C23	C24	C25	D1	D2	D3	D4	D5
6	1	D6	D7	D8	D9	D10	D11	D12
7	D13	D14	D15	D16	D17	D18	D19	D20
8..36								
37	D238	D239	D240	D241	D242	D243	D244	D245
38	1	D246	D247	D248	D249	D250	D251	D252
39	D253	D254	D255	D256	T1	T2	T3	T4

Figure 5.2.2.1-1: Structure of AMR_TFO_16k Frames

Table 5.2.2.1-2: Coding of the Control Bits for AMR_TFO_16k Frames

Control Bits	Description		Comment	
	FR_AMR, HR_AMR, UMTS_AMR_2, OHR_AMR	**UMTS_AMR**	**FR_AMR, HR_AMR, OHR_AMR**	**UMTS_AMR, UMTS_AMR_2**
C1 - C4 (0.0.0.1) 0.0.1.1 0.1.0.0 0.1.0.1 0.1.1.0 (1.0.0.1) (1.0.1.0) 1.0.1.1 (1.1.0.0) (0.0.1.0) (1.1.0.1)	**Frame_Type / Codec Type** (GSM_FR) **FR_AMR** **HR_AMR** **UMTS_AMR** **UMTS_AMR_2** (FR_AMR-WB) (UMTS_AMR-WB), **OHR_AMR** (OFR_AMR-WB), (OHR_AMR-WB) (GSM_EFR)		The coding is different from the coding in TFO Messages. It is also not identical to the coding on Abis/Ater. The TRAU shall translate the coding between TRAU and TFO Frames Codec Types in (brackets) are not supported by this TFO Frame format. They are listed to show their coding for convenience.	
C5 0 1	**EMBED** No TFO Message embedded A TFO Message is embedded		Indicates the presence of an embedded TFO Message. Set by the TRAU.	
C6 – C8	Set to "1.1.1"(see note)	**Codec Mode Request (CMR)**)	In GSM TRAU Frames, these bits carry part of the Time Alignment. They are set to 1.1.1 by the TRAU.	Coding as defined in 3GPP TS 48.060
C9 - C11 0.0.0 0.0.1 0.1.0 0.1.1 1.0.0 1.0.1 1.1.0 1.1.1	TFO andHandover_Notifications TFO_On TFO_Soon TFO_Off Handover_Soon Handover_Complete undefined undefined undefined		In GSM TRAU Frames these bits are part of the Time Alignment field. These bits are copied from TRAU frames to TFO Frames and vice versa. TFO_On is the default value in TFO Frames.	
C12	**RIF (Request or Indication Flag)**	set to 0	Copied from the uplink TRAU Frame in GSM Generated by the Transcoder in 3G systems for FR_AMR and HR_AMR: The changes of the uplink Codec Mode, as received via the Iu Frames, are monitored. Whenever the Codec Mode changes, the RIF bit is set to "0". The next frames are then alternatingly marked with RIF = "1", "0", "1" and so on.	
C13	Spare (set to 1)		C13 is spare in UL TRAU frames.	
C14 – C16	**Config_Prot**		Coding defined in Annex C.	
C17 C18	**Mess No**		Coding defined in Annex C.	
C19	DTXd (see note)		Copied from uplink TRAU Frame in GSM	
C20 0 1	**TFOE** TFO Disable TFO Enable		Copied from the uplink TRAU Frame in GSM Generated by the Transcoder in 3G systems with the same coding as in the 3GPP TS 48.060	
C21 – C22 1 1 1 0 0 1 0 0	**Frame_Classification** "Speech_Good" "Speech_Degraded" "Speech_Bad" "No_Speech"		Copied from the uplink TRAU Frame in GSM Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.2.2.1-3 below)	

C23 – C25	**(see 3GPP TS 48.060)** CMI (if RIF == 0) or CMR (if RIF == 1) or 0.0.0 (if Frame_Classification == 0.0)	**Codec Mode Indication (CMI)**; (RIF ==0 is always the case in UMTS_AMR)	Carry CMI or CMR depending of the value of RIF, if the Frame Classification bits are different from "0 0" (No_Speech), and set to "000" otherwise. Copied from the uplink TRAU Frame in GSM Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.2.2.1-3)	Coding as defined in **3GPP** TS 48.060
T1 - T4	**Time Alignment Bits**		In GSM copied from the uplink TRAU Frame In 3G, generated by the TC (UMTS) based on Iu Frame arrival time(s)	

NOTE 0: Any spare control bits shall be coded as binary "1". They are reserved for future use and may change.

The CRC1 covering also the control bits C1..C25 shall be recomputed in the transcoders.

The coding of the **Data Bits** is described in 3GPP TS 48.060.

In 3G systems, the Frame_Classification Bits must be derived from the Frame Quality Indicator (FQI) and Frame Type Index as defined in the 3GPP TS 26.101. Table 5.2.2.1-3 provides the conversion rules between the generic AMR Frames (as defined in 3GPP TS 26.101) and TFO Frames. In this table, the arrows in the fourth column indicate the direction for which the conversion applies.

NOTE 1: A one-to-one relationship between Generic AMR Frames and TFO Frames does not always exist, but the conversion is always possible.

NOTE 2: In the generic AMR Frames (3GPP TS 26.101), the differentiation between SID_FIRST and SID_UPDATE is done in the Data bits (SID Type Indicator). The Codec Mode Indication (CMI) is carried in 3G systems within the SID payload.

For ~~2G and 3G systems using~~ the FR_AMR, HR_AMR, UMTS_AMR_2 and OHR_AMR ~~FR_AMR or HR_AMR Speech~~ Codec Types, bits C23 - C25 shall carry either the Codec Mode Request (CMR) or the Codec Mode Indication (CMI), depending on the value of RIF, if the Frame_Classification bits are different from "0.0". If the Frame_Classification bits are equal to "0.0" (SID_First and SID_Update Frames), C23 - C25 are set to "0.0.0", and the CMI and CMR are carried in the data bits D35 - D40.

For 3G systems using the UMTS_AMR_2 ~~or FR_AMR or OHR_AMR Speech~~ Codec Types, the TC shall monitor the changes of the uplink Codec Mode, as received in the Iu Frames. Every time the Codec Mode changes in the Iu Frames the TC shall set RIF = "0" in the corresponding TFO Frame. The next TFO Frames are alternatively marked with RIF = "1", "0", "1" and so on.

NOTE 3: Per definition for UMTS_AMR_2 ~~or FR_AMR or OHR_AMR~~ the UE shall select the phase of potential Codec Mode changes in uplink once at call set-up and shall not alter this later on. At call set-up TFO is not active and the TC has enough time to find the phase of the RIF by the proposed implicit method, before the first TFO Frame has to be sent.

Nex modification

5.2.2.3 TFO Frame Format ~~FR_~~AMR_WB_TFO_16k and AMR_WB_TFO_32k

TFO Frames with format AMR_WB_TFO_16k and AMR_WB_TFO_32k are derived from the TRAU Frames for ~~Wide Band~~ Adaptive Multi-Rate Wide Band as defined in the 3GPP TS 48.060. The AMR_WB_TFO_16k Frame structure is illustrated in ~~Figures~~Table 5.2.2.3-1 ~~and 5.2.2.3-2~~ below, using the same notations as in 3GPP TS 48.060.

For AMR_WB_TFO_32k Frames the identical frame structure is used twice, once in the lower 16k main part (identical to the AMR_WB_TFO_16k) and in the upper 16k extension part (carrying some data bits, but no synchhronisation and no control bits, see Table 5.2.2.3-2). The unspecified bits in Table 5.2.2.3-2 shall not alter the bits of the PCM samples on the 64 kbit/s A interface.

Table 5.2.2.3-3 defines the coding of the Control Bits for the Frame Type (== Codec Type) field (C1..C4) in AMR_WB_TFO_16k and AMR_WB_TFO_32k frames. For the remaining control bits (C5…C25) the definition is as

for AMR_TFO_16k frames for FR_AMR. ~~AMR-WB TFO Frames. Note that additional TFO Configuration Parameters may be carried by the Data Bits of the TFO Frames, as defined in Annex C.~~

Table 5.2.2.3-1: Structure of AMR_WB_TFO_16k Frames and the lower 16k main part of AMR_WB_TFO_32k Frames

			Bit number					
Octet no.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
0a	0	0	0	0	0	0	0	0
1a	0	0	0	0	0	0	0	0
2a	1	C1	C2	C3	C4	C5	C6	C7
3a	C8	C9	C10	C11	C12	C13	C14	C15
4a	1	C16	C17	C18	C19	C20	C21	C22
5a	C23	C24	C25	D1	D2	D3	D4	D5
6a	1	D6	D7	D8	D9	D10	D11	D12
7a	D13	D14	D15	D16	D17	D18	D19	D20
8a..36a								
37a	D238	D239	D240	D241	D242	D243	D244	D245
38a	1	D246	D247	D248	D249	D250	D251	D252
39a	D253	D254	D255	D256	T1	T2	T3	T4

Table 5.2.2.3-2: Structure of the upper 16k extension part in AMR_WB_TFO_32k Frames

			Bit number					
Octet no.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
0b								
1b								
2b								
3b								
4b								
5b				D1	D2	D3	D4	D5
6b		D6	D7	D8	D9	D10	D11	D12
7b	D13	D14	D15	D16	D17	D18	D19	D20
8b..36b								
37b	D238	D239	D240	D241	D242	D243	D244	D245
38b	1	D246	D247	D248	D249	D250	D251	D252
39b	D253	D254	D255	D256				

Table 5.2.2.3-3: Coding of the Frame Type for AMR_WB_TFO_16k Frames and AMR_WB_TFO_32k Frames

Control Bits	**Description**	**Comment**
C1 - C4	**Frame_Type / Codec Type**	The coding is different from the coding in TFO Messages. It is also not identical to the coding on Abis/Ater. The TRAU shall translate the coding between TRAU and TFO Frames. Note: Codec Types in (brackets) are not supported by this TFO Frame format. They are listed to show their coding for convenience. Note: By definition FR_AMR-WB and OHR_AMR-WB do only use the AMR_WB_TFO_16k Frame, because they never use a Codec Mode higher than 12.65 kbit/s. UMTS_AMR-WB and OFR_AMR-WB use the AMR_WB_TFO_32k Frame when at least one Codec Mode is above 12.65 kbit/s.
(0.0.0.1)	(GSM_FR)	
(0.0.1.1)	(FR_AMR)	
(0.1.0.0)	(HR_AMR)	
(0.1.0.1)	(UMTS_AMR)	
(0.1.1.0)	(UMTS_AMR_2)	
1.0.0.1	**FR_AMR-WB**	
1.0.1.0	**UMTS_AMR-WB**	
(1.0.1.1)	(OHR_AMR)	
1.1.0.0	**OFR_AMR-WB**	
0.0.1.0	**OHR_AMR-WB**	
(1.1.0.1)	(GSM_EFR)	

Figure 5.2.2.3-1: Stucture of AMR_WB_TFO_16k Frames for Codec Mode 14.25 kbit/s

	Bit number							
Octet no.	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	1	C1	C2	C3	C4	C5	C6	C7
3	C8	C9	C10	C11	C12	C13	D1	D2
4	D3	D4	D5	D6	D7	D8	D9	D10
5	D11	D12	D13	D14	D15	D16	D17	D18
6	D19	D20	D21	D22	D23	D24	D25	D26
7	D27	D28	D29	D30	D31	D32	D33	D34
8..36								
37	D267	D268	D269	D270	D271	D272	D273	D274
38	D275	D276	D277	D278	D279	D280	D281	D282
39	D283	D284	D285	D286	D287	D288	T1	T2

Figure 5.2.2.3-2: Stucture of AMR_WB_TFO_16k Frames for No_Speech frames, Codec Modes 12.65, 8.85 and 6.60 kbit/s

	Bit number							
Octet no.	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	1	C1	C2	C3	C4	C5	C6	C7
3	C8	C9	C10	C11	C12	C13	D1	D2
4	1	D3	D	D5	D6	D7	D8	D9
5	D10	D11	D12	D13	D14	D15	D16	D17
6	1	D18	D19	D20	D21	D22	D23	D24
7	D25	D26	D27	D28	D29	D30	D31	D32
8..36								
37	D250	D251	D252	D253	D254	D255	D256	D257
38	1	D258	D259	D260	D261	D262	D263	D264
39	D265	D266	D267	D268	D269	D270	T1	T2

Table 5.2.2.3-3: Coding of the Control Bits for AMR_WB_TFO_16k Frames

Control Bits	Description	Comment
	FR_AMR-WB, UMTS_AMR-WB, OFR_AMR-WB, OHR_AMR-WB	FR_AMR-WB, UMTS_AMR-WB, **OFR_AMR-WB, OHR_AMR-WB**
C1 - C4 (0.0.0.1) (0.0.1.1) (0.1.0.0) (0.1.0.1) (0.1.1.0) **1.0.0.1** **1.0.1.0** (1.0.1.1) **1.1.0.0** **0.0.1.0** (1.1.0.1) C1 - C4 1.0.1.1 1.0.1.0	**Frame_Type / Codec Type** (GSM_FR) (FR_AMR) (HR_AMR) (UMTS_AMR) (UMTS_AMR_2) **FR_AMR-WB** **UMTS_AMR-WB** (OHR_AMR) **OFR_AMR-WB** **OHR_AMR-WB** (GSM_EFR) Frame_Type / Codec Type FR_AMR_WB UMTS_AMR_WB	The coding is different from the coding in TFO Messages. It is also not identical to the coding on Abis/Ater. The TRAU shall translate the coding between TRAU and TFO Frames. Codec Types in (brackets) are not supported by this TFO Frame format. They are listed to show their coding for convenience.
C5a 0 1	**EMBED** No TFO Message embedded A TFO Message is embedded	Indicates the presence of an embedded TFO Message. Set by the TRAU.
C6	**RIF (Request or Indication flag)**	Copied from the uplink TRAU frame in GSM. Generated by the Transocder in 3G systems for FR_AMR and HR_AMR. The changes of the uplink Codec Mode, as received via the Iu Frames, are monitored. Whenever the Codec Mode changes, the RIF bit is set to "0". The next frames are then alternatingly marked with RIF = "1", "0", "1" and so on.
C7	set to 1	Copied from the uplink TRAU Frame in GSM. Generated by the TC in UMTS.
C8	**DTXd**	Coding defined in Annex C.
C9 0 1	**TFOE** TFO Disable TFO Enable	Copied from the uplink TRAU Frame in GSM Generated by the Transcoder in 3G systems with the same coding as in the 3GPP TS 48.060
C10 – C11 1 1 1 0 0 1 0 0	**Frame_Classification** "Speech_Good" "Speech_Degraded" "Speech_Bad" "No_Speech"	Copied from the uplink TRAU Frame in GSM Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.2.2.3-4 below)
C10 – C11 1 1 1 0 0 1 0 0	**Frame_Classification** "Speech_Good" "Speech_Degraded" "Speech_Bad" "No_Speech"	Copied from the uplink TRAU Frame in GSM Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.2.2.3-4 below)
C10 – C11 1 1 1 0 0 1 0 0	**Frame_Classification** "Speech_Good" "Speech_Degraded" "Speech_Bad" "No_Speech"	Copied from the uplink TRAU Frame in GSM Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.2.2.3-4 below)
C12-C13 C25a and C23b- C25b	**(see 3GPP TS 48.060)** CMI (if RIF == 0) or CMR (if RIF == 1) or 0.0.0 (if Frame_Classification == 0.0)	Carry CMI or CMR depending of the value of RIF, if the Frame Classification bits are different from "0 0" (No_Speech), and set to "000" otherwise. Copied from the uplink TRAU Frame in GSM Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.2.2.3-4 below)
T1 – T2	**Time Alignment Bits**	In GSM copied from the uplink TRAU Frame In 3G, generated by the TC (UMTS) based on Iu Frame arrival time(s)

NOTE: Any spare control bits shall be coded as binary "1". They are reserved for future use and may change.

The CRC~~1~~ covering also the control bits C1..C25~~13~~ shall be recomputed in the transcoder~~s~~, because some control bits change between TRAU Frames and TFO Frames, e.g. the coding of the Frame Type.

The coding of the **Data Bits** is described in 3GPP TS 48.060. In AMR_WB_TFO_32k Frames the data bits in the upper 16k extension part shall be set as defined in TS 48.060. But in all unused bit positions of this upper extension part the bits of the PCM samples shall not be altered in order to minimise the audible effect.

In 3G systems, the Frame_Classification Bits must be derived from the Frame Quality Indicator (FQI) and Frame Type Index as defined in the 3GPP TS 26.2~~1~~01. ~~Table 5.2.2.3-4 provides the conversion rules between the generic WB-AMR Frames (as defined in 3GPP TS 26.101) and TFO Frames. In this table, the arrows in the fourth column indicate the direction for which the conversion applies.~~The conversion rules are the same as for the FR_AMR.

NOTE 1: A one-to-one relationship between Generic WB AMR Frames and TFO Frames does not always exist, but the conversion is always possible.

NOTE 2: In the generic WB AMR Frames (3GPP TS 26.2~~1~~01), the differentiation between SID_FIRST and SID_UPDATE is done in the Data bits (SID Type Indicator). The Codec Mode Indication (CMI) is carried in 3G systems within the SID payload.

~~For 2G using the FR_AMR-WB, UMTS_AMR-WB, OFR_AMR-WB or OHR_AMR-WB and 3G systems using UMTS_AMR-WB, OFR_AMR-WB or OHR_AMR-WB bits C12 – C13 shall carry either the Codec Mode Request (CMR) or the Codec Mode Indication (CMI), depending on the value of RIF, if the Frame_Classification bits are different from "0.0". If the Frame_Classification bits are equal to "0.0" (NO_SPEECH Frames), C12 – C13 are set to 0.0, and the CMI and CMR are carried in the data bits D35 – D40.~~

3G systems using the UMTS_AMR-WB~~, OFR_AMR-WB or OHR_AMR-WB Speech~~ Codec Type, the TC shall monitor the changes of the uplink Codec Mode, as received in the Iu Frames. Every time the Codec Mode changes in the Iu Frames the TC shall set RIF = "0" in the corresponding TFO Frame. The next TFO Frames are alternatively marked with RIF = "1", "0", "1" and so on.

NOTE 3: Per definition for UMTS_AMR-WB ~~FR_WB-AMR~~ the UE selects the phase of potential Codec Mode changes in uplink once at call set-up and does not alter this later on. At call set-up TFO is not active and the TC has enough time to find the phase of the RIF by the proposed implicit method, before the first TFO Frame has to be sent.

~~Table 5.2.2.3-4: Conversion between Generic AMR Frames and FR_WB_AMR_TFO_16k Frames~~

~~Generic AMR Frame~~				~~AMR_WB_TFO_16k Frame~~			
~~Frame Quality Indicator~~	~~Frame Type Index~~	~~TX_TYPE or RX_TYPE (see 3GPP TS 26.101)~~		~~Frame_ Classification C10 – C11~~	~~CMI or CMR C12 – C13~~	~~Data bits in No_Speech frames D32 .. D34~~	~~Equivalent Frame Type in 3GPP TS 48.060)~~
~~1~~	~~0-7~~	~~SPEECH_GOOD~~	~~<->~~	~~1 1~~	~~0-7~~	~~-~~	~~Speech_Good~~
~~1~~	~~0-7~~	~~SPEECH_GOOD~~	~~<~~	~~1 0~~	~~0-7~~	~~-~~	~~Speech_Degraded~~
~~0~~	~~0-7~~	~~SPEECH_BAD~~	~~<->~~	~~0 1~~	~~0-7~~	~~-~~	~~Speech_Bad~~
~~1~~	~~8~~	~~SID_FIRST~~	~~<->~~	~~0 0~~	~~0 0 0~~	~~SID_First~~	~~No_Speech~~
~~1~~	~~15~~	~~NO_DATA~~	~~<~~	~~0 0~~	~~0 0 0~~	~~Onset~~	~~No_Speech~~
~~1~~	~~8~~	~~SID_UPDATE~~	~~<->~~	~~0 0~~	~~0 0 0~~	~~SID_Update~~	~~No_Speech~~
~~0~~	~~8~~	~~SID_BAD~~	~~<->~~	~~0 0~~	~~0 0 0~~	~~SID_Bad~~	~~No_Speech~~
~~1~~	~~15~~	~~NO_DATA~~	~~<->~~	~~0 0~~	~~0 0 0~~	~~No_Data~~	~~No_Speech~~

The **Synchronisation Pattern** is similar to the Synchronisation Pattern in 3GPP TS 48.060, with some exceptions related to the value of the EMBED Bit:

EMBED equal "0": the Synchronisation Pattern is exactly as described in the 3GPP TS 48.060;
EMBED equal "1": the Synchronisation Pattern contains an embedded TFO Message.

For the coding of the **Data Bits** see 3GPP TS 48.060. ~~and Annex C for the bits reserved for TFO Configuration Parameters.~~

For the coding of the **Time Alignment Bits** (T_Bits, T1 .. T4~~2~~) see 3GPP TS 48.060 ~~and Annex C~~. When the TFO Frame is generated by a GSM Network, the T_Bits normally correspond to the T_Bits received in the up-link TRAU Frame.

5.2.3 Transmission of the bits of 16 kbit/s TFO Frames

For the purpose of this description the 320 bits of one TFO Frame are arranged in 40 rows (0..39), with 8 bit each (1..8: one octet) as in 3GPP TS 48.060.

The bits of 16 kbit/s TFO Frames are transmitted in the following order:

Bit m of octet n, shall be transmitted in the **Least** Significant Bit of the

PCM sample k = n*4 + (m+1)/2 for m = (1, 3, 5, 7) and n = (0..39).

Bit m of octet n shall be transmitted in the **second Least** Significant Bit of the

PCM sample k = n*4 + m/2 for m = (2, 4, 6, 8) and n = (0..39).

PCM sample (k=1) is the first PCM sample of the TFO Frame, which follows the received uplink TRAU frame with a small delay (Tultfo), as described in clause 8, see figure 8.1.2-1.

5.2.3 Transmission of the bits of 32 kbit/s TFO Frames

For the purpose of this description the 640 bits of one TFO Frame are arranged in 2 x 40 rows (0a..39a, 0b…39b), with 8 bit each (1..8: one octet) as in 3GPP TS 48.060, see also Table 5.2.2.3-1and Table 5.2.2.3-2.

The bits of 32 kbit/s TFO Frames are transmitted in the following order:

Bit m of octet n, shall be transmitted in the **Least** Significant Bit of the

PCM sample k = n*4 + (m+1)/2 for m = (1, 3, 5, 7) and n = (0a...39a).

Bit m of octet n shall be transmitted in the **second Least** Significant Bit of the

PCM sample k = n*4 + m/2 for m = (2, 4, 6, 8) and n = (0a..39a).

Bit m of octet n, shall be transmitted in the **third Least** Significant Bit of the

PCM sample k = n*4 + (m+1)/2 for m = (1, 3, 5, 7) and n = (0b...39b).

Bit m of octet n shall be transmitted in the **forth Least** Significant Bit of the

PCM sample k = n*4 + m/2 for m = (2, 4, 6, 8) and n = (0b..39b).

PCM sample (k=1) is the first PCM sample of the TFO Frame, which follows the received uplink TRAU frame with a small delay (Tultfo), as described in clause 8, see figure 8.1.2-1.

It is important that the lower main 16k part and the upper 16k extension part are exactly synchronised as described above, see also clause 8.

Nex modification

5.2.5 Optional AMR_TRAU_8+8k Frames

For TFO Connections with FR_AMR on the local side and HR_AMR on the distant side the local side may use the AMR_TRAU_8+8k frame format after TFO has been established. The AMR_TRAU_8+8k Frame is based on the

TRAU Frame formats for the AMR for 8 kBit/s sub-multiplexing as defined in 3GPP TS 428.061 (TRAU_8k), with the additional Synchronisation pattern as defined in Figure 5.2.2.2-4. The differences to AMR_TFO_8+8k frames are:

Nex modification

5.4 TFO Frames for 32 kbit/s sub-multiplexing

5.4.1 TFO Frame Format AMR_WB_TFO_32k

TFO Frames with format AMR_WB_TFO_32k are derived from the TRAU Frames for Wide Band Adaptive Multi Rate as defined in the 3GPP TS 48.060. The AMR_WB_TFO_32k Frame structure is illustrated in figures 5.4.1-1 and 5.4.1-2 below, using the same notations as in 3GPP TS 48.060. Table 5.4.1-3 defines the coding of the Control Bits for AMR WB TFO Frames. Note that additional TFO Configuration Parameters may be carried by the Data Bits of the TFO Frames, as defined in Annex C.

In the following, the control bits C1 to C25 refer to both sub-channels, the control bits C1a to C25a refer to the sub-channel a and the control bits C1b to C25b refer to the sub-channel b.

Figure 5.4.1-1: Stucture of AMR_WB_TFO_32k Frames, first channel (channel a)

	Bit number							
Octet no.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	1	C1a	C2a	C3a	C4a	C5a	C6a	C7a
3	C8a	C9a	C10a	C11a	C12a	C13a	C14a	C15a
4	1	C16a	C17a	C18a	C19a	C20a	C21a	C22a
5	C23a	C24a	C25a	D1a	D2a	D3a	D4a	D5a
6	1	D6a	D7a	D8a	D9a	D10a	D11a	D12a
7	D13a	D14a	D15a	D16a	D17a	D18a	D19a	D20a
8..36								
37	D238a	D239a	D240a	D241a	D242a	D243a	D244a	D245a
38	1	D246a	D247a	D248a	D249a	D250a	D251a	D252a
39	D253a	D254a	D255a	D256a	T1	T2	T3	T4

Figure 5.4.1-2: Stucture of AMR_WB_TFO_32k Frames, second channel (channel b)

	Bit number							
Octet no.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	1	C1b	C2b	C3b	C4b	C5b	C6b	C7b
3	C8b	C9b	C10b	C11b	C12b	C13b	C14b	C15b
4	1	C16b	C17b	C18b	C19b	C20b	C21b	C22b
5	C23b	C24b	C25b	D1b	D2b	D3b	D4b	D5b
6	1	D6b	D7b	D8b	D9b	D10b	D11b	D12b
7	D13b	D14b	D15b	D16b	D17b	D18b	D19b	D20b
8..36								
37	D238b	D239b	D240b	D241b	D242b	D243b	D244b	D245bb
38	1	D246b	D247b	D248b	D249b	D250b	D251b	D252b
39	D253b	D254b	D255b	D256b	T1	T2	T3	T4

Table 5.4.1-3: Coding of the Control Bits for AMR_WB_TFO_32k Frames

Control Bits	Description	Comment
	FR_AMR-WB, UMTS_AMR-WB, OFR_AMR-WB, OHR_AMR-WB	FR_AMR-WB, UMTS_AMR-WB, OFR_AMR-WB, OHR_AMR-WB
C1 - C4 (0.0.0.1) (0.0.1.1) (0.1.0.0) (0.1.0.1) (0.1.1.0) 1.0.0.1 1.0.1.0 (1.0.1.1) 1.1.0.0 0.0.1.0 (1.1.0.1) C1 - C4 1.0.1.1 1.0.1.0	Frame_Type / Codec Type (GSM_FR) (FR_AMR) (HR_AMR) (UMTS_AMR) (UMTS_AMR_2) FR_AMR-WB UMTS_AMR-WB (OHR_AMR) OFR_AMR-WB OHR_AMR-WB (GSM_EFR) Frame_Type / Codec Type FR_AMR_WB UMTS_AMR_WB	The coding is different from the coding in TFO Messages. It is also not identical to the coding on Abis/Ater. The TRAU shall translate the coding between TRAU and TFO Frames. Codec Types in (brackets) are not supported by this TFO Frame format. They are listed to show their coding for convenience.
C5a 0 1	EMBED No TFO Message embedded A TFO Message is embedded	Indicates the presence of an embedded TFO Message. Set by the TRAU.
C5b	Set to 1 (spare)	
C6 - C8	Set to "1.1.1"(see note)	In GSM TRAU Frames, these bits carry part of the Time Alignment. They are set to 1.1.1 by the TRAU.
C9 - C11 0.0.0 0.0.1 0.1.0 0.1.1 1.0.0 1.0.1 1.1.0 1.1.1	TFO and Handover_Notifications TFO_On TFO_Soon TFO_Off Handover_Soon Handover_Complete undefined undefined undefined	In GSM TRAU Frames these bits are part of the Time Alignment field. These bits are copied from TRAU frames to TFO Frames and vice versa. TFO_On is the default value in TFO Frames.
C12	RIF (Request or Indication Flag)	Copied from the uplink TRAU Frame in GSM Generated by the Transcoder in 3G systems for FR_AMR and HR_AMR: The changes of the uplink Codec Mode, as received via the Iu Frames, are monitored. Whenever the Codec Mode changes, the RIF bit is set to "0". The next frames are then alternatingly marked with RIF = "1", "0", "1" and so on.
C13	set to 1	Copied from the uplink TRAU Frame in GSM. Generated by the TC in UMTS.
C14 – C16	Config_Prot	Coding defined in Annex C.
C17 C18	Mess No	Coding defined in Annex C.
C19	DTXd (see note)	Copied from uplink TRAU Frame in GSM
C20 0 1	TFOE TFO Disable TFO Enable	Copied from the uplink TRAU Frame in GSM Generated by the Transcoder in 3G systems with the same coding as in the 3GPP TS 48.060
C21 – C22 1 1 1 0 0 1 0 0	Frame_Classification "Speech_Good" "Speech_Degraded" "Speech_Bad" "No_Speech"	Copied from the uplink TRAU Frame in GSM Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.4.1-4 below)
C23a – C25a and C23b- C25b	(see 3GPP TS 48.060) CMI (if RIF == 0) or CMR (if RIF == 1) or 0.0.0 (if Frame_Classification == 0.0)	Carry CMI or CMR depending of the value of RIF, if the Frame Classification bits are different from "0 0" (No_Speech), and set to "000" otherwise. Copied from the uplink TRAU Frame in GSM Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.4.1-4 below)

Control Bits	Description	Comment
T1 - T4	Time Alignment Bits	In GSM copied from the uplink TRAU Frame In 3G, generated by the TC (UMTS) based on Iu Frame arrival time(s)

NOTE: Any spare control bits shall be coded as binary "1". They are reserved for future use and may change.

The CRC1 covering also the control bits C1…C25 shall be recomputed in the transcoders.

The coding of the **Data Bits** is described in 3GPP TS 48.060.

In 3G systems, the Frame_Classification Bits must be derived from the Frame Quality Indicator (FQI) and Frame Type Index as defined in the 3GPP TS 26.101. Table 5.4.1-4 provides the conversion rules between the generic WB-AMR Frames (as defined in 3GPP TS 26.101) and TFO Frames. In this table, the arrows in the fourth column indicate the direction for which the conversion applies.

NOTE 1: A one-to-one relationship between Generic WB-AMR Frames and TFO Frames does not always exist, but the conversion is always possible.

NOTE 2: In the generic WB-AMR Frames (3GPP TS 26.101), the differentiation between SID_FIRST and SID_UPDATE is done in the Data bits (SID Type Indicator). The Codec Mode Indication (CMI) is carried in 3G systems within the SID payload.

For 2G systems using the FR_AMR-WB, UMTS_AMR-WB, OFR_AMR-WB or OHR_AMR-WB and 3G systems using UMTS_AMR-WB, OFR_AMR-WB or OHR_AMR-WB, bits C23a – C25a and C23b-C25b shall carry either the Codec Mode Request (CMR) or the Codec Mode Indication (CMI), depending on the value of RIF, if the Frame_Classification bits are different from "0.0". If the Frame_Classification bits are equal to "0.0" (NO_SPEECH Frames), C23a – C25a and C23b – C25b are set to 0.0, and the CMI and CMR are carried in the data bits D35 – D40.

3G systems using the UMTS_AMR-WB, OFR_AMR-WB or OHR_AMR-WB Speech Codec Type, the TC shall monitor the changes of the uplink Codec Mode, as received in the Iu Frames. Every time the Codec Mode changes in the Iu Frames the TC shall set RIF = "0" in the corresponding TFO Frame. The next TFO Frames are alternatively marked with RIF = "1", "0", "1" and so on.

NOTE 3: Per definition for FR_AMR-WB the UE selects the phase of potential Codec Mode changes in uplink once at call set-up and does not alter this later on. At call set-up TFO is not active and the TC has enough time to find the phase of the RIF by the proposed implicit method, before the first TFO Frame has to be sent.

Table 5.4.1-4: Conversion between Generic AMR Frames and FR_WB_AMR_TFO_32k Frames

Generic AMR Frame				AMR_WB_TFO_32k Frame			
Frame Quality Indicator	Frame Type Index	TX_TYPE or RX_TYPE (see 3GPP TS 26.101)		Frame_ Classification C21 – C22	CMI or CMR C23b – C25b C23a – C25a	Data bits in No_Speech frames D32 … D34	Equivalent Frame Type in 3GPP TS 48.060)
1	0-9	SPEECH_GOOD	<->	1 1	0 4 0 7	-	Speech_Good
1	0-9	SPEECH_GOOD	<	1 0	0 4 0 7	-	Speech_Degraded
0	0-9	SPEECH_BAD	<->	0 1	0 4 0 7	-	Speech_Bad
1	8	SID_FIRST	<->	0 0	0 0 0 0 0 0	SID_First	No_Speech
1	15	NO_DATA	<	0 0	0 0 0 0 0 0	Onset	No_Speech
1	8	SID_UPDATE	<->	0 0	0 0 0 0 0 0	SID_Update	No_Speech
0	8	SID_BAD	<->	0 0	0 0 0 0 0 0	SID_Bad	No_Speech
1	15	NO_DATA	<->	0 0	0 0 0 0 0 0	No_Data	No_Speech

The **Synchronisation Pattern** is similar to the Synchronisation Pattern in 3GPP TS 48.060, with some exceptions related to the value of the EMBED Bit:

EMBED equal "0": the Synchronisation Pattern is exactly as described in the 3GPP TS 48.060;
EMBED equal "1": the Synchronisation Pattern contains an embedded TFO Message.

For the coding of the **Data Bits** see 3GPP TS 48.060 and Annex C for the bits reserved for TFO Configuration Parameters.

For the coding of the **Time Alignment Bits** (T_Bits, T1... T4) see 3GPP TS 48.060 and Annex C. When the TFO Frame is generated by a GSM Network, the T_Bits normally correspond to the T_Bits received in the up-link TRAU Frame.

5.4.2 Transmission of the bits of 32 kbit/s TFO Frames

For the purpose of this description the 640 bits of one TFO Frame are arranged in 2 x 40 rows (0a..39a, 0b…39b), with 8 bit each (1..8: one octet) as in 3GPP TS 48.060.

The bits of 32 kbit/s TFO Frames are transmitted in the following order:

Bit m of octet n, shall be transmitted in the **Least** Significant Bit of the

PCM sample k = n*4 + (m+1)/2 for m = (1, 3, 5, 7) and n = (0a...39a).

Bit m of octet n shall be transmitted in the **second Least** Significant Bit of the

PCM sample k = n*4 + m/2 for m = (2, 4, 6, 8) and n = (0a..39a).

Bit m of octet n, shall be transmitted in the **third Least** Significant Bit of the

PCM sample k = n*4 + (m+1)/2 for m = (1, 3, 5, 7) and n = (0b...39b).

Bit m of octet n shall be transmitted in the **forth Least** Significant Bit of the

PCM sample k = n*4 + m/2 for m = (2, 4, 6, 8) and n = (0b..39b).

~~PCM sample (k=1) is the first PCM sample of the TFO Frame, which follows the received uplink TRAU frame with a small delay (Tultfo), as described in clause 8, see figure 8.1.2-1.~~

~~It is important that the lower main 16k frame and the upper extension frame are exactly synchronised as described above.~~

~~Editor´s note: if decided so by the TFO subgroup and approved by SA4 the unused bits in the upper 16k extension shall not modify the PCM sample bits to minimise the acoustic influence of TFO Frames. Then the upper extension frame does not need own synchronisation or control bits. Only data bits are necessary.~~

5.5 Determination of the TFO Frame format

The TFO Frame format is depending on the Codec Types at both ends of the TFO connection.

For the GSM FR and GSM EFR Speech Codec Types, the TFO Frame format shall be 16 kbit/s (see clause 5.2.1).

For the GSM HR Speech Codec Type, the TFO Frame format shall be 8 kbit/s (see clause 5.3.1).

For any TFO connection with at least one side using the HR_AMR (HR_AMR–HR_AMR, HR_AMR–FR_AMR, HR_AMR-UMTS_AMR_2, HR_AMR-OHR_AMR) the TFO frame format shall be AMR_TFO_8+8k (see clause 5.2.2.2).

For the AMR TFO connections involving OHR_AMR-OHR_AMR, UMTS_AMR-UMTS_AMR, UMTS_AMR_2-UMTS_AMR_2 and UMTS_AMR_2-FR_AMR-FR_AMR the TFO Frame format shall be AMR_TFO_16k (see clause 5.2.2.1).

For any AMR-WB TFO connection not supporting codec modes higher than ~~14,25~~12.65 kbit/s, the TFO frame format shall be AMR_WB_TFO_16k ~~(see 5.2.2.3)~~.

For all other AMR WB TFO connections, the TFO frame format shall be AMR_WB_TFO_32k ~~(see 5.4)~~.

Nex modification

8.1.2 Time Alignment of TFO Frames to Uplink TRAU Frames

The contents of the Uplink TRAU Frame, received from the BTS via the Abis/Ater Interface, undergo the small, constant delay (Tultfo) required to perform the modifications of the EMBED, Sync and potentially CRC bits, before being forwarded to the other TRAU over the A Interface as TFO Frame. Since this delay is substantially smaller than the delay for the decoded speech signal, the TFO Frames precede the corresponding speech samples. Figure 8.1.2-1 shows the relations. Note that no exact delay value for Tultfo is defined or need to be defined.

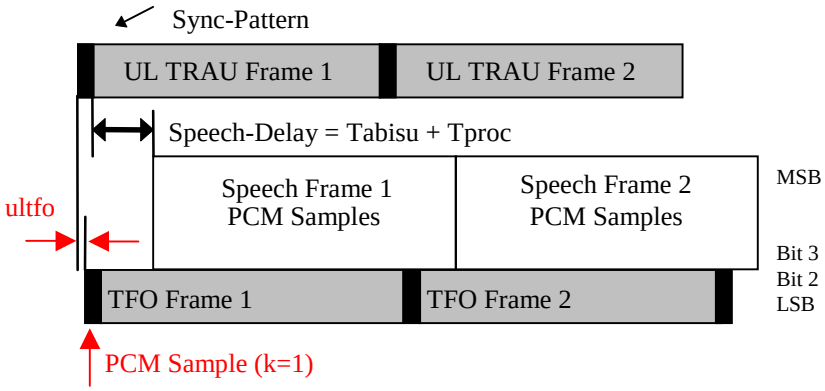


Figure 8.1.2-1: Uplink TFO Frame Time Alignment in GSM

In case of AMR-WB with Codec Modes higher than 12.65 kbit/s the lower main 16k part defines synchronisation and control bits, while the upper 16k extension carries only data bits. It is important that these lower and upper part are exactly synchronised to each other on the A-interface. If this is not already the case on the uplink Abis/Ater interface, then the TRAU shall delay the earlier arriving part to achieve the synchronisation.

3GPP TSG-SA4 #21
Rennes, France, 13-17 May 2002

S4-020364

CR-Form-v6.1

CHANGE REQUEST

⌘	TS 28.062	CR	021	⌘ rev	3	⌘	Current version:	5.0.0	⌘
	Spec Title:	Inband Tandem Free Operation (TFO) of speech codecs							⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network [X] Core Network ☐

Title:	⌘	**Corrections to TS 28.062, Annex C**
Source:	⌘	TSG SA WG4
Work item code:	⌘	AMRWB
Date:	⌘	2002-06-11
Category:	⌘	**F**
Release:	⌘	**REL-5**

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
REL-4	*(Release 4)*
REL-5	*(Release 5)*

Reason for change:	⌘	Inconsistencies
Summary of change:	⌘	Smaller details, but more than editorial
Consequences if not approved:	⌘	Spec. is wrong or potentially misleading

Clauses affected:	⌘	Annex C		
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ O&M Specifications	⌘	
Other comments:	⌘			

Annex C (normative): Tandem Free Operation in GSM, including AMR-WB

C.1 Scope

Annex C describes the mandatory and optional actions within the BSS in GSM for Tandem Free Operation.

C.2 Overview

TFO in GSM implies that the different entities of the BSS collaborate. This is achieved by the distribution of TFO processes on these entities. Figure C.2-1 provides an overview of the TFO processes inside the BSS. This figure shows also the interfaces between these TFO processes.

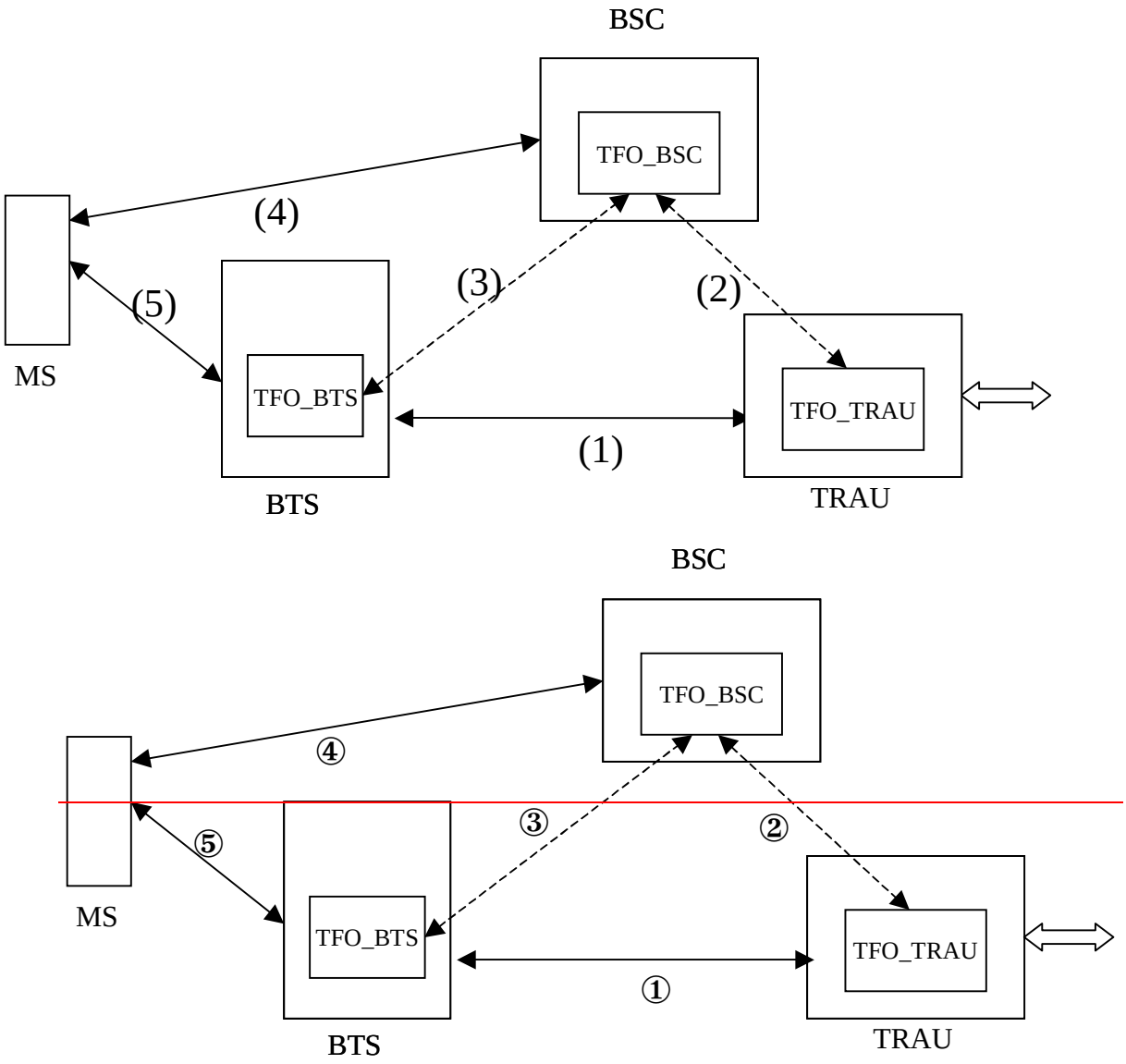


Figure C.2-1: Processes and Interfaces for TFO in GSM

The interfaces as shown in Figure C.2-1 are:

(1) The Abis/Ater Interface (traffic): Only for the AMR or AMR-WB speech Codec Types the Abis/Ater interface is influenced by the TFO. In this case TFO information is exchanged in Config frames and Time Alignment and Rate Control is influenced.

(2) An optional proprietary interface between the BSC and the TRAU; may be used for non-AMR and AMR and AMR-WB Speech Codec Types (FR_AMR, HR_AMR, GSM_FR, GSM_EFR, OHR_AMR, GSM_HR, FR_AMR-WB, UMTS_AMR-WB, OFR_AMR-WB, OHR_AMR-WB) to exchange messages on the distant and local codec configurations, or the optimal configuration.

(3) Layer 3 signalling between the BSC and the BTS.

(4) Layer 3 signalling between the BSC and the MS to modify a Codec Type or a Codec Configuration.

(5) Air interface (RATSCCH, see 3GPP TS 45.009 [9]) to change the Codec Mode Indication phase in downlink or the Codec Configuration in case of AMR TFO.

~~① The Abis/Ater Interface (traffic): Only for the AMR or AMR-WB speech Codec Types the Abis/Ater interface is influenced by the TFO. In this case TFO information is exchanged in Config frames and Time Alignment and Rate Control is influenced.~~

~~② An optional proprietary interface between the BSC and the TRAU;, may be used for non-AMR and AMR and AMR-WB Speech Codec Types (FR_AMR, HR_AMR, GSM_FR, GSM_EFR, OHR_AMR, GSM_HR, FR_AMR-WB, UMTS_AMR-WB, OFR_AMR-WB, OHR_AMR-WB) to exchange messages on the distant and local codec configurations, or the optimal configuration.~~

~~③ Layer 3 signalling between the BSC and the BTS.~~

~~④ Layer 3 signalling between the BSC and the MS to modify a Codec Type or a Codec Configuration .~~

~~⑤ Air interface (RATSCCH, see 3GPP TS 45.009 [9]) to change the Codec Mode Indication phase in downlink or the codec configuration in case of AMR TFO.~~

Next Modification

C.2.4 Modifications of the Codec Type and/or the Codec Configuration

The following clauses provide a brief overview over all possible versions (not to be mixed up with “AMR TFO Version” or “TFO Version”). They differ in the Node where the TFO Decision is performed and the Node that executes the decided change. The following table provides an overview:

Next Modification

C.3.2 Tx_TRAU Process

The Tx_TRAU Process builds autonomously the relevant Downlink TRAU Frames and sends them in the correct phase relation onto the Abis/Ater-Interface as commanded by the time alignment from the BTS.

Tx_TRAU has two major States: TFOdl == OFF (start-up default state) and TFOdl == ON (see Figure C.3.2-1).

TFO_Protocol Protocol controls the transitions between these states using the Accept_TFO (AT) and Ignore_TFO (IT) commands.

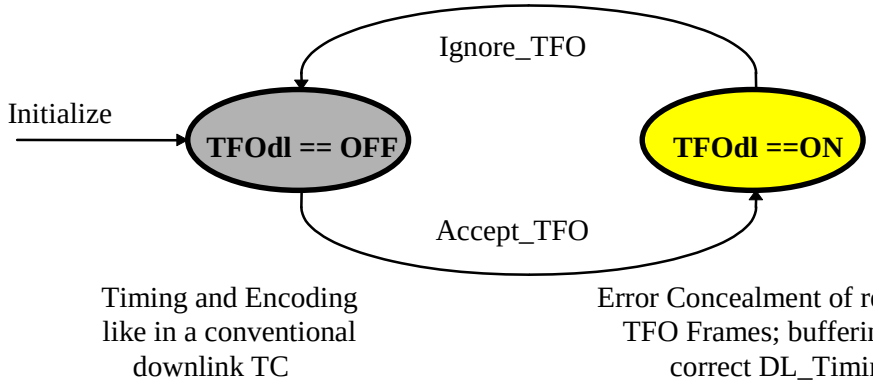


Figure C.3.2-1: States of the Tx_TRAU Process

During TFOdl == OFF Tx_TRAU performs all actions of a conventional downlink TRAU (see 3GPP TS 48.060 [3] respectively 3GPP TS 48.061 [4]): On command from Rx_TRAU it performs necessary downlink time alignments and starts or stops sending TRAU Frames. It samples one frame of speech samples in the correct phase position and calls the Speech Encoder. ~~In case of AMR-WB a successive downsampling is performed before G.711 encoding.~~ In case of AMR-WB the received PCM samples are decoded and upsampled before the WB speech encoder is called. The resulting speech parameters are then transmitted downlink on the Abis/Ater interface.

Next Modification

Table C.5.2.2-2: Optimal or Distant Configuration (OD)

OD Code	Optimal or Distant Configuration	Comment
0	Distant	TRAU shall send Distant Configuration Parameters
1	Optimal	TRAU shall send Optimal Configuration Parameters

In case of OM = Change, the TRAU provides the BTS and further on the BSC (see 3GPP TS 48.058 clause 4.15) with the Distant Configuration (OD = Distant) or the Optimal Configuration (OD = Optimal).
OD is a configuration parameter set by the BTS (respectively the BSC) and send to the local TRAU.

Next Modification

C.5.2.3 Handovers and the AMR TFO

Handover in an ongoing AMR-TFO connection needs more attention. It can be handled more efficiently, if the BSC takes the configurations (the active local one in the serving, old BTS, the future local one in the new BTS and the distant one in the distant BTS) into account and informs the serving BTS a before performing the handover ("Pre-Handover Notification", see clause C.4.6). The sending of the Pre-Handover Notification should take into account the round-trip delay if it has been reported by the BTS (see clause C.4.5).

The BSC, as a central point of the BSS, manages the AMR Speech Service configuration along the communication. ~~This is done in such a way that the point ③ of the list provided above can be achieved.~~

The BSC has at any time control over the ongoing call, especially over all used resources. Some AMR specific adaptation procedures are, however, handled by lower layer inband signalling directly, e.g. time alignment, CMI/CMC phase alignment and Codec_Mode adaptation (Rate Control).

Next Modification

AMR TFO Version Number (ATVN: 1 bit)
The current AMR TFO Version Number is 0~~1~~.

3GPP TSG-SA4 Meeting #21
Rennes, France, 13-17 May 2002

Tdoc S4-020357

CR-Form-v6.1

CHANGE REQUEST

⌘	TS 28.062	CR	022	⌘ rev	2	⌘	Current version:	5.0.0	⌘

Spec Title: Inband Tandem Free Operation (TFO) of speech codecs ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network X Core Network X

Title:	⌘	**TFO Version Handling**
Source:	⌘	TSG SA WG4
Work item code:	⌘	AMRWB
Date:	⌘	11-June-2002
Category:	⌘	**F**
Release:	⌘	**REL-5**

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
REL-4	*(Release 4)*
REL-5	*(Release 5)*

Reason for change:	⌘	TFO Version Handling is not defined in REL 5.0.0
Summary of change:	⌘	Define TFO Version Handling
Consequences if not approved:	⌘	Incomplete specification, potential source for misbehaviour

Clauses affected:	⌘	4			
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ O&M Specifications	⌘	none	
Other comments:	⌘				

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

4.3 ~~AMR~~ TFO Standard Version Handling

~~The present document applies to the version 1 of the AMR TFO standard.~~

In TFO Specifications before REL-4 no TFO version handling is defined.

In TFO Specifications of REL-4 an "AMR TFO version number" is defined in the Ver (Version number) field of the AMR_ACS and AMR_SCS Extension Blocks (see clause 7) and the ATVN field in AMR Configuration frames (see Annex C). Only one REL-4 AMR TFO version is defined: version "0".

From REL-5 onwards the "TFO Version number" contained in the "TFO Version" extension block (section 7) and in "Generic Configuration Frames" (Annex H) shall reflect the Version and Subversion of the corresponding TS 28.062 (first and second digit of the TS version number, see foreword). The AMR TFO version number (Ver, AVTN, as in REL-4) shall be treated as "undefined" in case the TFO Version Number (as in REL-5 and onwards) is indicated in the TFO Messages.

~~This version~~The current TFO Version supports the GSM_FR, GSM_HR, GSM_EFR-, five AMR (Narrow Band) speech codec types (FR_AMR, HR_AMR, UMTS_AMR, UMTS_AMR_2, OHR_AMR: AMR-NB family) and four AMR Wide Band speech codec types (FR_AMR-WB, UMTS_AMR-WB, OFR_AMR-WB, OHR_AMR-WB: AMR-WB family).

~~The version number is only indicated in the Ver (Version number) field of the AMR_ACS and AMR_SCS Extension Blocks (see clause 7) and the ATVN field in Configuration frames (see annex C) and the AMR-WB_ACS Extension Block.~~

~~When no version number is indicated in the TFO Messages, version 0 applies.~~

The smallest defined TFO Version number is 0~~5~~.0. It stands for all TFO Versions before 5.1. All numbers between 0.0 and 5.1 are reserved for future use. If the Local and Distant version numbers differ, the smallest version number shall have precedence and shall be applied on both sides. The following features (table 4.3-1) are optional or ~~and~~ mandatory for the different Codec Types, depending on the applicable version number:

Table 4.3-1: TFO Version Handling

Feature→ Codec Type↓	**TFO Version**	**Immediate Codec Type Optimisation**	**Generic Configuration Frames**
GSM_FR GSM_HR GSM_EFR	**Optional.** The TFO Version extension block need not to be sent. If not contained in TFO Messages, or is lower than 5.1, then Pre-REL-5 handling shall apply	**Mandatory,** if TFO Version is 5.1 or higher.~~, if possible~~	If the TFO Version is lower than 5~~.~~.1 then Generic Configuration Frames shall not be used. Only TFO_REQ_L and (TFO_ACK_L) shall be used. If the TFO Version is 5.1 or higher, then Generic Configuration Frames shall be used. TFO_REQ_L and TFO_ACK_L shall not be used embedded into TFO Frames.
FR_AMR HR_AMR UMTS_AMR UMTS_AMR2 OHR_AMR	**Optional.** The TFO Version extension block need not to be sent. If not contained in TFO Messages, or is lower than 5.1, then Pre-REL-5 handling shall apply	**Mandatory,** if TFO Version is 5.1 or higher.~~, if possible~~	If the TFO Version is lower than 5.1, then Generic ConfigurationFrames shall not be used. It is optional to use either TFO_REQ_L or AMR Configuration frames, see Annex C.8.5. If the TFO Version is 5.1 or higher, then Generic Configuration Frames shall be used. The parameter field in REL-4 AMR Configuration frames shall be treated as undefined. TFO_REQ_L and TFO_ACK_L shall not be used embedded into TFO Frames.
FR_AMR-WB UMTS_AMR-	**Mandatory.** The TFO Version extension	**Mandatory.~~,~~**	Generic Configuration Frames shall be used. TFO_REQ_L and TFO_ACK_L shall not be

WB OFR_AMR-WB OHR_AMR-WB	block shall always be sent.	~~if possible~~	used embedded into TFO Frames.

3GPP TSG-SA4 #21
Rennes, France, 13-17 May 2002

S4-020358

CR-Form-v6.1

CHANGE REQUEST

⌘ **TS 28.062** **CR** **023** ⌘ **rev** **2** ⌘ Current version: **5.0.0** ⌘

Spec Title: Inband Tandem Free Operation (TFO) of speech codecs ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network ☒ Core Network ☐

Title:	⌘	**Configuration Exchange in Annex C**
Source:	⌘	**TSG SA WG4**
Work item code:	⌘	AMRWB
Date:	⌘	2002-06-11
Category:	⌘	**F**
Release:	⌘	**REL-5**

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
REL-4	*(Release 4)*
REL-5	*(Release 5)*

Reason for change:	⌘	Inconsistencies between REL-4 and REL-5 and not complete REL-5 specification
Summary of change:	⌘	Specification how to handle Configuration Exchange
Consequences if not approved:	⌘	Spec. is wrong or potentially misleading

Clauses affected:	⌘	Annex C
Other specs affected:	⌘	☐ Other core specifications ⌘ ☐ Test specifications ☐ O&M Specifications
Other comments:	⌘	See also S4-020281 TFO Version Handling

Annex C (normative): Tandem Free Operation in GSM, including AMR-WB

first modification in C.6

C.6 The Dialogue between TFO_TRAU and TFO_BTS

From REL-5 onwards the "Generic Configuration Frame" is defined (Annex H) as a mechanism to exchange Configuration parameters between BTS and TRAU, between the TRAUs and between local and distant BTSs. These generic configuration frames are codec-type-independent and may in principle be used also for older Codec Types. Clause 4.3 defines when to use this generic configuration frame and when to use the AMR Configuration frame or the TFO_REQ_L / TFO_ACK_L mechanism.

The BTS ~~is~~ need not to be involved in TFO when GSM_FR, GSM_EFR or GSM_HR Speech Codec Types are used. But from REL-5 onwards the generic configuration frames may also be used for these Codec Types. Then the BTS shall be able to handle them, at least to ignore them, when they appear in downlink on the Abis/Ater interface. If this is not possible, then the TRAU shall not use it either.

The following clauses address therefore mainly the dialog between the BTS and TRAU or between the Local and Distant TRAUs or BTSs in case of ~~FR_AMR and HR_AMR and FR_AMR-WB.~~the AMR-NB and AMR-WB families of Codec Types.

C.6.1 Configuration Parameters in AMR-NB TRAU/TFO frames

C.6.1.1 Configuration Protocol Format

"TRAU AMR Configuration frames" and "TFO AMR Configuration frames" contain AMR-NB and TFO configuration parameters. The "generic configuration frames" contain configuration parameters for all codec types. These parameters are exchanged by the following configuration protocol between several entities (local BTS to local TRAU, local BTS to distant BTS, local TRAU to distant BTS and local TRAU to local BTS).

Three control fields are defined for the TFO and TRAU AMR Configuration frames and in generic configuration frames:

- Config_Prot field defines the sender and the recipient;
- Message_No field is a protocol counter;
- Par_Type field defines the contents of the parameter fields.

The Parameter fields carry the TFO and AMR Configuration parameters.

Each TFO (or TRAU) AMR configuration frame contains a set or a subset of these configuration parameters. Some exceptions exist (12,2 kbit/s for instance, see mapping of Configuration Parameters clause C.6.1.5).
Generic configuration frames do always contain a full set, see Annex H..

C.6.1.2 Config_Prot field

This field serves for the Configuration Protocol on the Abis/Ater interface and the A interface in both directions to indicate the source and meaning of the configuration parameters. It is defined in UL TRAU frames, in DL TRAU frames and in TFO frames, both for the AMR Configuration Frames and the Generic Configuration Frames.

Table C.6.1.2-1: Coding of Config_Prot

Config_Prot	Name	Exists on	Meaning	sent by	recipient
0.0.0	No_Con	UL, DL, TFO frame	No configuration included, shall not be acknowledged		
0.0.1	Con_Req	UL, DL, TFO frame	configuration included, shall be acknowledged	L_BTS	D_BTS, L_TRAU
0.1.0	Dis_Req	DL	(subset of) configuration shall be acknowledged	L_TRAU	L_BTS
0.1.1	Con_Ack	UL, DL, TFO frame	acknowledge for Con_Req	L_BTS, D_BTS	D_BTS, L_BTS
1.0.0	Spare	-	for future use		
1.0.1	UL_Ack	UL	acknowledge for Dis_Req	L_BTS	L_TRAU
1.1.0	DL_Ack	DL	acknowledge for Con_Req	L_TRAU	L_BTS
1.1.1	Spare	-	for future use		

Notation: L_TRAU: local TRAU, L_BTS: local BTS, D_BTS: distant BTS.

For the mapping of these bits on TRAU/TFO frames, see clause C.6.1.5 for AMR Configuration frames and Annex H for generic configuration frames.

For the use of the Config_Prot, see clause C.8.

C.6.1.3 Message_No Field

The Message_No is used to mark a configuration request message at sender side in order to bind the acknowledgement from the receiver side. It is two bits long. For the mapping of these bits ~~on TRAU/TFO frames,~~ see clause C.6.1.5 and Annex H.

C.6.1.4 Configuration Parameters Fields

The configuration parameters are:

TFOE (1 bit)
TFOE (TFO_Enable) set to 0: TFO disabled; set to 1: TFO enabled.
By this bit set to 1 the BTS enables the TRAU to perform TFO negotiation and to go into Tandem Free Operation, if possible. Respectively, if this bit is set to 0, the TRAU shall terminate TFO as soon as possible and shall not initiate or respond to any TFO negotiation message.

TFOE in AMR Configuration frames or generic configuration frames is also used to signal to the distant TFO partner that TFO is terminated (see Annex G.3).

Time Alignment Field (6 bits)
The Time Alignment Field is defined in 3GPP TS 48.060 [3] for time and phase alignment.
In addition five more code points, which are reserved in3GPP TS 48.060 [3] are defined for TFO and Handover Notifications:

Time Alignment Field	Name	defined on
1.1.1.**0.0.0**	TFO_On	Abis/Ater
1.1.1.**0.0.1**	TFO_Soon	Abis/Ater
1.1.1.**0.1.0**	TFO_Off	Abis/Ater
1.1.1.**0.1.1**	Handover_Soon	Abis/Ater and A
1.1.1.**1.0.0**	Handover_Complete	Abis/Ater and A

The protocol for the exchange of these Notifications is defined in Annex C.6.2.

Par_Type (2 bits)
Par_Type defines the meaning of the Configuration Parameters. It is set by the sender of the configuration frame.
MSB.LSB:

0.0	Configuration Parameters not valid
0.1	local Configuration Parameters
1.0	distant Configuration Parameters
1.1	optimal Configuration Parameters

Codec_List (13 bits)
The supported Codec Types are coded as defined in 3GPP TS 26.103, clause "Codec Bitmap", bit 1 to bit 13. Bit 13 is defined to be the MSB of the Codec List field. For the mapping of these bits on TRAU/TFO frames, see clause C.6.1.5 for AMR Configuration frames. This field is not present in generic configuration frames.

Sys_ID (4 bits)
The Sys_ID codes the System_Identification of the sending side, see table Annex A.5-1. Only the four LSBs are used here (short form) in AMR Configuration frames. The four MSBs are assumed to be "0". In generic configuration frames this parameter is coded with 8 bits.

Active_Codec_Type (ACT: 4 bits)
The Active_Codec_Type identifies the Codec_Type actually used. The coding is according to 3GPP TS 26.103, table 6.3-1. The lower four bits are used here in AMR configuration frames (short form). The long form is used in generic configuration frames.

Active_Codec_Set (ACS: 8 ~~or 9~~ bits see 3GPP TS 45.009 [9]):
The ACS is defined, if the Active_Codec_Type is ~~FR_AMR, HR_AMR or FR_AMR_WB)~~either from the AMR-NB or the AMR-WB family. The coding is according to 3GPP TS 26.103.

Supported_Codec_Set (SCS: 8 or 9 bits; see 3GPP TS 45.009 [9]):
The SCS is defined, if the Active_Codec_Type is either from the AMR-NB or the AMR-WB family.~~FR_AMR, HR_AMR of FR_AMR_WB~~. The coding is according to 3GPP TS 26.103.~~.~~

Maximum Number of Modes in the ACS (MACS: 3 ~~or 4~~ bits)
The MACS is defined, if the Active_Codec_Type is either from the AMR-NB or the AMR-WB family.~~FR_AMR, HR_AMR of FR_AMR_WB.~~ The coding is according to 3GPP TS 26.103.

AMR TFO Version Number (ATVN: 1 bit)
The current AMR TFO Version Number is ~~1~~0.

Optimisation Mode (OM: 1 bit)
The Optimisation Mode is defined, if the Active_Codec_Type is either from the AMR-NB or the AMR-WB family. The coding is according to 3GPP TS 26.103.

~~The Optimisation Mode is defined, if the Active_Codec_Type is AMR. The coding is according to 3GPP TS 26.103.~~

Optimal or Distant Configuration (OD: 1 bit)
The "Optimal or Distant Configuration" parameter is described in clause C.5.2.2.

CRC_A: 3-bit CRC (see clause 7.3).

CRC_B: 3-bit CRC (see clause 7.3).

CRC_C: 3-bit CRC (see clause 7.3).

C.6.1.5 Mapping of the Configuration Parameters on 16 and 8 kbit/s TRAU/TFO frames for AMR Configuration

AMR Configuration frames are defined for REL-4 and REL-5. In case generic configuration frames shall be used (see clause 4.3) the AMR Configuration bits in TFO/TRAU Speech and No_Speech Frames shall be set to -spare = "1".

Table C.6.1.5-1 gives the mapping of the AMR configuration fields for each frame (TRAU/TFO) format:

Table C.6.1.5-1: Mapping of the configuration parameters in the TRAU/TFO frames

Sub-multiplexing		**8 kbit/s**	**8 kbit/s**	**8 kbit/s**		**16 kbit/s**	**16 kbit/s**	**16 kbit/s**
Codec Modes	**#bits**	**No_Data**	**SID**	**Speech ≤5,9 kbit/s**		**No_Speech**	**Speech ≤7,95 kbit/s**	**Speech 10,2kbit/s**
Time Align. Field	6	D1..D6	D1..D6	# (= TFO_On)		C6..C11	C6..C11	C6..C11
Config_Prot	3	D55..D57	D55..D57	D55..D57		C14..C16	C14..C16	C14..C16
Message_No	2	D58..D59	D58..D59	D58..D59		C17..C18	C17..C18	C17..C18
TFO_Enable	1	D64	D64	# (= 1)		C20	C20	C20
Par_Type(5)	2	D65..D66	D65..D66	# (= 0.0)		D1..D2	D1..D2	D1..D2
OD	1	D67	D67	#		D3	D3	D3
OM(3)	1	D68	D68	#		D4	D4	D4
ACS(3) **(Optimal ACS)**(5)	8	D69..D76	D69..D76	#		D5..D12	D5..D12	D5..D12
SCS(3)	8	D77..D84	D77..D84	#		D13..D20	D13..D20	D13..D20
ATVN(3)**, short**(6)	1	D85	D85	#		D21	D21	# (= 0)
Sys_ID, short(6)	4	D86..D89	D86..D89	#		D22..D25	D22..D25	# (= 0..0)
spare (= 0)	3	D90..D92	D90..D92	#		D26..D28	D26..D28	# (= 0)
CRC_A (of 28 bits:)	3	D93..D95 (D65..92)	D93..D95 (D65..92)	#		D29..D31 (D1..D28)	D29..D31 (D1..D28)	#(1)
ACT(3) **(Optimal ACT)**(5)	4	D96..D99	D96..D99	#		D234..D237	D234..D237	D234..D237
MACS(3)	3	D100..D102	D100..D102	#		D238..D240	D238..D240	D238..D240
Codec List	13	D103..D115	D103..D115	#		D241..D253	D241..D253	D241..D253
CRC_B (of 20 bits:)	3	D116..D118 (D96..115)	D116..D118 (D96..115)	#		D254..D256 (D234..253)	D254..D256 (D234..253)	#(2)
SCS_2(4)	8	D17..D24	# (= 1..1) (7)	#		D203..D210	D203..D210	# (= 1..1) (7)
OM_2(4)	1	D25	# (= 0)	#		D211	D211	# (= 0)
MACS_2(4)	3	D26..D28	# (= 1.0.0)	#		D212..D214	D212..D214	# (= 1.0.0)
ATVN_2(4)(6)	1	D29	# (= 0)	#		D215	D215	# (= 0)
SCS_3(4)	8	D30..D37	# (= 1..1) (7)	#		D216..D223	D216..D223	# (= 1..1) (7)
OM_3(4)	1	D38	# (= 0)	#		D224	D224	# (= 0)
MACS_3(4)	3	D39..D41	# (= 1.0.0)	#		D225..D227	D225..D227	# (= 1.0.0)
ATVN_3(4)(6)	1	D42	# (= 0)	#		D228	D228	# (= 0)
spare (=0)	2	D43..D44	#	#		D229..D230	D229..D230	#
CRC_C (of 28 bits:)	3	D45..D47 (D17..44)	#	#		D231..D233 (D203..230)	D231..D233 (D203..230)	#
8k_spare	7	D48..D54	#	#				
8k_spare	7	D119..D125	D119..D125	#				
16k_spare	14					D44..D57	#	#

The bit positions refer to the positions reserved in 3GPP TS 48.060 [3] and 3GPP TS 48.061 [4] : D bits are data bits, C bits are control bits. The parameters are mapped into the field with MSB first, example:
Par_Type: MSB => D65, LSB => D66 in 8k frames.

# denotes not existing fields; the entries in brackets () denote the default values of the missing parameters, see Note(7). Only if the missing parameters are set to these default values, these frames may be used. Otherwise No_Data frames shall be used.

NOTE 1: In Mode 10,2 the bits D93..D95 are already used for the CRC1 of the first sub-frame. The bits otherwise protected by CRC_A shall be protected in Mode 10,2 by CRC1 (see 3GPP TS 48.060 [3]).

NOTE 2: In Mode 10,2 the bits D254..D256 are already used for the CRC4 of the fourth sub-frame. The bits otherwise protected by CRC_B shall be protected in Mode 10,2 by CRC4 (see 3GPP TS 48.060 [3]).

NOTE 3: The fields ACS, SCS,MACS, OM and ATVN shall always be used for the Active Codec Type, if from the AMR or AMR-WB families.

NOTE 4: The fields SCS_2 ... ATVN_3 are reserved for the other AMR Codec Types, when flagged in the Codec_List, according to the following mapping:

Active Codec Type	ACS, SCS, OM, MACS, ATVN	SCS_2, OM_2, MACS_2, ATVN_2	SCS_3, OM_3, MACS_3, ATVN_3
none of AMR	FR_AMR	HR_AMR	UMTS_AMR(_2)
FR_AMR	FR_AMR	HR_AMR	UMTS_AMR(_2)
HR_AMR	HR_AMR	FR_AMR	UMTS_AMR(_2)
UMTS_AMR(_2) [8]	UMTS_AMR(_2)	FR_AMR	HR_AMR

If a Codec Type is not within the Codec_List, then the corresponding fields are undefined and shall be set to "0".

NOTE 5: If Par_Type is set to "Optimal Configuration", then ACT and ACS shall carry the optimal configuration. All other configuration parameters shall carry the Codec List and the relevant configuration parameters.

NOTE 6: For Sys_ID and ATVN a short form is used: only lower 4 bits for Sys_ID, only LSB for AVTN. The missing bits are defined to be "0".

NOTE 7: The default setting for the SCS fields shall be "1111.1111" for FR_AMR and UMTS_AMR and "0001.1111" for HR_AMR.

NOTE 8: Either UMTS_AMR or UMTS_AMR_2 shall be indicated, but not both together, with preference to UMTS_AMR_2.

Note for the AMR_TFO_8+8k frames: Only the "No_Data" frames convey all configuration parameters. Thus, a speech frame has to be stolen when this configuration information has to be sent. The frames with a rate lower or equal to 5,9 kbit/s can convey only the Config_Prot and Mess_No without stealing a speech frame. Par_Type in these speech frames is assumed to be "0.0".

Note for the AMR_TFO_16k frames: All the configuration parameters are included in the rates below the 10,2 kbit/s. The 12,2 kbit/s conveys TFO enable and the Config_Prot only. Par_Type in 12,2 kbit/s speech frames is assumed to be "0.0". Thus a speech frame has to be stolen to send configuration parameters.

C.6.2 TFO and Handover Status of the Connection

C.6.2.1 TFO Status Messages

The TRAU shall inform the BTS of its TFO status with three TFO Notifications:

- *TFO_Off* TFO is not established.
- *TFO_Soon* TFO is likely to be established.
- *TFO_On* TFO is established and ongoing.

The BTS may inform the TRAU and the distant partner with two Handover Notifications

- *Handover_Soon* Handover is to be expected soon.
- *Handover_Complete* Handover has been performed.

C.6.2.2 Notification of Status of Connection

The Messages "*TFO_Soon*", "*TFO_On*" and "*TFO_Off*" are sent by the Tx_TRAU within the Time Alignment Field.

The BTS shall acknowledge the correct receipt of TFO Notifications by sending the received TFO Notification back to the TRAU. If the TRAU does not get a correct acknowledgement within *N_out_1* frames, then it shall repeat the TFO

Notification. *N_out_1* shall be initialised at resource allocation to [4], but shall be adapted to the round trip delay between TRAU and BTS during the connection.

The Handover Notifications "*Handover_Soon*" and "*Handover_Complete*" are sent by the BTS to the TRAU within the Time Alignment. Field, always embedded in Con_Req() frames. Since Con_Req() frames shall always be acknowledged, no further acknowledgement for the Handover Notifications is required. If the BTS does not get a correct acknowledgement within *N_out_2* frames, then it shall repeat the Handover Notification. *N_out_2* is set to [4]. It should be adapted according to the round-trip delay.

The Time Alignment Field is used for several purposes: TFO Notifications, Handover Notifications, Time Alignment Request and Time Alignment Acknowledgement. The TRAU and BTS may initiate requests independently and uncoordinated. In case of conflicts the following priority shall be obeyed: Time Alignment Message may always be overwritten. Otherwise: Acknowledgements shall always have higher priorities than requests. With other words: an ongoing exchange shall first be terminated before a new one is started.

In case of ongoing TFO all uplink TRAU frames shall be relayed with minimal delay onto the A-interface as TFO frames. Likewise the received TFO frames shall be relayed as TRAU frames down to the BTS. The time alignment field of the TFO frames shall be copied, too.

C.7 The Dialogue between TFO_BTS and TFO_BSC

~~This clause addresses AMR case only.~~

~~The BTS and the BSC exchange messages through Layer 3 signalling. The BTS is also in contact with the TRAU and extracts the information sent by the TRAU in the TRAU Configuration frames. These pieces of information are afterward sent to the BSC. The Layer 3 messages are specified in 3GPP TS 48.058 [12].~~

~~Reciprocally the BTS relays information received from the BSC toward the TRAU within the TRAU Configuration frames.~~

This clause is valid for all Codec Types of the AMR-NB and AMR-WB families. If BTS and TRAU exchange Configuration information, then they shall use the meachnism defined here. From REL-5 onwards the generic configuration frames may also be used for all other codec types.

The BSC and the BTS exchange messages through Layer 3 signalling as specified in 3GPP TS 48.058 [12].

First, the BSC sends local configuration information to the BTS.

The BTS is also in contact with the TRAU and relays information received from the BSC toward the TRAU within the AMR Configuration frames (REL-4) or in generic configuration frames (REL-5~~.1~~ and onwards).

The BTS also extracts the configuration information sent downlink by the TRAU or the distant BTS in the AMR Configuration frames (REL-4) or in generic configuration frames (REL-5~~.1~~ and onwards).

Finally, the BTS relays this received configuration information back to the BSC.

C.7.1 BSC to BTS messages

The BSC at Channel activation informs the BTS of the local codec configuration. It enables or disable TFO too. It can also delegate the ACS modification to the BTS (MultiRate Control by RATSCCH).

The BSC can enable or disable TFO at any moment during a call whether TFO is ongoing or not (TFO MODIFICATION REQUEST).

The BSC informs the BTS of any change of the local configuration, if the Codec Type Mismatch resolution and/or AMR optimization is supported (MultiRate Codec Mode Req).

The BSC should notify to the BTS when an handover procedure is about to be launched (PRE-HANDOVer NOTIFication). It should also notify the BTS is the handover procedure has failed (PRE-HANDOVer NOTIFication).

C.7.2 BTS to BSC messages

The BTS should report to the BSC the status of the TFO, i.e. when TFO starts and stops (TFO REPort).

The BTS should report the Round trip delay it has estimated (Round Trip Delay REPort). It should report it every time a significant change (e.g. 60 ms) is detected in the round trip delay (see clause 8.2.4).

The BTS should report to the BSC the distant codec configuration (REMOTE CODEC CONFiguration REPort). It should also report any modification of this configuration. It should report the optimal TFO configuration, if the Optimal or Distant Configuration (OD) tells so (MultiRate Codec Mode Req).

C.8 Configuration Parameter Exchange on Abis/Ater and A Interfaces for AMR and AMR-WB

The TFO Speech Service Configuration parameters for TFO may be sent from the BSC via the BTS to the TRAU;

The following block diagram is intended for guidance only. If no TFO is ongoing, then the Config_Prot ends always in the (local) TRAU. If TFO is ongoing, then a mirrored (distant) BSS´ exists. Between the local TRAU and the distant TRAU´ an unknown transit network exists, which is transparent for the TFO Messages and the TFO Frames, but may contain devices involved in the TFO connection (e.g. TFO specific Circuit Multiplication Equipments, TCMEs, for cost efficient transmission).

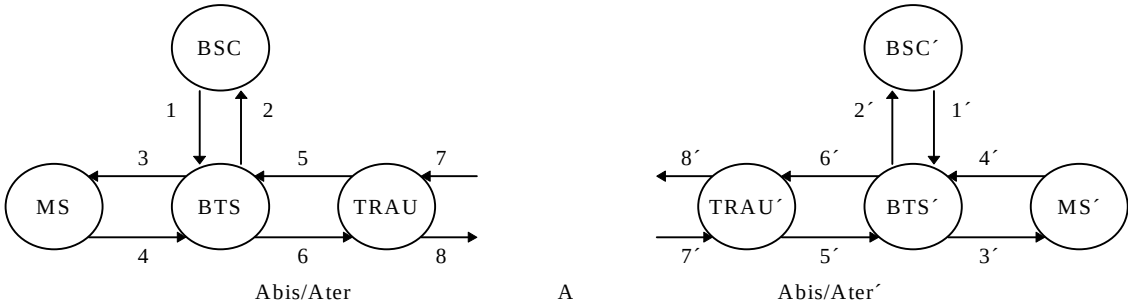


Figure C.8-1: Block diagram of the transmission paths for the exchange of Configuration Parameter

The Configuration parameters received from the BSC (1) shall be sent uplink to the TRAU by inband signalling on the Abis/Ater interface (6). In most Codec_Modes the TRAU speech frames have sufficiently spare capacity to transmit these configuration parameters. Otherwise a No_Speech frame (mainly a No_Data Frame) shall be used, i.e. a speech frame shall be stolen. No_Data Frames are naturally used at call setup or after handover. From REL-5 onwards generic configuration frames ~~shall be~~are used, when both sides support this (see clause 4.3).

C.8.1 Protocol for the Exchange of Configuration Parameters

A simple protocol is defined to ensure correct receipt. It uses the Config_Prot field to code a Request or Acknowledge message and the Message_No field to bind Request and Acknowledgement together. Both are defined in clauses C.6.1.2 and C.6.1.3.

The Par_Type field defines whether a Request or Acknowledgement has defined configuration parameters or not, and which type of parameters are included: None, Local, Distant or Optimal. If a Con_Req has no configuration parameters, then the corresponding Con_Ack shall include the local ones. If Con_Req contains new or modified distant Configuration parameters, then the corresponding Con_Ack shall contain the local configuration parameters. If no configuration is to be exchanged, then the Config_Prot field shall be set to "No_Con". In this case the configuration parameter field is undefined. The receiver shall not acknowledge a No_Con message.

The configuration exchange shall start always with a Request from one side and shall end with an Acknowledgement from the other side. If the Acknowledgement is not received before *N_Out_3* frames are elapsed, then the Request shall be repeated without modifying the Message_No. *N_Out_3* is at resource allocation initialised (e.g. *N_Out_3*:= 4), but shall be adapted to the round trip delay during the connection (see clause C.4.5).

If more than three consecutive repetitions are without success, then TFO shall be terminated and the TFO Protocol shall enter State FAILURE.

The sender of the Request shall always use a new Message_No, e.g. by incrementing a counter, for a new Request. The receiver shall acknowledge by sending the appropriate Acknowledge_Code and the received Message_No back, if the Request was received without detectable errors. Otherwise, in case of detected errors, it shall not acknowledge, but wait for a repetition.

Typically no new request shall be sent before the previous configuration exchange is terminated. Exceptions exist at Resource Allocation, because it is not clear if and when the path between BTS and TRAU is connected through.

C.8.2 Initial Configuration at Resource Allocation

The BTS shall send "Con_Req" Messages. Typically at resource allocation no speech is received from the air interface or at least some FACCH arrive. Therefore "No_Data" frames may be used. Generic configuration frames ~~shall be~~are used from REL-5 onwards. The local TRAU shall acknowledge with "DL_Ack".

As long as No_Speech frames are sent in uplink direction the BTS shall increment the Message_No and send the configuration in every new frame, until a DL_Ack is received, i.e. the TRAU is synchronized. The exchange is considered as terminated, when the last sent Message_No is received back.

If, however, already speech frames are received in uplink direction from the air interface before the TRAU is synchronized, then appropriate speech frames shall be sent. If the configuration parameters can be included in these speech frames (e.g. as for all Codec_Modes below 10,2 kbit/s in 16 kbit/s sub-multiplexing), then the procedure is exactly as described for No_Speech frames. If, however, the configuration parameters cannot be included, then every 4th speech frame shall be stolen on the Abis/Ater interface and be replaced by a No_Speech (No_Data) frame (generic configuration frame) to transmit the configuration.

C.8.3 Distant Configuration before TFO is established

After call set-up the TRAU may try to establish a TFO connection by using the TFO Protocol. During that time and before TFO is established the TRAU may get already knowledge about the distant configuration, either by TFO_REQ or TFO_Ack.

If distant and local configurations allow TFO (see Clauses 11 and 12 for the TFO Decision algorithm) then the TRAU shall immediately send TFO_Soon with the appropriate Rate Control to its local BTS. It may also include the partially known distant configuration parameters by using Dis_Req together with *TFO_Soon*.

Otherwise the distant configuration parameters shall be sent by using Dis_Req together with TFO_Off, when the information required for Codec Type and/or Configuration mismatch resolutions are available, either after TFO_REQ_L or TFO_ACK_L.

Dis_Req shall be used by the TRAUin downlink to transmit the distant or the optimal configuration parameters, when these have not been received by Con_Req or Con_Ack from the distant side.

C.8.4 Optimal TFO configuration

In TFO mode versions 5 and 6 (see C.2.4), the TFO Decision algorithm is only run by the TRAU. In this case the TRAU does not send the distant configuration to the BTS or the BSC, but the result of the TFO Decision algorithm, i.e. the optimal Codec Type and the optimal configuration parameters.

As soon as the optimal TFO configuration is known (result of the TFO Decision algorithm), the TRAU shall send it to the BTS by using Dis_Req.

C.8.5 Configuration Exchange in TFO

If TFO is ongoing (state OPERATION: the BTS is informed about that by *TFO_On*, see clause C.6.2) then the configuration sent by the BTS with Con_Req shall be relayed through by the local TRAU and the distant TRAU´ down to the distant BTS´. All devices in the path (TRAUs, but maybe also others, e.g. TCMEs) are updated to the new

configuration. The distant BTS´ shall acknowledge this by Con_Ack. This message takes the same way back. The exchange shall be considered terminated when the originating BTS received the Con_Ack.

NOTE: The round trip delay in TFO connections shall be considered.

In case of TFO with a non_AMR Codec Type of a release lower than REL-5 only TFO_REQ_L and TFO_ACK_L messages can be used for exchange of TFO Configuration data (mainly the Codec_List).

In case of TFO with an AMR or AMR-WB Codec Type the Config_Frames may be used instead, because they are substantially faster in transmission and are exactly traffic frame synchronised and they may come anyhow from the BTS within the traffic flow. TFO_REQ_L messages with the same piece of information may be transmitted as for non AMR Codec Types, but only one of these methods shall be used, either Con_Req or TFO_REQ_L, not both in parallel. In case of discrepancy between the Config_Frames and the TFO messages, the receiving side decides which shall have precedence.
In any case TFO_REQ_L must be acknowledged by a TFO_ACK_L and a Con_Req by a Con_Ack. . In the (rare) case that a TFO_ACK_L contains an embedded Con_Req frame, the parameters of the TFO_ACK_L shall be ignored, because the Con_Req travels faster and contains more recent configuration parameters.

C.8.6 Handover_Complete Notification in TFO

A new BTS shall reset an internal "Handover_Flag", when it is activated for a new call setup.
A new BTS shall set this internal Handover_Flag, when it is activated for a handover.

The new BTS shall send the "Handover_Complete Notification" within each Con_Req in the uplink direction as long as the Handover_Flag is set. The Handover_Flag shall be reset when receiving a Con_Ack from the distant side. A DL_Ack from the local TRAU shall not reset the Handover_Flag.

After a local handover, there are two events that trigger the new BTS to enter the TFO_YES State:

- a TFO_On Message (Inter-BSC handover and call setup);
- a Con_Ack Frame (Intra-BSC handover).

In the case of a local Inter-BSC handover a new TRAU is initialized. This new TRAU starts the TFO protocol with Not_Active. The Con_Req(loc) (with the Handover_Complete Notification) of the new BTS is acknowledged directly with a DL_Ack(empty) by the local TRAU. This shall not reset the Handover_Flag within the new BTS, but shall terminate the sending of the Con_Req(loc) in uplink. Later, a TFO_On message from the new local TRAU will trigger the new BTS to enter TFO_YES. In this case a Con_Req(loc) shall be sent to the distant side, because the time delay is not measured yet. Since the Handover_Flag is still set, the "Handover_Complete Notification" shall be included and the distant side is informed that a handover has taken place and the time delay has to be measured again. The distant BTS therefore shall send a Con_Ack(dis) to acknowledge the Con_Req(loc) and then a Con_Req(dis) and wait for the Con_Ack(loc) for delay determination.

In the case of a local Intra-BSC handover the TRAU typically doesn't change and therefore doesn´t interrupt the ongoing TFO connection. It remains in State Operation. Therefore no TFO_On message will be sent to the new local BTS. In this case, the Con_Req(loc) (with the Handover_Complete Notification) of the local BTS will not be acknowledged by the local TRAU, but directly with a Con_Ack(dis) by the distant BTS. This Con_Ack(dis) allows to determine the round trip delay on the local side, resets the Handover_Flag and triggers the local BTS to enter TFO_YES. No further Con_Req(loc) has to be sent to the distant side because the time delay was already measured. Since the distant side has received the Handover_Complete Notification, it knows that the time delay has to be measured again on its side. The distant BTS therefore shall send a Con_Req(dis) and wait for the Con_Ack(loc) for delay determination.

3GPP TSG-SA4 Meeting #21
Rennes, France, 13-17 May 2002

Tdoc S4-020359

CR-Form-v6.1

CHANGE REQUEST

⌘ **TS 28.062** **CR** **024** ⌘ **rev** **2** ⌘ Current version: **5.0.0** ⌘

Spec Title: **Inband Tandem Free Operation (TFO) of speech codecs** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network **X** Core Network **X**

Title:	⌘	**Corrections to Annex H**
Source:	⌘	**TSG SA WG4**
Work item code:	⌘	AMRWB
Date:	⌘	11-June-2002
Category:	⌘	**F**
Release:	⌘	**REL-5**

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
REL-4 (Release 4)
REL-5 (Release 5)

Reason for change:	⌘	Minor errors and some missing parts in Annex H: Generic Configuration Frames
Summary of change:	⌘	Corrections and Additions in Annex H
Consequences if not approved:	⌘	Incomplete specification

Clauses affected:	⌘	Annex H		
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ O&M Specifications	⌘	none
Other comments:	⌘	Please observe editor´s notes on document formatting (in yellow)		

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

Annex H (normative):
Definition of the Generic Configuration Frames for TFO

H.1 Scope

Annex H describes the **Generic Configuration Frames** for TFO. They may be used on the A-Interface and on the Abis/Ater-Interface for all Codec Types.

They are designed to carry the same information as the TFO_REQ_L (TFO_ACK_L) Messages, see section 7.

These Generic Configuration Frames are based on the design of the Codec List as layed down in TS 26.103.

H.2 Structure for Generic Configuration Frames

H.2.1 Frame Structure for 8 kBit/s sub-multiplexing

The frame structure is defined in TS 48.061 REL-5 and is reprinted here for ease of use.

	TRAU8k / TFO8k Generic Configuration Frame Bit number							
Octet no	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	0
2	1	C1 = 1	C2 = 1	C3 = 1	C4 = 1	C5 = EMBED	D1	D2
3	0	1	D3	D4	D5	D6	D7	D8
4	1							D15
5	1							D22
6	1							D29
7	1							D36
8	1							D43
9	1							D50
10	1							D57
11	1							D64
12	1							D71
13	1							D78
14	1							D85
15	1							D92
16	1							D99
17	1							D106
18	1							D113
19	1							D120
20	1	D121	D122	D123	D124	**D125**	T1	T2

H.2.2 Frame Structure for 16 kBit/s sub-multiplexing

The frame structure is defined in TS 48.060 REL-5 and is reprinted here for ease of use.
(Editor´s note to MCC: please add table grid lines and delete this note)

	TRAU16k / TFO16k Generic Configuration Frame Bit number							
Octet no.	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	1	C1 = 1	C 2= 1	C3 = 1	C4 = 1	C5= EMBED	D1	D2
3	D3	D4	D5	D6	D7	D8	D9	D10
4	1							
5								D25
6	1							
7								D40
8	1							
9								D55
10	1							
11								D70
12	1							
13								D85
14	1							
15								D100
16	1							
17								D115
18	1							
19								D130
20	1							
21								D145
22	1							
23								D160
24	1							
25								D175
26	1							
27								D190
28	1							
29								D205
30	1							
31								D220
32	1							
33								D235
34	1							
35								D250
36	1							
37								D265
38	1							D272
39	D273	D274	D275	**D276**	T1	T2	T3	T4

H.3 Coding of Generic Configuration Frames

The coding of Generic Configuration Frames in 8 kBit/s and 16 kBit/s sub-multiplexing follow exactly the same rules. The only difference is that 8k frames carry less configurations bits and may need an extension frames earlier.

H.3.1 Generic Configuration Frame Administration Section

H.3.1.1 Extendability

The first bits of each Generic Configuration Frame is reserved for the configuration frame administration.

FOLLOW: D1, 1 bit.
If FOLLOW is set to "0", then this is the first Generic configuration frame, if FOLLOW is set to"1", then this is a second or further Generic configuration frame.

EXTEND: D2, 1 bit.
If EXTEND is set to "0", then no further Generic configuration frame is following, –if EXTEND is set to "1", then an additional Generic configuration frame will follow. This next Generic configuration frame may follow immediately, or with a maximum distance of ~~[~~3~~]~~ frames in between to allow "house-keeping" for the active codec type.

Then follows a sub-selector field that allows future extension to the Generic Configuration Frame design.

CON_SEL: D3..D5, 3 bits
Coding: D3.D4.D5 = 0.0.0: **TFO_Configuration_Frame**, all other codes are reserved.
A receiver that does not understand a (for it) reserved code shall ignore the whole configuration frame.
Note: A potential application in future could be the introduction of a DTMF_Frame.

H.3.1.2 Version Handling

A field for a Version**.**Subversion is following:
Ver.Sver: D6..D9.D10..D13, 4+4 bits;
Example for Coding: 0101.0001~~0100.0011~~ is used to code "REL 5.1.x ~~4.3.x~~".
Details for handling of the version in the TFO procedures ~~is ffs.~~ are defined in clause 4.3.

H.3.1.3 Configuration Exchange Protocol

Then the next part of each Generic Configuration Frame shall contain the protocol related parameters:
Config_Prot: D14..D16, 3 bits
Mess_No: D17.D18, 2 bits
ParType: D19.D20, 2 bits
~~**OD:** 1 bit~~ Editor´s note: moved down to H.3.1.4, because it is not a protocol element, but rather a configuration parameter, sent from BTS (BSC) to TRAU to define the TRAU behaviour, like DTXd.
The definitions are~~is~~ given in Annex C. ~~This is for further editorial improvments.~~

Editor´s note: It seems that it would be more readable and better structured,
if clause C.6 of Annex C would be moved into Annex H.

H.3.1.4 System Identification, TFO and DTX control

System Identification (Sys_ID), DTXd~~, and~~ TFOE and OD are included in Generic Configuration Frames:
Sys_ID: D21~~2~~..D28~~9~~, 8 bits (see TS 26.103 and Annex A.5).
DTXd: D29~~30~~, 1 bit
TFOE: D30~~1~~, 1 bit
OD: D31~~21~~, 1 bit

H.3.1.5 Specific Section for the Active Codec Type

Now follows a specific section for the Active Codec Type (==Local Used Codec). This section has a flexible design to allow future adaptations. It carries signals that are important for the real-time operation of the active codec type ~~(e.g. CMI/CMR and RIF for AMR and AMR-WB)~~.

Active_Codec_Type: D32..D39, 8 bits
ACT_Specific_Length: D40..D42, 3 bits.
ACT_Specific_Extend: D43, 1 bit.
Active_Codec_Type defines the Codec Type that is currently used. The coding is according to TS 26.103, clause 5 (CoID).
ACT_Specific_Length defines the length of the proprietary section in multiples of 8 bits (octets).
ACT_Specific_Extend specifies an extension of this, in case these 56 bits are not sufficient. If ACT_Specific_Extend is set to "0", then no additional proprietary section follows. If ACT_Specific_Extend is set to "1" then after the first proprietary section again a second ACT_Specific_Length and ACT_Specific _Extend Field are following, and so on.

~~Note: Typically ACT_Specific_Length may be set to "1" and Prop_Extend to "0" and so 12 bits are used for the proprietary section.~~

H.3.1.5.1 Specific Section for GSM_FR, GSM_HR, GSM_EFR

If the Active Codec Type is either GSM_FR or GSM_HR or GSM_EFR, then the parameters are set to:

ACT_Specific_Length := 0.0.0 (no byte is following)
ACT_Specific_Extend := 0 (no further extension).

H.3.1.5.2 Specific Section for the AMR Narrow Band Family

If the Active Codec Type is either FR_AMR, HR_AMR, UMTS_AMR, UMTS_AMR2 or OHR_AMR, then the parameters are set to:

ACT_Specific_Length := 0.0.1 (one byte is following)
ACT_Specific_Extend := 0 (no further extension),

and the following parameters are defined in addition:

RIF:	D44,	1 bit, Request or Indication Flag, as defined in TS 48.060.
CMI_abs:	D45..D47,	3 bits, Codec Mode Indication, as defined in TS 48.060.
CMR_abs:	D48..D50,	3 bits, Codec Mode Request, as defined in TS 48.060.
spare:	D51,	1 bit, reserved for future use, set to “0”.

H.3.1.5.3 Specific Section for the AMR Wide Band Family

If the Active Codec Type is either FR_AMR-WB, UMTS_AMR-WB, OFR_AMR-WB or OHR_AMR-WB, then the parameters are set to:

ACT_Specific_Length := 0.0.1 (one byte is following)
ACT_Specific_Extend := 0 (no further extension),

and the following parameters are defined in addition:

RIF:	D44,	1 bit, Request or Indication Flag, as defined in TS 48.060.
CMI_abs:	D45..D47,	3 bits, Codec Mode Indication, as defined in TS 48.060.
CMR_abs:	D48..D50,	3 bits, Codec Mode Request, as defined in TS 48.060.
spare:	D51,	1 bit, reserved for future use, set to “0”.

H.3.1.6 Spare Bits

If bits remain after the last used configuration parameters, see H.~~2.~~3.2, then these bits shall be filled with "0" (spare code in Generic Configuration Frames).

H.3.1.7 Error Detection and Error Handling

~~The error detection is placed at the end of each Configuration Frame.~~ A Generic Configuration Frame contains important information and is protected by an 8-bit-CRC including C1..C5, all data bits and all spare bits.

The 8-bit-CRC parity bits ~~(as defined in ...)~~ shall be placed at a fixed position at the very end of the Generic Generic Configuration Frame:
CRC8k: D118 to D125 in TRAU8k / TFO8k frames and
CRC16k: D269 to D276 in TRAU16k / TFO16k frames.

These 8 parity bits are in both cases generated by the cyclic generator polynomial:
$g(D) = D^8 + D^4 + D^3 + D^2 + 1.$
The encoding is performed in a systematic form, which means that, in GF(2), the polynomial:

- $b(1)D^{(N+8-1)} + b(2)D^{(N+8-2)} + ... + b(N)D^8 + p(1)D^7 + p(2)D^6 + ... + p(7)D^1 + p(8)$;
- p(1) - p(8): the parity bits (D118 - D125 or the parity bits (D269 - D276);
- b(1) - b(N): the data bits (C1- C5, D1-D117) or the data bits (C1- C5, D1 – D268);

when divided by *g(D)*, yields a remainder equal to 0.

A Generic Configuration Frame with CRC-error shall be regarded as invalid and shall be ignored, i.e. its parameters shall not be used and it shall not be acknowledged. A TRAU passing these Generic Configuration Frames from the Abis interface to the A interface or vice versa shall not correct the CRC, if errors are detected. If the TRAU has to recalculate the CRC and it detects at the end that the incoming CRC indicated a transmission error, then the TRAU shall deliberately invert the newly calculated CRC before sending it along.

H.3.2 Configuration Parameter Section

The Configuration Parameter section fits between the ACT specific section and the Error Protection section. If not enough space is left there, then another Generic Configuration Frame shall be used for the remaining parameter bits. These remaining bits shall be placed in the next Configuratioin Frame starting after the ACT specific section, and so on.

H.3.2.1 Mapping for Single Codec Type

An exactly defined **Mapping between TS 26.103 and TFO_Configuration_Frames** exists.
This is defined as follows:

The "Single_Codec" identifier as defined in TS 26.103 is omitted.

The "Length_Indicator" is shortened to 3 bits and an "Extension_Indicator" is introduced in addition.
That allows directly up to 7 octets for parameters per Codec Type. If this is not sufficient (potentially in future cases), then the "Extension_Indicator" is ~~-~~set to "1" and then a Length_Indicator and Extension_Indicator is again following with again a parameter field of up to 7 octets for the remaining configuration parameters, and so on.
The Length_Indicator counts all octets after the OID_Indicator.

The "Compatibility Information" is omitted, when not necessary. This is indicated by a single bit ("Compatibility_Information_Indicator") that is set to "0" normally and to "1" if the Compatibility_Information octet is present.

The "Organisation IDentifier" (OID) is omitted, when not necessary. This is indicated by a single bit ("OID_Indicator") that is set to "0" normally and to "1", if the OID octet is present.
When the OID is omitted then OID==”ETSI” shall be assumed.

The "CoID" (Codec Type Identifier) is exactly copied (8 bits).

The configuration parameters are exactly copied as specified in TS 26.103, MSB first.
Table H.3.2-1 summarises the design for the example “FR AMR” as one Codec Type in the Codec List.

Table H.3.2-1 Design of the Codec Type Configuration for the example FR AMR

Name	TS 26.103	TS 28.062	Comment
Single_Codec_Indicator	8~~0~~ bits	0 bits	o~~O~~mitted in TS 28.062
Length_Indicator	8~~3~~ bits	3 bits	“1.0.0” (4 octets following after the Organisation_Identifier_Indicator)
Extension_Indicator	~~1 bit~~ -	1 bits	“0” no further Extension necessary
Compatibility_Information_Indicator	~~1 bit~~ -	1 bit	“0”~~0~~ Compatibility_Information is omitted
Organisation_Identifier_Indicator	1 bit	1 bit	“0” Organisation_Identifier is omitted
Compatibility_Information	8 bits	0 bits	o~~O~~mitted, when not indicated
Organisation_Identifier	8 bits	0 bits	o~~O~~mitted, when not indicated
Codec_Type_Identifier	8 bits	8 bits	“FR_AMR_CoID”

ACS	8 bits	8 bits	0.1.0.0.1.1.0.1 (e.g.)
SCS	8 bits	8 bits	1.1.1.1.1.1.1.1 (can be omitted)
OM, MACS	8 bits	8 bits	0.0.0.0.0.0.0.0 (can be omitted)

For the example "AMR with all configuration parameters present" the coding in TS 26.103 takes 8*8=64 bits, while the coding in the Configuration frame takes 6+8+3*8=38 bits, with de facto identical contents. In the case of full support (i.e. SCS and OM, MACS omitted) the relation is 48 bits to 22 bits.

H.3.2.2 Codec List

If more Codec Types are present in the Codec List, then they shall follow one by one, each one coded as specified in H3.2.1 above.

The Codec Types shall be ordered according to their preference.

Per default the most preferred Codec Type shall be the first in the list (as in TS 26.103). Then Par_Type shall be set to “0.1” (local configuration parameters) or “1.0” (distant configuration parameters).

The first Codec Type in the Codec_List shall be the optimal Codec Type, when sent by the TRAU downlink with Par_Type set to “1.1”.

3GPP TSG-SA4 #21 **S4-020353**
Rennes, France, 13-17 May 2002

CR-Form-v6.1

CHANGE REQUEST

⌘ TS 28.062	CR 026	⌘ rev 1	⌘ Current version:	5.0.0 ⌘

Spec Title: Inband Tandem Free Operation (TFO) of speech codecs ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network X Core Network ☐

Title:	⌘	**Corrections to sections 9 and 10**
Source:	⌘	TSG SA WG4
Work item code:	⌘	AMRWB
Date:	⌘	2002-06-11
Category:	⌘	**F**
Release:	⌘	**REL-5**

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
REL-4	*(Release 4)*
REL-5	*(Release 5)*

Reason for change:	⌘	Inconsistencies
Summary of change:	⌘	Add TFO_Term in Fig. 9-1 and text of Clause 9. Correct Conditions for TFO_Frame to include AMR-WB. Minor changes in protocol tables in Clause 10 (insert/delete semicolons, spaces, …) .
Consequences if not approved:	⌘	Spec. is less readable and understandable; may result in misunderstandings.

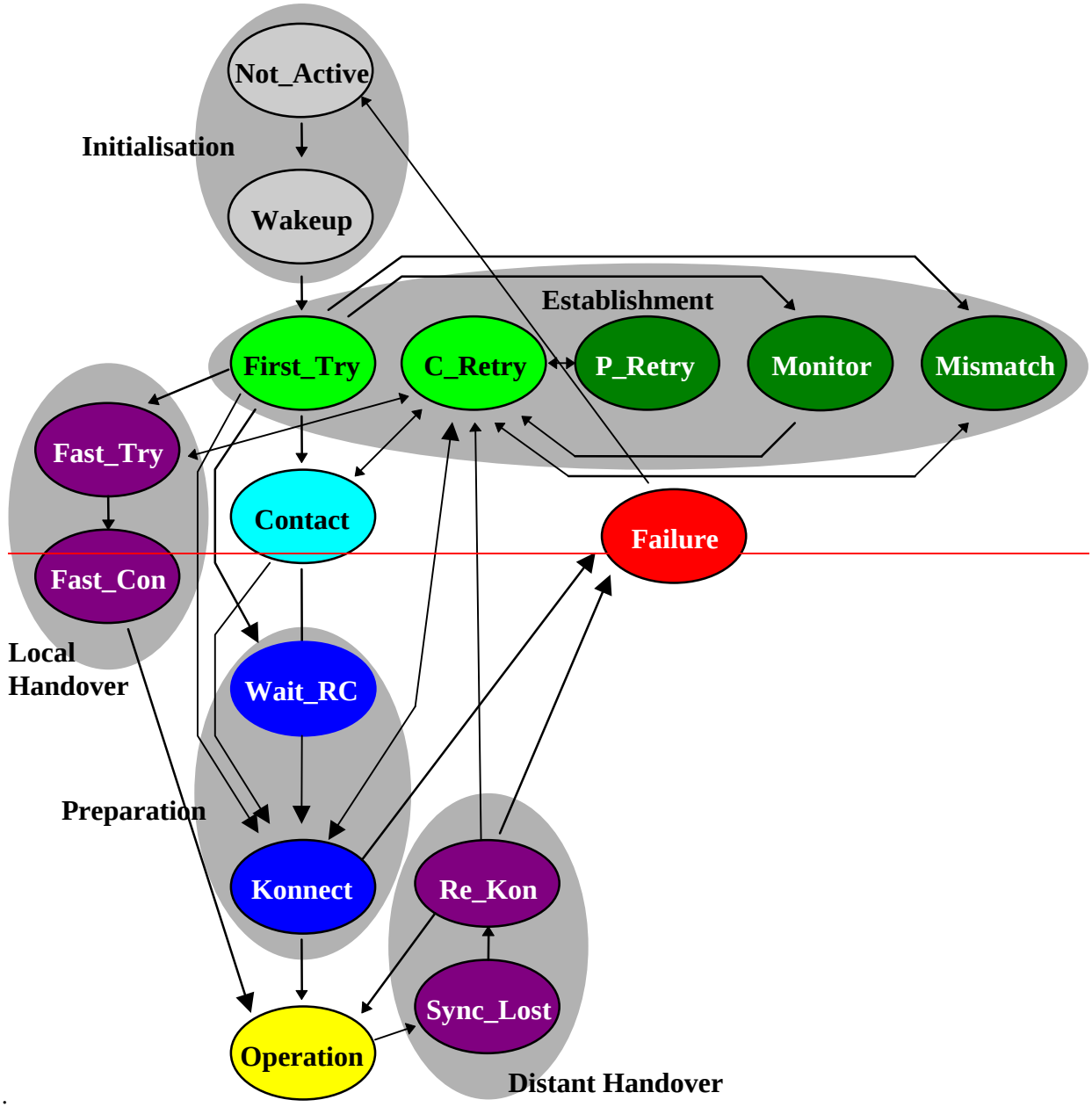
Clauses affected:	⌘	9, 10
Other specs affected:	⌘	☐ Other core specifications ⌘ ☐ Test specifications ☐ O&M Specifications
Other comments:	⌘	

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

9 TFO State Machine

A State Machine, consisting of 17~~16~~ States can describe the TFO_Protocol Process, see the following figure.


Not_Active
Initialisation
Wakeup
Establishment
First_Try
C_Retry
P_Retry
Monitor
Mismatch
Fast_Try
Contact
Failure
Fast_Con
Local
Handover
Wait_RC
Preparation
Konnect
Re_Kon
Sync_Lost
Operation
Distant Handover

.

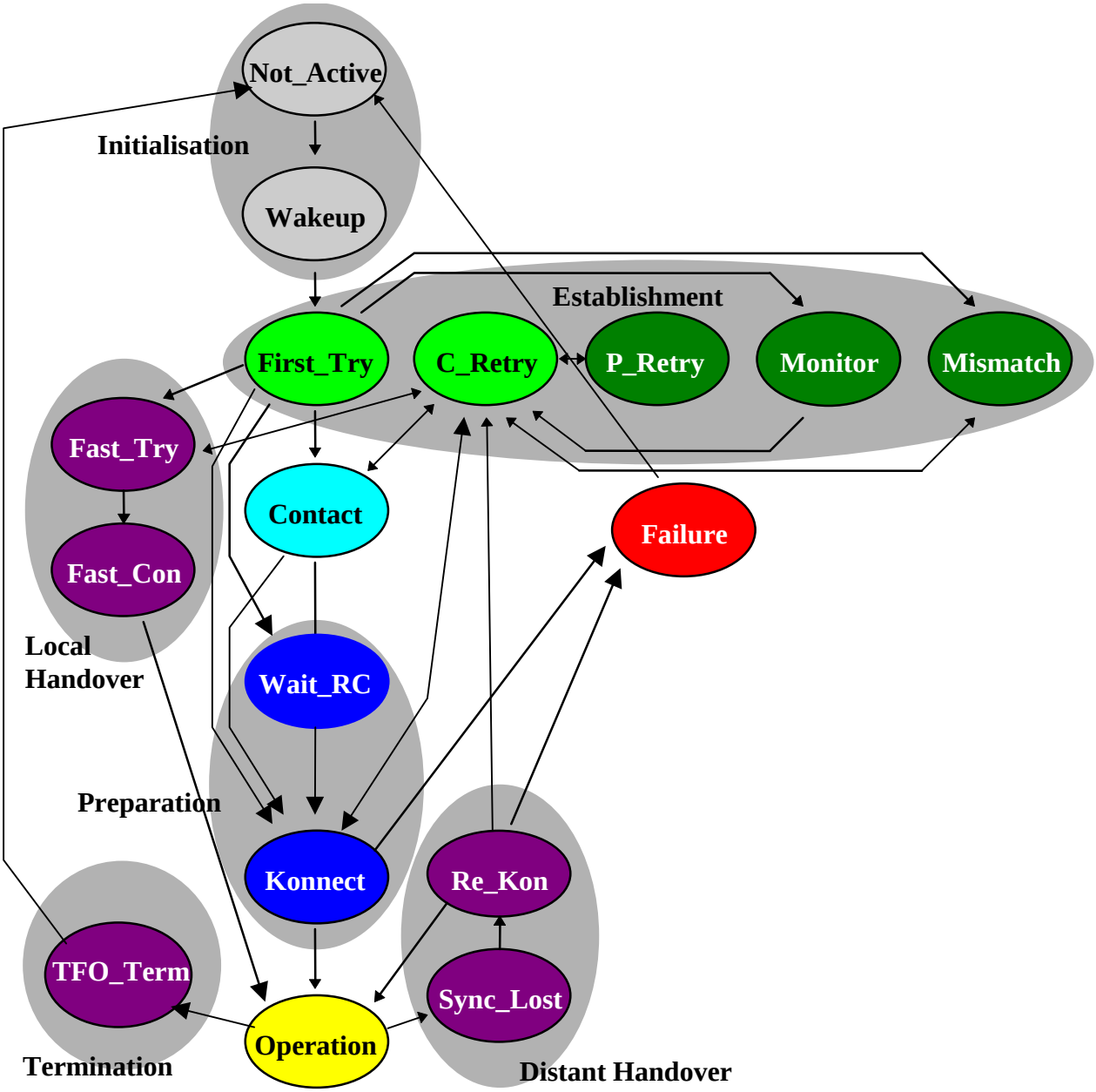


Figure 9-1: TFO_Protocol State Machine with most important transitions

There are five main States:

- Initialisation (• Not_Active, • Wakeup)
- Establishment (• First_Try, • Continuous_Retry, • Periodic_Retry, • Monitor, • Mismatch)
- Contact (• Contact)
- Preparation (• Wait_RC, • Konnect)
- Operation (• Operation)

Exception handling needs further States (see figure 9-1):

- Local Handover (• Fast_Try, • Fast_Contact).
- Distant Handover (• Sync_Lost, • Re_Konnect).
- Misbehaviour (• Failure).

- Termination (• TFO_Term).

It is assumed that Events (Conditions checking), Actions and Transitions to another State are handled almost instantaneous and in any case significantly faster than the time required to complete the transmission of any TFO Message or TFO Frame.

10 Detailed Description of the TFO Protocol

10.2.2 Conditions for TFO_Frame

In the context of a TFO_Frame event the conditions Match_1, Match_2, Mismatch_1, and Mismatch_2 are used. N represents the number of consecutive TFO frames received, corresponding to the conditions.

Match_1
Match_1 is fulfilled if one of the following conditions is true:

- A non-AMR codec type is used and
  the distant used codec type is equal to the local used codec type (Duc==Luc) and
  n<3.

- An AMR or AMR-WB codec type is used and
  the local used codec type and the distant used codec type are compatible and
  the distant used codec mode is contained in the local ACS and
  n<3

- An AMR or AMR-WB codec type is used and
  the local used codec type and the distant used codec type are compatible and
  a Non_Speech TFO frame (i.e. Sid_First, Sid-Update, Sid_Bad, No_Data and Onset) is received and
  n<3.

Match_2
Match_2 is fulfilled if one of the following conditions is true:

- A non-AMR codec type is used and
  the distant used codec type is equal to the local used codec type (Duc==Luc) and
  n>2.

- An AMR or AMR-WB codec type is used and
  the local used codec type and the distant used codec type are compatible and
  the distant used codec mode is contained in the local ACS and
  n>2

- An AMR or AMR-WB codec type is used and
  the local used codec type and the distant used codec type are compatible and
  a Non_Speech TFO frame (i.e. Sid_First, Sid-Update, Sid_Bad, No_Data and Onset) is received and
  n>2.

Mismatch_1
Mismatch_1 is fulfilled if one of the two following conditions is true:

- A non-AMR codec type is used and
  the distant used codec type is different from the local used codec type (Duc!=Luc) and
  n==1.

- An AMR or AMR-WB codec type is used and
  the TFO frame doesn't match because of incompatible codec types or a used codec mode that is not in the ACS and
  n<3.

Mismatch_2
Mismatch_2 is fulfilled if one of the following conditions is true:

- A non-AMR codec type is used and
  the distant used codec type is different from the local used codec type (Duc!=Luc) and
  n>1.

- An AMR or AMR-WB codec type is used and
  the TFO frame doesn't match because of incompatible codec types or a used codec mode that is not in the ACS and n>2.

10.5 Actions Table

Table 10.5-2 list all actions that can be performed by the TFO protocol. The syntax is defined in Table 10.5-1.

Table 10.5-1: Definition of Syntax for Action Table

Name	Action List	Comment
<Action Name>	<Action >;[<Action >;]	<Comment>
…		
<Action Name>	<Action >;[<Action >;]	<Comment>

The following notations are used in Table 10.5-2.

The **Transmit Queue** or **Tx_Queue** is a **F**irst-**I**n **F**irst-**O**ut command queue. It is filled by TFO_Protocol and read by the Transmit Process (e.g. Tx_TFO in Annex C).

The **Transmit Process** or **Tx_TFO** is the Process responsible for the scheduling and transmission of TFO Messages and TFO Frames to the distant partner.

The **Receive Process** or **Rx_TFO** is the Process responsible for the reception of TFO Messages and transfer to the TFO_Protocol.

Tx := TFO_REQ means, that TFO_Protocol places a command TFO_REQ in Tx_Queue. The Transmit Process should then generate a TFO_REQ Message for transmission when it comes to that command.

Tx := 31*TFO_REQ means: put 31 TFO_REQ commands in Tx_Queue. Not necessarily all will generate TFO_REQ Messages. In most cases Tx_Queue will be cleared before. Similar definitions hold for the other messages.

Clear Tx_Queue means that all remaining commands are deleted from the Tx_Queue in that very moment (time *Tc*).

Note that due to the duration required to fully transmit a TFO Message, the TFO_Protocol Process is often already in a different state while TFO Messages commanded in earlier States are still in the Tx_Queue or under transmission.

BSS := TFO () means that a message is sent to the local RAN.

Tx_TRAU := … means that a message is sent to the downlink Transmit Process of the Transcoder.

Tx_TFO := … means that a message is sent to the uplink transmit process of the transcoder.

One Timer **T :=** <**Time_out**> is required to describe time out situations. The notation **T := DIS** means that the Timer is disabled. Positive values are decremented in a hidden background process in steps of 20 ms. When T reaches '0', the TFO_Protocol Process is invoked.

Table 10.5-2: Defined Actions

Name	Actions	Comments
C	Clear Tx_Queue; T := DIS;	Initialise Tx_Queue and disable the timer.
T1	T := 1s;	Set Timeout to 1 second.
T2	T := 2s;	Set Timeout to 2 seconds.
T5	T := 5s;	Set Timeout to 5 seconds.
NoAc	.	No Action required.
S	Lsig := New_Random_Number; Old_Sig := UNKNOWN	Generate new Signature and set Old_Sig to unknown.
SO	Old_Sig := Lsig; Lsig := New_Random_Number	Remember old Signature and generate a new Signature.
U	Old_Sig := UNKNOWN;	Reset Old_Sig.
F	Tx := 3*TFO_FILL;	Put three TFO_FILL messages into Tx_Queue.
T	Tx := TFO_TRANS ();	Put one TFO_TRANS message into Tx_Queue.
N	Tx := TFO_NORMAL;	Put one TFO_NORMAL message into Tx_Queue.
REQ	Tx := 35*TFO_REQ;	Put 35 TFO_REQ messages into Tx_Queue.
ACK	Tx := 7*TFO_ACK;	Put seven TFO_ACK messages into Tx_Queue.
SYL1	Tx := TFO_SYL;	Put one TFO_SYL message into Tx_Queue.
SYL	Tx := 4*TFO_SYL;	Put four TFO_SYL messages into Tx_Queue.
DUP	Tx := 5*TFO_DUP;	Put five TFO_DUP messages into Tx_Queue.
L1	Tx := TFO_REQ_L;	Put one TFO_REQ_L message into Tx_Queue.
L	Tx := 6*TFO_REQ_L;	Put six TFO_REQ_L messages into Tx_Queue.
LA	Tx := TFO_ACK_L;	Put one TFO_ACK_L message into Tx_Queue.
BT	Tx := Begin_TFO;	Begin Transmission of TFO Frames.
DT	Tx := Discontinue_TFO;	Discontinue Transmission of TFO Frames.
IT	Tx_TRAU := Ignore_TFO; Tx_TRAU := TFO_Off;	A soon as no TFO frames are received any longer, the downlink transmit process works as conventional downlink TRAU/TC. Additionally, a TFO_Off message is sent at this time.
AT	Tx_TRAU := Accept_TFO; Tx_TRAU := TFO_On;	Downlink Transmit Process bypasses TFO_Frames. Additionally, a TFO_On message is sent.
B	BSS := TFO ();	Send TFO relevant information to the BSS or MSC. Successive identical information shall not be sent more than once.
RCm	Tx_TRAU := Set_Max_Rate(); Tx_TFO := Set_Max_Rate();	RCm (Rate Control maximum value): This action is only relevant for AMR or AMR-WB codec types and releases the codec mode steering by setting the local max rate to the maximum value (i.e. 7).
RCs	Tx_TRAU := Set_Max_Rate(); Tx_TFO := Set_Max_Rate();	RCs (Rate Control for Subset): This action is only relevant for AMR or AMR-WB codec types and steers the rate control depending on the TFO decision situation in order to continue TFO on a subset of the ACS if necessary.
RCi	Tx_TRAU := Set_Max_Rate(); Tx_TFO := Set_Max_Rate(); Tx_TRAU := TFO_Soon;	RCi (Rate Control initial): In the case of an AMR or AMR-WB codec type, this action steers the rate control down to the TFO_Setup_Mode in order to start TFO using this mode. Additionally, a TFO_Soon message is sent to the BTS. This TFO_Soon message will be acknowledged by the BTS. The acknowledgement yields as an event to leave the WAIT_RC state.)
RCh	Tx_TRAU := Set_Max_Rate(); Tx_TFO := Set_Max_Rate();	RCh (Rate Control for hand-over): This action is only relevant for AMR or AMR-WB codec types and steers the rate control down to the Hand_Over_Mode in order to continue TFO after hand-over using this mode.
CA	Tx_TFO := Con_Ack();	Send a Con_Ack (config frame) to the distant TRAU/TC.
CA1	Wait round trip time to RNC; Tx_TFO := Con_Ack();	Wait round trip time to RNC (e.g. send first a RC_REQ to the RNC and wait for the corresponding RC_ACK). Then send a Con_Ack to the distant TRAU/TC.
CR	TX_TFO := Con_Req();	This action is conditional and only relevant for 3G systems (TC). If the entity is a TC then send a Con_Req with TFO_Disable to the distant TRAU/TC.

10.6 Protocol Tables

Note to the MCC editor: In several cells of the following tables the required changes are very minor, e.g., the deletion or insertion of single characters like ";" (semicolon) or " " (space). In general, each abbreviation like "NoAc", "S", or "IT" needs to be terminated with a semicolon. There shall be no space in between the abbreviation and the semicolon, e.g., "NoAc;" is correct but "NoAc ;" isn't. In the first column of each table, the state-abbreviations ("NAC", "WAK", … "TT") are terminated by ":". Unfortunately, the consisten implementation of these changes is important because code may be generated automatically from these tables. Hence, additional notes highlight these changes in order to be overlooked less likely. These notes, highlited with yellow background, are not part of the specification! Furthermore, in Table 10.6-1, three cells need to be merged into one. Though this is not visible in a printout, it is necessary for automatic code generation.

Table 10.6-1: Enabling/Disabling/New_Speech_Call/TRAU_Idle

Event: or	**TFO_Enable New_Speech_Call**	**TFO_Disable TRAU_Idle**
Number:	1, 2	3, 4
Condition: &		
Comment:	TFO gets active.	Local disable.
State:		
NAC: Not_Active	C;S;IT;RCm; WAK; ins. «;»	NoAc; NAC;
WAK: Wakeup	NoAc; WAK; ins. «;»	NoAc; NAC;
FIT: First_Try	---------- ----------	C;N; NAC;
COR: Continuous Retry	---------- ----------	C;N; NAC;
PER: Periodic Retry	---------- ----------	C;N; NAC;
MON: Monitor	---------- ----------	C;N; NAC;
MIS: Mismatch	---------- ----------	C;N; NAC;
CON: Contact	---------- ----------	C;N; NAC;
FAT: Fast Try	---------- ----------	C;N;RCm; NAC;
FAC: Fast Contact	---------- ----------	C;N;RCm; NAC;
WRC: Wait_RC	---------- ----------	C;N;RCm; NAC;
KON: Konnect	---------- ----------	C;RCm;CR;DT;N;T1; TT;
REK: Re_Konnect	---------- ----------	C;RCm;CR;DT;N;T1; TT;
SOS: Sync_Lost	---------- ----------	C;RCm;IT;N; NAC;
OPE: Operation	---------- ---------- merge cells	C;RCm;CR;DT;N;T1; TT; rem. «;»
FAI: Failure	---------- ----------	C; NAC; **Exit from FAI**
TT: TFO_Term	---------- ----------	NoAcC; TT;

Table 10.6-2: PCM_Non_Idle and Loopback Handling

Event:	PCM_Non_Idle	TFO_REQ	TFO_REQ
Number:	5	6	7
Condition: & &		(NA_TP | A_TP) Dsig==Lsig Dsig!=Old_Sig	(NA_TP | A_TP) Dsig==Old_Sig
Comment: **State:**	Occurs only at the beginning	Loopback (LB) or distant handover (HO)? wrong Sig	Loopback (LB) or distant handover (HO)?
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	C;F;REQ; FIT; **Typ 2nd Event**	---------- ----------	---------- ----------
FIT: First_Try	---------- ----------	C;SO;REQ; FIT; **LB!**	NoAc; FIT; **Ignore LB**
COR: Continuous Retry	---------- ----------	C;SO;REQ; COR; LB!?	NoAc; COR; **Ignore LB**
PER: Periodic Retry	---------- ----------	C;F;S;ACK; CON; Dist HO!	---------- ----------
MON: Monitor	---------- ----------	C;F;S;REQ; FIT; Dist HO!	---------- ----------
MIS: Mismatch	---------- ----------	C;F;S;ACK; CON; Dist HO!	---------- ----------
CON: Contact	---------- ----------	C;SO;REQ; COR; Safe way	---------- ----------
FAT: Fast Try	---------- ----------	C;SO;REQ;RCm; COR; Safe way	---------- ----------
FAC: Fast Contact	---------- ----------	C;SO;REQ;RCm; COR; Safe way	---------- ----------
WRC: Wait_RC	---------- ----------	C;SO;RCm;REQ; COR;	---------- ----------
KON: Konnect	---------- ----------	C;DT;SO;RCm;-REQ;T1; COR; rem. «,» ins. «;» IPEs transparent!	---------- ----------
REK: Re_Konnect	---------- ----------	C;DT;SO;RCm;REQ;IT;B;T1; COR; IPEs transparent!	---------- ----------
SOS: Sync_Lost	---------- ----------	C;IT;S;RCm;REQ;B;T1; COR; Contact is back	---------- ----------
OPE: Operation	---------- ----------	---------- ----------	---------- ----------
FAI: Failure	---------- ----------	NoAc; FAI;	---------- ----------
TT: TFO_Term	---------- ----------	---------- ----------	---------- ----------

Table 10.6-3: Most Important Cases, Especially at Call Set-up

Event:	TFO_REQ	TFO_ACK	TFO_ACK	TFO_TRANS	TFO_Frame
Number:	8	9	10	11	12
Condition: & &	(NA_TP | A_TP) Dsig!=Lsig Dsig!=Old_Sig	NA_TP Dsig==Lsig	(NA_TP | A_TP) Dsig!=Lsig	Luc != AMR DCh==LCh	Match_1
Comment: **State:**	Distant REQ Good Signature	Distant ACK Good Signature	Wrong Response Handover?	similar to ACK As response to loc ACK_?	First or second TFO Frame
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;U;ACK; CON; **Typical**	C;U;T;BT;T;T1; KON; **Typical; IPEs!**	C;REQ; FIT;	NoAc; FIT; Wait for Frame	C;U;DUP;RCi; FAT; **1: HO**
COR: Continuous Retry	C;U;ACK; CON; Typical	C;U;T;BT;T;T1; KON; **Typical; IPEs!**	C;REQ; COR;	NoAc; COR; Wait for Frames	C;U;DUP; FAT; 1: Call is back?
PER: Periodic Retry	C;F;ACK; CON; OK, Contact is back	C;F;S;REQ; COR; Rare case, test	C;F;REQ; COR;	NoAc; PER; Wait for Frames	C;DUP; FAT; 1: Call is back?
MON: Monitor	C;F;REQ; FIT; IPEs?	C;F;S;REQ; FIT; Rare case, test	C;F;REQ; FIT;	NoAc; MON; ins. «;» Wait for Frames	C;DUP; FAT; 1: Call is back?
MIS: Mismatch	C;F;ACK; CON; Mismatch resolved	C;F;S;REQ; COR; Rare case, test	C;F;REQ; COR;	NoAc; MIS; Wait for Frames	C;DUP; FAT; 1: Call is back?
CON: Contact	C;ACK; CON; **Typical: wait**	C;T;BT;T;T1; KON; **Typical: yes!**	C;REQ; COR;	C;T;BT;T;T1; KON; **yes! Fast way**	C;T;BT;T;T1; KON; Missed TRANS?
FAT: Fast Try	C;REQ;RCm; COR; Safe way	C;REQ;RCm; COR; Safe way	C;REQ;RCm; COR; Safe way	NoAc; FAC; Wait for Frames	NoAc; FAT; **2: Typ. Loc HO**
FAC: Fast Contact	C;REQ;RCm; COR; Safe way	C;REQ;RCm; COR; Safe way	C;REQ;RCm; COR; Safe way	NoAc; FAC; Wait for Frames	C;BT;T;L;T2;AT;B; OPE; **5: Typ. Loc HO**
WRC: Wait_RC	C;RCm;REQ;T1; COR;	---------- ----------	C;RCm;REQ; COR;	---------- ----------	AT; ins. «;» WRC;
KON: Konnect	C;RCm;DT;REQ;T1; COR; IPEs transparent!	NoAc; KON; **Typical: wait**	NoAc; KON;	NoAc; KON; **Typical: wait**	RCs;AT;L;T2;B; OPE; **Typ: call set-up**
REK: Re_Konnect	C;RCm;DT;REQ;IT;B;T1; COR; IPEs transparent!	C;DT;REQ;IT;B;T1; COR;	C;DT;RCm;REQ;IT;B; T1 COR;	NoAc; REK; Wait for Frames	AT;L;T2;B; OPE; **5: Typ. Dis HO**
SOS: Sync_Lost	C;RCm;IT;REQ;B;T1; COR; Contact is back	C;IT;REQ;B;T1; COR; Contact is back	C;IT;RCm;REQ;B;T1; COR; Contact is back	NoAc; SOS; Wait for Frames	C;BT;T;L;T2;B; OPE; short Interrupt?
OPE: Operation	---------- ----------	---------- ----------	---------- ----------	NoAc; OPE; Typical in HO	NoAc; OPE; **Main! TFO!**
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------

Table 10.6-4: In Call Modification and Handover

Event: or	**New_Local_Codec New_Local_Config**	**New_Local_Codec New_Local_Config**	**TFO_Frame**	**TFO_SYL**	**TFO_DUP**
Number:	13, 14	15, 16	17	18	19
Condition: &	(NA_TP | A_TP)	TM	Match_2		
Comment: **State:**	In Call Modif. Mismatch resolv	In Call Modif. Mismatch occurs	Three or more TFO Frames	The dist TC lost sync in OPE	The dist TC recognised HO Identical #17
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	NoAc; WAK;	NoAc; WAK;	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;REQ; FIT; Restart	C;REQ; FIT; Restart	---------- ----------	NoAc; FIT; HO? Ignore	NoAc; FIT; HO? Ignore
COR: Continuous Retry	C;REQ; COR;	C;REQ; COR;	---------- ----------	NoAc; COR; Ignore	NoAc; COR; Ignore
PER: Periodic Retry	L1;T5; PER;	L1;T5; PER;	---------- ----------	C;F;REQ; COR; Rare case, test	C;F;REQ; COR; Rare case, test
MON: Monitor	NoAc; MON; ins. «;»	NoAc; MON; ins. «;»	---------- ----------	C;F;REQ; FIT; Rare case, test	C;F;REQ; FIT; Rare case, test
MIS: Mismatch	C;F;REQ; COR; **Mismatch Res.**	C;L;T2;B; MIS; **Direct info**	---------- ----------	C;F;REQ; COR; Rare case, test	C;F;REQ; COR; Rare case, test
CON: Contact	C;REQ; COR;	C;L;T2;B; MIS;	---------- ----------	C;F;REQ; COR; Rare case, test	C;F;REQ; COR; Rare case, test
FAT: Fast Try	NoAc; FAT;	C;L;T2;B;RCm; MIS;	NoAc; FAC;	NoAc; FAC; **3: Typ. Loc HO**	C;F;REQ;RCm; COR; Rare case, test
FAC: Fast Contact	NoAc; FAC;	C;L;T2;B;RCm; MIS;	C;BT;T;L;T2;AT;B;RCs; OPE; assume matching ACS	NoAc; FAC; **4: Typ Loc HO**	C;F;REQ;RCm; COR; rare case, test
WRC: Wait_RC	C;RCm;REQ; COR;	C;RCm;L;T2;B; MIS;	NoAc; WRC;	NoAc; WRC;	NoAc; WRC;
KON: Konnect	C;RCm;DT;REQ; COR;	C;RCm;DT;L;T2;B; MIS;	RCs;AT;L;T2;B; OPE;	NoAc; KON; Wait, short int?	NoAc; KON; Other TC?
REK: Re_Konnect	C;RCm;DT;IT;REQ; COR;	C;RCm;DT;IT;L;T2;B; MIS;	---------- ----------	C;DT;SYL; SOS; IPEs not transp?	NoAc; REK; **4: Typ. Dist HO**
SOS: Sync_Lost	C;RCm;IT;REQ; COR;	C;RCm;IT;L;T2;B; MIS;	---------- ----------	NoAc; SOS; Short Interrupt.?	C;BT;T;T1; REK; **3: typ Dis HO**
OPE: Operation	RCs;L;T2; OPE;	C;RCm;DT;IT;L;T2;B; MIS;	NoAc; OPE; Main! TFO!	NoAc; OPE; Short interrupt?	NoAc; OPE; Typical
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	C;F;REQ; COR;	NoAc; TT;	NoAc; TT;	IT;N; NAC;	NoAc; TT;

Table 10.6-5: Special Matching TFO Messages

Event:	**TFO_REQ_L**	**TFO_REQ_L**	**TFO_ACK_L**	**TFO_ACK_L**
Number:	20	21	22	23
Condition: & Dsig==Lsig	(NA_TP | A_TP)	(NA_TP | A_TP) Dsig!=Lsig	(NA_TP | A_TP) Dsig==Lsig	(NA_TP | A_TP) Dsig!=Lsig
Comment: **State:**	Only sent in MIS/OPE/PER HO? Loop?	Only sent in MIS/OPE/PER Codec_List	Only sent in MIS; HO?	HO?
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	NoAc; FIT; Ignore	NoAc; FIT; Ignore	NoAc; FIT; Ignore	NoAc; FIT; Ignore
COR: Continuous Retry	NoAc; COR; Ignore	NoAc; COR; Ignore	NoAc; COR; Ignore	NoAc; COR; Ignore
PER: Periodic Retry	C;F;S;REQ; COR; Start again	C;F;REQ; COR; Start again	C;F;S;REQ; COR; Test	C;F;REQ; COR; Test
MON: Monitor	C;F;S;REQ; FIT; Test	C;F;REQ; FIT; Test	C;F;S;REQ; FIT; Test	C;F;REQ; FIT; Test
MIS: Mismatch	C;F;S;REQ; COR; Test	C;F;REQ; COR; Test	C;F;S;REQ; COR; Test	C;F;REQ; COR; Test
CON: Contact	C;S;REQ; COR; Safe way!	C;REQ; COR; Safe way!	C;S;REQ; COR; Safe way!	C;REQ; COR; Safe way!
FAT: Fast Try	C;S;REQ;RCm; COR; Safe way!	C;REQ;RCm; COR; Safe way!	C;S;REQ;RCm; COR; Safe way!	C;REQ;RCm; COR; Safe way!
FAC: Fast Contact	C;S;REQ;RCm; COR; Safe way!	C;REQ;RCm; COR; Safe way!	C;S;REQ;RCm; COR; Safe way!	C;REQ;RCm; COR; Safe way!
WRC: Wait_RC	C;S;RCm;REQ; COR;	C;RCm;REQ; COR;	C;S;RCm;REQ; COR;	C;RCm;REQ; COR;
KON: Konnect	C;RCm;DT;S;REQ;T1; COR; Safe way!	C;RCm;DT;REQ;T1; COR; Safe way!	C;RCm;DT;S;REQ;T1; COR; Safe way!	C;RCm;DT;REQ;T1; COR; Safe way!
REK: Re_Konnect	C;RCm;DT;IT;S;REQ;T1; COR; Safe way!	C;RCm;DT;IT;REQ;T1; COR; Safe way!	C;RCm;DT;IT;S;REQ;T1; COR; Safe way!	C;RCm;DT;IT;REQ;T1; COR; Safe way!
SOS: Sync_Lost	C;RCm;IT;S;REQ;B;T1; COR; Safe way!	C;RCm;IT;REQ;B;T1; COR; Safe way!	C;RCm;IT;S;REQ;B;T1; COR; Safe way!	C;RCm;IT;REQ;B;T1; COR; Safe way!
OPE: Operation	S;L;T2;B; OPE; Tx Codec_List	C;RCs;LA;B; OPE; Ack List, stop	C;RCs;B; OPE; Ack ok, stop	S;L;T2;B; OPE; Exchange list
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	---------- ----------	C;B; TT;	C;B; TT;	---------- ----------

Table 10.6-6: TFO Messages with mismatching Codec Type / Configuration

Event:	**TFO_REQ**	**TFO_REQ**	**TFO_ACK**	**TFO_REQ_L**	**TFO_REQ_L**	**TFO_ACK_L**
Number:	24	25	26	27	28	29
Condition: &	TM Dsig==Lsig	TM Dsig!=Lsig	TM Dsig=?	TM Dsig==Lsig	TM Dsig!=Lsig	TM Dsig==?
Comment: **State:**	Mismatch Wrong Sig, HO?	Mismatch Good Sig	Mismatch w/wo HO identical #8	Mismatch Codec_List Wrong Sig, HO?	Mismatch Codec_List Identical #20	Mismatch Codec_List Identical #19
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;S;L;T2;B; MIS; Rare	C;U;L;T2;B; MIS; **Typical: Setup**	C;U;L;T2;B; MIS; HO?	C;S;LA;B; MIS; rare	C;U;LA;B; MIS; **Typical: Setup**	C;U;LA;B; MIS; HO?
COR: Continuous Retry	C;S;L;T2;B; MIS;	C;U;L;T2;B; MIS;	C;U;L;T2;B; MIS;	C;S;LA;B; MIS;	C;U;LA;B; MIS;	C;U;LA;B; MIS;
PER: Periodic Retry	C;F;S;L;T2;B; MIS;	C;F;L;T2;B; MIS;	C;F;L;T2;B; MIS;	C;F;S;LA;B; MIS;	C;F;LA;B; MIS;	C;F;LA;B; MIS;
MON: Monitor	C;F;S;L;T2;B; MIS;	C;F;L;T2;B; MIS;	C;F;L;T2;B; MIS;	C;F;S;LA;B; MIS;	C;F;LA;B; MIS;	C;F;LA;B; MIS;
MIS: Mismatch	C;S;L;T2;B; MIS;	C;L;T2;B; MIS;	C;L;T2;B; MIS;	C;S;LA;B; MIS;	C;LA;B; MIS; **Terminate Prot.**	C;LA;B; MIS; **Terminate Prot.**
CON: Contact	C;S;L;T2;B; MIS;	C;L;T2;B; MIS;	C;L;T2;B; MIS;	C;S;LA;B; MIS;	C;LA;B; MIS;	C;LA;B; MIS;
FAT: Fast Try	C;S;L;T2;B;RCm; MIS;	C;L;T2;B;RCm; MIS;	C;L;T2;B;RCm; MIS;	C;S;LA;B;RCm; MIS;	C;LA;B;RCm; MIS;	C;LA;B;RCm; MIS;
FAC: Fast Contact	C;S;L;T2;B;RCm; MIS;	C;L;T2;B;RCm; MIS;	C;L;T2;B;RCm; MIS;	C;S;LA;B;RCm; MIS;	C;LA;B;RCm; MIS;	C;LA;B;RCm; MIS;
WRC: Wait_RC	C;S;RCm;L;T2;B; MIS;	C;-RCm;L;T2;B; MIS; rem. space	C;-RCm;L;T2;B; MIS; rem. space	C;S;-RCm;LA;B; MIS; rem. space	C;-RCm;LA;B; MIS; rem. space	C;-RCm;LA;B; MIS; rem. space
KON: Konnect	C;RCm;DT;S;L;T2; B; MIS;	C;RCm;DT;L;T2; B; MIS;	C;RCm;DT;L;T2; B; MIS;	C;RCm;DT;S;LA; B; MIS;	C;RCm;DT;LA;B; MIS;	C;RCm;DT;LA;B; MIS;
REK: Re_Konnect	C;RCm;DT;S;L;T2; IT;B; MIS;	C;RCm;DT;L;T2; IT;B; MIS;	C;RCm;DT;L;T2; IT;B; MIS;	C;RCm;DT;S;LA; IT;B; MIS;	C;RCm;DT;LA;IT ;B; MIS;	C;RCm;DT;LA;IT; B; MIS;
SOS: Sync_Lost	C;RCm;S;L;T2;IT; B; MIS;	C;RCm;L;T2;IT; B; MIS;	C;RCm;L;T2;IT; B; MIS;	C;RCm;S;LA;IT; B; MIS;	C;RCm;LA;IT;B; MIS; **In_Call_Mod**	C;RCm;LA;IT;B; MIS;
OPE: Operation	---------- ----------	---------- ----------	---------- ----------	NoAc; OPE; Trans Error?	NoAc; OPE; Trans Error?	---------- ----------
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	---------- ----------	---------- ----------	---------- ----------	---------- ----------	C;B; TT;	C;B; TT;

Table 10.6-7 AMR and AMR-WB Cases: TFO_TRANS, TFO_ACK, RC_ack

Event:	**TFO_TRANS**	**TFO_ACK**	**RC_ack**
Number:	30	31	32
Condition: &	Luc == AMR DCh==LCh	A_TP Dsig==Lsig	
Comment: **State:**		Good Sig Immediate TFO possible	BTS has steered the mode.
NAC: Not_Active	---------- ----------	---------- ----------	NoAc; NAC;
WAK: Wakeup	---------- ----------	---------- ----------	NoAc; WAK;
FIT: First_Try	NoAc; FIT; Wait for Frame	C;U;RCi;ACK;T1; WRC; **Typical;**	NoAc; FIT;
COR: Continuous Retry	NoAc; COR; Wait for Frames	C;U;RCi;ACK;T1; WRC; **Typical**	NoAc; COR;
PER: Periodic Retry	NoAc; PER; Wait for Frames	C;F;S;REQ; COR; Rare case, test	NoAc; PER;
MON: Monitor	NoAc; MON; ins. «;» Wait for Frames	C;F;S;REQ; FIT; Rare case, test	NoAc; MON;
MIS: Mismatch	NoAc; MIS; Wait for Frames	C;F;S;REQ; COR; Rare case, test	NoAc; MIS;
CON: Contact	C;RCi;ACK;T1; WRC; Missed Ack	C;RCi;ACK;T1; WRC; **Typical**	NoAc; CON;
FAT: Fast Try	NoAc; FAC; Wait for Frames	C;REQ;RCm; COR; Safe way	NoAc; FAT;
FAC: Fast Contact	NoAc; FAC; Wait for Frames	C;REQ;RCm; COR; Safe way	NoAc; FAC;
WRC: Wait_RC	NoAc; WRC;	NoAc; WRC;	C; T;BT;T;T1; KON; **Typical**
KON: Konnect	NoAc; KON; **Typical: wait**	NoAc; KON; **Typical: wait**	NoAc; KON;
REK: Re_Konnect	NoAc; REK; Wait for Frames	C;DT;REQ;IT;B;T1; COR; ins. «;»	NoAc; REK;
SOS: Sync_Lost	NoAc; SOS; Wait for Frames	C;IT;REQ;B;T1; ins. «;» COR; Contact is back	NoAc; SOS;
OPE: Operation	NoAc; OPE; Typical in HO	---------- ----------	NoAc; OPE;
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	---------- ----------	---------- ----------	NoAc; TT;

Table 10.6-8 Handover_Soon

Event:	**Handover_Soon**	**Handover_Soon**
Number:	35	36
Condition: &	(NA_TP | A_TP)	TM
Comment: **State:**	Local hand-over future parameters	Local hand-over future parameters
NAC: Not_Active	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------
FIT: First_Try	C; NAC;	C; NAC;
COR: Continuous Retry	C; NAC;	C; NAC;
PER: Periodic Retry	C; NAC;	C; NAC;
MON: Monitor	C; NAC;	C; NAC;
MIS: Mismatch	C; NAC;	C; NAC;
CON: Contact	C; NAC;	C; NAC;
FAT: Fast Try	C;RCm; NAC;	C;RCm; NAC;
FAC: Fast Contact	C;RCm; NAC;	C;RCm; NAC;
WRC: Wait_RC	C;RCm; NAC;	C;RCm; NAC;
KON: Konnect	RCh; KON;	C;RCm;DT; NAC;
REK: Re_Konnect	RCh; REK;	C;RCm;DT;IT; NAC;
SOS: Sync_Lost	RCh; SOS;	C;RCm;IT; NAC;
OPE: Operation	RCh; OPE;	C;RCm;DT;T1; TT;
FAI: Failure	---------- ----------	---------- ----------
TT: TFO_Term	NoAc; TT;	NoAc; TT;

Table 10.6-9: Mismatching TFO_TRANS and TFO Frames

Event:	**TFO_TRANS**	**TFO_Frame**	**TFO_Frame**
Number:	37	38	39
Condition: &	DCh!=LCh	Mismatch_1	Mismatch_2
Comment: **State:**	Mismatch of channel type	Mismatch for one or two TFO Frames	Continued Mismatch
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;U;L;T2;B; MIS; HO?	NoAc; FIT; HO? be tolerant	C;U;L;T2;B; MIS; **Typical in HO**
COR: Continuous Retry	C;U;L;T2;B; MIS;	NoAc; COR; Call Forw?	C;U;L;T2;B; MIS;
PER: Periodic Retry	C;F;L;T2;B; MIS;	NoAc; PER; Call Forw?	C;F;L;T2;B; MIS;
MON: Monitor	C;F;L;T2;B; MIS;	NoAc; MON; ins. «;» Call Forw?	C;F;L;T2;B; MIS;
MIS: Mismatch	C;L;T2;B; MIS;	NoAc; MIS; Call Forw?	C;L;T2;B; MIS;
CON: Contact	C;L;T2;B; MIS;	NoAc; CON;	C;L;T2;B; MIS;
FAT: Fast Try	C;L;T2;B;RCm; MIS;	NoAc; FAT;	C;L;T2;B;RCm; MIS;
FAC: Fast Contact	C;L;T2;B;RCm; MIS;	NoAc; FAC;	C;L;T2;B;RCm; MIS;
WRC: Wait_RC	C;RCm;L;T2;B; MIS;	NoAc; WRC;	C;-RCm;L;T2;B; MIS; rem. space
KON: Konnect	C;RCm;DT;L;T2;B; MIS;	NoAc; KON;	C;RCm;DT;L;T2;B; MIS;
REK: Re_Konnect	C;RCm;DT;L;T2;IT;B; MIS;	NoAc; REK;	C;RCm;DT;L;T2;IT;B; MIS;
SOS: Sync_Lost ins. «:»	C;RCm;L;T2;IT;B; MIS;	NoAc; SOS;	C;RCm;L;T2;IT;B; MIS;
OPE: Operation	NoAc; OPE; Ignore?	NoAc; OPE; Hard HO?	C;RCm;DT;L;T2;IT;B; MIS; Hard HO into TFO
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	---------- ----------	---------- ----------	---------- ----------

Table 10.6-10: Local Events, TFO_FILL, TFO_NORMAL

Event:	**New_Local_Codec_List**	**Data_Call**	**TFO_FILL**	**TFO_NORMAL**
Number:	40	41	42	43
Condition: &				
Comment: **State:**	From RAN	In Call Modif. Stop TFO (see TFO_Disable)	Ignore is just Filler	Ignore alternative: Soft Reset
NAC: Not_Active	NoAc; NAC;	NoAc; NAC;	---------- ----------	---------- ----------
WAK: Wakeup	NoAc; WAK;	NoAc; NAC;	---------- ----------	---------- ----------
FIT: First_Try	NoAc; FIT; Update loc. Par.	C;N; NAC;	NoAc; FIT;	NoAc; FIT;
COR: Continuous Retry	NoAc; COR;	C;N; NAC;	NoAc; COR;	NoAc; COR;
PER: Periodic Retry	NoAc; PER;	C;N; NAC;	NoAc; PER;	NoAc; PER;
MON: Monitor	NoAc; MON; ins. «;»	C;N; NAC;	NoAc; MON; ins. «;»	NoAc; MON; ins. «;»
MIS: Mismatch	C;L;T2; MIS; **direct info**	C;N; NAC;	NoAc; MIS;	NoAc; MIS;
CON: Contact	NoAc; CON;	C;N; NAC;	NoAc; CON;	NoAc; CON;
FAT: Fast Try	NoAc; FAT;	C;N;RCm; NAC;	NoAc; FAT;	NoAc; FAT;
FAC: Fast Contact	NoAc; FAC;	C;N;RCm; NAC;	NoAc; FAC;	NoAc; FAC;
WRC: Wait_RC	NoAc; WRC;	C;N; NAC;	NoAc; WRC;	NoAc; WRC;
KON: Konnect	NoAc; KON;	C;DT;N; NAC;	NoAc; KON;	NoAc; KON;
REK: Re_Konnect	NoAc; REK;	C;DT;IT;N; NAC;	NoAc; REK;	NoAc; REK;
SOS: Sync_Lost	NoAc; SOS;	C;IT;N; NAC;	NoAc; SOS;	NoAc; SOS;
OPE: Operation	L;T2; OPE; **direct info**	C;DT;IT;N; NAC;	NoAc; OPE;	NoAc; OPE;
FAI: Failure	NoAc; FAI;	C; NAC; **exit from FAI**	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	NoAc; TT;	IT;N; NAC;	---------- ----------	---------- ----------

Table 10.6-11: Special Events, Timeouts

Event:	**Runout**	**T==0**	**Frame_Sync_Lost**	**Frame_Sync_Lost**	**Mes_Sync_Lost**
Number:	44	45	46	47	48
Condition: &			n<3	n>2 !change! TFO_Disabled	
Comment: **State:**	IPEs may become unsynchronised	Time-Out	start to send SYL already	Stop TFO Frames if 3 Frames missing	
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	U;N; MON; **PSTN Call**	---------- ----------	---------- ----------	---------- ----------	NoAc; FIT;
COR: Continuous Retry	U;L1;T5; PER; **at end of COR**	C;N;REQ; COR; Reset IPEs	---------- ----------	---------- ----------	NoAc; COR;
PER: Periodic Retry	NoAc; PER;	L1;T5; PER; **Periodic Test**	---------- ----------	---------- ----------	NoAc; PER;
MON: Monitor	---------- ----------	C;N; MON;	---------- ----------	---------- ----------	---------- ----------
MIS: Mismatch	NoAc; MIS; **typ Final state**	N;B; MIS; List not Ack_ed!	NoAc; MIS;	NoAc; MIS;	NoAc; MIS;
CON: Contact	REQ; COR; can this occur?	---------- ----------	---------- ----------	---------- ----------	C;REQ; COR;
FAT: Fast Try	REQ;RCm; COR; fast HO failed	---------- ----------	NoAc; FAT; typical in HO	NoAc; FAT; typical in HO	C;REQ;RCm; COR; fast HO failed
FAC: Fast Contact	REQ;RCm; COR; fast HO failed	---------- ----------	NoAc; FAC; typical in HO	NoAc; FAC; typical in HO	C;REQ;RCm; COR; fast HO failed
WRC: Wait_RC	C;RCm; FAI; Missing RC_Ack	C;RCm; FAI; Missing RC_Ack	NoAc; WRC;	IT; WRC;	C;RCm;REQ; COR;
KON: Konnect	NoAc; KON; may happen	C;RCm;DT;N; FAI; Misbehaviour!	---------- ----------	---------- ----------	C;RCm;DT;REQ;T1; COR; after Timeout: N
REK: Re_Konnect	NoAc; REK; may happen	C;RCm;DT;N;IT;B; FAI; Misbehaviour!	---------- ----------	---------- ----------	C;RCm;DT;REQ;IT;B;T1; COR; after Timeout: N
SOS: Sync_Lost	RCm;REQ;IT;B;T1; COR; after Timeout: N	---------- ----------	---------- ----------	NoAc; SOS; wait for Runout	C;RCm;REQ;IT;B;T1; COR; after Timeout: N
OPE: Operation	NoAc; OPE; **typ Final event**	B; OPE; List not Ack_ed!	SYL1; OPE; **1: Alarm, go on**	C;DT;SYL; SOS; **2: Alarm, stop!**	NoAc; OPE; **Typ Final event**
FAI: Failure	NoAc; FAI; typical	---------- ----------	---------- ----------	---------- ----------	NoAc; FAI; don´t trust!
TT: TFO_Term	NoAc; TT;	IT;N: NAC;	NoAc; TT;	IT;N; NAC;	NoAc; TT;

Table 10.6-11b: Special Events, Timeouts (continuation)

Event:	**Frame_Sync_Lost**
Number:	57
Condition: &	n>2 TFO_Enabled
Comment: **State:**	Stop TFO Frames if 3 Frames missing
NAC: Not_Active	---------- ----------
WAK: Wakeup	---------- ----------
FIT: First_Try	---------- ----------
COR: Continuous Retry	---------- ----------
PER: Periodic Retry	---------- ----------
MON: Monitor	---------- ----------
MIS: Mismatch	NoAc; MIS;
CON: Contact	---------- ----------
FAT: Fast Try	NoAc; FAT; typical in HO
FAC: Fast Contact	NoAc; FAC; typical in HO
WRC: Wait_RC	IT; WRC;
KON: Konnect	---------- ----------
REK: Re_Konnect	---------- ----------
SOS: Sync_Lost	NoAc; SOS; wait for Runout
OPE: Operation	C;DT;SYL; SOS; **2: Alarm, stop!**
FAI: Failure	---------- ----------
TT: TFO_Term	C;RCm;B; MON;

Table 10.6-12 Distant Config Frame for 3G systems (TC)

Event:	**Distant_Config**	**Distant_Config**	**Distant_Config**	**Distant_Config**
Number:	49	50	51	52
Condition: &	(NA_TP | A_TP) Con_Req & TC	TM Con_Req & TC	(NA_TP | A_TP) Con_Ack & TC	TM Con_Ack & TC
Comment: **State:**	Config request Matching parameters	Config request TFO Mismatch	Config acknowledgement Matching parameters	Config acknowledgement TFO Mismatch
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;U;DUP;RCi; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;U;DUP;RCi; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;
COR: Continuous Retry	C;U;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;U;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;
PER: Periodic Retry	C;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;
MON: Monitor	C;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;
MIS: Mismatch	C;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;
CON: Contact	C;T;BT;T;T1; KON; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;T;BT;T;T1; KON; Same as 1. TFO_Frame	C;RCm;B; MIS;
FAT: Fast Try	NoAc; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	NoAc; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;
FAC: Fast Contact	C;BT;T;L;T2;AT;B; OPE; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;BT;T;L;T2;AT;B; OPE; Same as 1. TFO_Frame	C;RCm;B; MIS;
WRC: Wait_RC	NoAc; WRC;	C;RCm;B; MIS;	NoAc; WRC;	C;RCm;B; MIS;
KON: Konnect	RCs;CA1;AT;L;T2;B; OPE; Same as 1. TFO_Frame	C;RCm;CA;DT;B;T1; MIS;	RCs;AT;L;T2;B; OPE; Same as 1. TFO_Frame	C;RCm;DT;B;T1; MIS;
REK: Re_Konnect	RCs;CA1;AT;L;T2;B; OPE; Same as 1. TFO_Frame	C;RCm;CA;DT;IT;B;T1; MIS;	RCs;AT;L;T2;B; OPE; Same as 1. TFO_Frame	C;RCm;DT;IT;B;T1; MIS;
SOS: Sync_Lost	C;RCs;CA1;BT;T;L;T2;B; OPE; Same as 1. TFO_Frame	C;RCm;CA;DT;IT;B;T1; MIS;	C;RCs;BT;T;L;T2;B; OPE; Same as 1. TFO_Frame	C;RCm;DT;IT;B;T1; MIS;
OPE: Operation	RCs;CA1; OPE; Same as 1. TFO_Frame	C;RCm;CA;DT;IT;B;T1; MIS;	RCs; OPE; Same as 1. TFO_Frame	C;RCm;DT;IT;B;T1; MIS;
FAI: Failure	---------- ----------	---------- ----------	---------- ----------	---------- ----------
TT: TFO_Term	B; TT;	B; TT;	B; del. «:» ins. «;» TT;	B; TT;

Table 10.6-13 Distant Config Frame for GSM systems (TRAU) and Distant_Disable

Event:	**Distant_Config**	**Distant_Config**	**Distant_Config**	**Distant_Disable**
Number:	53	54	55	56
Condition: &	(NA_TP | A_TP) TRAU	TM Con_req & TRAU	TM Con_Ack & TRAU	
Comment: **State:**	Config req or Config ack Matching parameters	Config request TFO Mismatch	Config acknowledgement TFO Mismatch	Distant side has disabled TFO
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;U;DUP;RCi; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;RCm;B; MIS;	C;RCm;B; MON;
COR: Continuous Retry	C;U;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;RCm;B; MIS;	C;RCm;B; MON;
PER: Periodic Retry	C;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;RCm;B; MIS;	C;RCm;B; MON;
MON: Monitor	C;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;RCm;B; MIS;	C;RCm;B; MON;
MIS: Mismatch	C;DUP; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;RCm;B; MIS;	C;RCm;B; MON;
CON: Contact	C;T;BT;T;T1; KON; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;RCm;B; MIS;	C;RCm;B; MON;
FAT: Fast Try	NoAc; FAT; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;RCm;B; MIS;	C;RCm;B; MON;
FAC: Fast Contact	C;BT;T;L;T2;AT;B; OPE; Same as 1. TFO_Frame	C;RCm;B; MIS;	C;RCm;B; MIS;	C;RCm;B; MON;
WRC: Wait_RC	NoAc; WRC;	C;RCm;B; MIS;	C;RCm;B; MIS;	C;RCm;B; MON;
KON: Konnect	RCs;AT;L;T2;B; OPE; Same as 1. TFO_Frame	C;RCm;CA;DT;B;T1; MIS;	C;RCm;DT;B;T1; MIS;	C;RCm;CA;DT;B;T1; MON;
REK: Re_Konnect	RCs;AT;L;T2;B; OPE; Same as 1. TFO_Frame	C;RCm;CA;DT;IT;B;T1; MIS;	C;RCm;DT;IT;B;T1; MIS;	C;RCm;CA;DT;IT;B;T1; MON;
SOS: Sync_Lost	C;RCs;BT;T;L;T2;B; OPE; Same as 1. TFO_Frame	C;RCm;CA;DT;IT;B;T1; MIS;	C;RCm;DT;IT;B;T1; MIS;	C;RCm;IT;B;T1; MON;
OPE: Operation	RCs; OPE; Same as 1. TFO_Frame	C;RCm;CA;DT;IT;B;T1; MIS;	C;RCm;DT;IT;B;T1; MIS;	C;RCm;CA;DT;IT;B;T1; MON;
FAI: Failure	---------- ----------	---------- ----------	---------- ----------	---------- ----------
TT: TFO_Term	B; TT;	B; TT;	B;IT;N; NAC;	B;IT;N; NAC;

3GPP TSG-SA4 #21
Rennes, France, 13-17 May 2002

S4-020356

CR-Form-v6.1

CHANGE REQUEST

⌘	TS 28.062	CR	027	⌘ rev	1	⌘	Current version:	5.0.0	⌘
	Spec Title:	Inband Tandem Free Operation (TFO) of speech codecs							⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network [X] Core Network ☐

Title:	⌘	**Immediate Codec Type Optimisation**
Source:	⌘	**TSG SA WG4**
Work item code:	⌘	AMRWB
Date:	⌘	2002-06-11
Category:	⌘	**F**
Release:	⌘	**REL-5**

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
REL-4	*(Release 4)*
REL-5	*(Release 5)*

Reason for change:	⌘	Optimised TFO establishment for alternative codecs
Summary of change:	⌘	Addition of new events in TFO protocol and extension of procedures.
Consequences if not approved:	⌘	The default Codec Type Optimisation has to be used causing increased delay and reduced speech quality during TFO establishment.

Clauses affected:	⌘	9, 10, 11, G		
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ O&M Specifications	⌘	
Other comments:	⌘			

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

The first modification is in clause 9

9 TFO State Machine

…

9.3 Contact State

In this state the TFO_Protocol knows that there is a distant TFO Partner, which has sent TFO_REQ. The Codecs do match and the ACSs are compatible. The link from the distant partner is transparent. Now TFO_ACK need to be sent to check the transparency of the link to the distant partner.

After the exchange of TFO_REQ and/or TFO_ACK messages, it may become obvious that a preferred TFO configuration is possible when changing the codec type at the local and/or the distant side. For example, this is the case when both sides support AMR-WB but one of both sides is currently using AMR-NB. In this case, the TFO protocol stays in the Contact state and performs an Immediate Codec Type Optimization (see 11.7). After the codecs have been changed, the normal protocol flow continues.

As soon as a TFO_ACK or TFO_TRANS from a distant partner has been received, the TC knows that the links in both directions are digitally transparent. In the case of a Non_AMR Codec Type the TC sends TFO_TRANS to bypass the IPEs and starts sending TFO Frames, and the TFO_Protocol transits into Konnect State. In the case of an AMR or AMR-WB Codec Type the TC sends a Rate Control Command downlink to its BTS/RNC in order to steer the uplink Codec Mode down to the TFO_Setup_Mode for a safe TFO Setup. Additionally, TFO_ACK is sent to the distant TFO Partner and the TFO_Protocol transits into the Wait_RC State.

10 Detailed Description of the TFO Protocol

…

10.2.1 Conditions for TFO_REQ, TFO_ACK, TFO_REQ_L, TFO_ACK_L, New_Local_Codec, New_Local_Config, Distant Config

In the context of TFO_REQ, TFO_ACK, TFO_REQ_L, TFO_ACK_L, New_Local_Codec, New_Local_Config, Distant_Config the following conditions are used:

A_TP (AMR_TFO_Possible)
This condition is fulfilled if an AMR NB or AMR-WB codec type is used and the TFO decision algorithms results in an immediate TFO situation. According to clause 11.2.3 these immediate TFO situations are:

- Immediate TFO with LACS == DACS
- Immediate TFO with FR – HR – Matching
- Immediate TFO with IACS == OACS
- Immediate TFO with the IACS is a subset of the OACS

NA_TP (Non_AMR_TFO_Possible)
This condition is fulfilled if a non-AMR codec type is used and the distant used codec type is equal to the local used codec type (Duc==Luc).

TM (TFO_Mismatch)

This condition is fulfilled if the TFO decision algorithm does not result in an immediate TFO situation. This is the case in the following situations:

- The local and distant side use incompatible codec types.
- Both sides use compatible AMR or compatible AMR-WB codec types and the OACS doesn't exist or the OACS isn't acceptable (Codec Mismatch Resolution has to be invoked).
- Both sides use compatible AMR or compatible AMR-WB codec types and the OACS is acceptable for TFO, but first the ACS has to be changed to the OACS.

ICO (Immediate_Codec_Type_Optimisation)

This condition is fulfilled if

- both sides indicate a TFO version greater than or equal to 5 and
- the available information on alternative codec types indicates that a change of the local and/or distant codec type results in a TFO configuration with a higher preference level.

The condition is re-evaluated whenever new information on alternative codec types becomes available.

10.4 Detailed Description of the Events

Table 10.4-1 lists all events of the Protocol Tables.

Table 10.4-1: Events of the State Machine Description

#	Event	Description
1	TFO_Enable	The event TFO_Enable occurs when all TFO parameters get available in the transcoder and the controlling entity enables TFO. In GSM, it means that the TFOE bit of AMR or AMR-WB TRAU Frames toggles from '0' to '1'. Enabling TFO might involve a proprietary process not further addressed in the present document.
2	New_Speech_Call	This event occurs when a new speech call is set-up or the TRAU/TC is re-initialised (e.g. after a handover failure). In GSM, this means that the transcoder is initialised by the BTS by two consecutive TRAU frames with identical codec types (GSM_FR, GSM_HR, GSM_EFR) or by a config frame (AMR or AMR-WB codec types). In 3G, this means that the Iu User Plan is initialised.
3	TFO_Disable	The event TFO_Disable occurs when TFO is disabled by the controlling entity. In GSM, the TFO_Disable event is also controlled by the TFOE bit of AMR or AMR-WB TRAU Frames.
4	TRAU_Idle	This event occurs when the transcoder is set into idle mode.
5	PCM_Non_Idle	The event PCM_Non_Idle occurs if more than one PCM samples are received that are different to PCM_Idle.
12	TFO_Frame and Match_1	This event means that a valid TFO Frame was received by the transcoder and the condition Match_1 is fulfilled.
17	TFO_Frame and Match_2	This event means that a valid TFO Frame was received by the transcoder and the condition Match_2 is fulfilled.
38	TFO_Frame and Mismatch_1	This event means that a valid TFO Frame was received by the transcoder and the condition Mismatch_1 is fulfilled.
39	TFO_Frame and Mismatch_2	This event means that a valid TFO Frame was received by the transcoder and the condition Mismatch_2 is fulfilled.
13	New_Local_Codec and (NA_TP | A_TP) and ICO==0	This event occurs when the local used codec type changes, ~~and~~ either the condition NA_TP or the condition A_TP is fulfilled, and Immediate Codec Type Optimisation is not performed.
15	New_Local_Codec and TM and ICO==0	This event occurs when the local used codec type changes, ~~and~~ the condition TM is fulfilled, and Immediate Codec Type Optimisation is not performed.
14	New_Local_Config and (NA_TP | A_TP) and ICO==0	This event occurs when an AMR or AMR-WB codec type is used, ~~and~~ the local codec configuration changes, ~~and~~ either the condition A_TP or NA_TP is fulfilled, and Immediate Codec Type Optimisation is not performed.
16	New_Local_Config and TM and ICO==0	This event occurs when an AMR or AMR-WB codec type is used, ~~and~~ the local codec configuration changes, ~~and~~ the condition TM is fulfilled, and Immediate Codec Type Optimisation is not performed.
32	RC_ack	This event (rate control acknowledgement) occurs when an acknowledgement to the RCi action is received from the BTS/RNC indicating that the rate control command was understood (TFO_Soon acknowledgement in GSM, Rate_Ack in UMTS).
40	New_Local_Codec_List	This event occurs when the local codec list changes.
41	Data_Call	This event is only relevant for GSM systems. It occurs when the transcoder is informed that a Data Call is set-up.
44	Runout	The event Runout occurs when the last TFO message has been taken from the Transmit Queue and the last 10 bits are going to be sent. So there is still some time for TFO_Protocol to react and place a further TFO Message in the Transmit Queue, which then shall be transmitted without gap to the messages before.
45	T==0	This event occurs when a time-out has been reached.
46	Frame_Sync_Lost and n<3	This event occurs when the TFO frame synchronisation is lost for the first or the second time. For further details see Annex C.
47	Frame_Sync_Lost and n>2 and TFO_Disabled	This event occurs when the TFO frame synchronisation is lost for more than two times and TFO has been disabled. For further details see Annex C.
57	Frame_Sync_Lost and n>2 and TFO_Enabled	This event occurs when the TFO frame synchronisation is lost for more than two times and TFO is still enabled. For further details see Annex C.
48	Mes_Sync_Lost	This event corresponds to a loss of TFO message synchronisation. For further details see Annex C.

#	Event	Description
35	Handover_Soon and (NA_TP | A_TP)	This event occurs when the TRAU/TC is informed that a local hand-over will soon take place and either the condition NA_TP or the condition A_TP is fulfilled.
36	Handover_Soon and TM	This event occurs when the TRAU/TC is informed that a local hand-over will soon take place and the condition TM is fulfilled.
6	TFO_REQ and (NA_TP | A_TP) and Dsig==Lsig and Dsig!=Old_Sig	This event occurs when a TFO_REQ message is received, either the condition NA_TP or the condition A_TP is fulfilled and the distant signature is equal to the local signature but different from the old (local) signature.
7	TFO_REQ and (NA_TP | A_TP) and Dsig==Old_Sig	This event occurs when a TFO_REQ message is received, the condition NA_TP or A_TP is fulfilled, and the distant signature is equal to the old signature.
8	TFO_REQ and (NA_TP | A_TP) and Dsig!=Lsig and Dsig!=Old_Sig and ICO==0	This event occurs when a TFO_REQ message is received, either the condition NA_TP or the condition A_TP is fulfilled, ~~and~~ the distant signature is different from the local signature and old (local) signature, and Immediate Codec Type Optimisation is not performed.
24	TFO_REQ and TM and Dsig==Lsig	This event occurs when a TFO_REQ message is received, the condition TM is fulfilled, and the distant and the local signatures are equal.
25	TFO_REQ and TM and Dsig!=Lsig and ICO==0	This event occurs when a TFO_REQ message is received, the condition TM is fulfilled, ~~and~~ the distant signature is different from the local signature, and Immediate Codec Type Optimisation is not performed.
9	TFO_ACK and NA_TP and Dsig==Lsig and ICO==0	This event occurs when a TFO_ACK message is received, the condition NA_TP is fulfilled, ~~and~~ the local and distant signatures are equal, and Immediate Codec Type Optimisation is not performed.
10	TFO_ACK and (NA_TP | A_TP) and Dsig!=Lsig	This event occurs when a TFO_ACK message is received, either the condition NA_TP or the condition A_TP is fulfilled, and the distant signature is different from the local signature.
26	TFO_ACK and TM and ICO==0 (Dsig==?)	This event occurs when a TFO_ACK message is received, ~~and~~ the condition TM is fulfilled, and Immediate Codec Type Optimisation is not performed. The distant signature is ignored for this event.
31	TFO_ACK and A_TP and Dsig==Lsig and ICO==0	This event occurs when a TFO_ACK message is received, the condition A_TP is fulfilled, ~~and~~ the distant signature is equal to the local signature, and Immediate Codec Type Optimisation is not performed.
11	TFO_TRANS and Luc != AMR and DCh==LCh	This event occurs when a TFO_TRANS message is received when a non-AMR codec type is used on the local side and the distant and local channel types do match.
30	TFO_TRANS and Luc == AMR and DCh==LCh	This event occurs when a TFO_TRANS message is received while a AMR or AMR-WB codec type is used and the distant and local channel types do match.
37	TFO_TRANS and DCh!=LCh	This event occurs when a TFO_TRANS message is received and a channel mismatch occurs.
18	TFO_SYL	This event occurs when a TFO_SYL message is received.
19	TFO_DUP	This event occurs when a TFO_DUP message is received.
20	TFO_REQ_L and (NA_TP | A_TP) and Dsig==Lsig	This event occurs when a TFO_REQ_L message is received, either the condition NA_TP or the condition A_TP is fulfilled, and the local signature is equal to the distant signature.
21	TFO_REQ_L and (NA_TP | A_TP) and Dsig!=Lsig	This event occurs when a TFO_REQ_L message is received, either the condition NA_TP or the condition A_TP is fulfilled, and the local and distant signatures are different.
27	TFO_REQ_L and TM and Dsig==Lsig	This event occurs when a TFO_REQ_L message is received, the condition TM is fulfilled, and the local and distant signatures are equal.
28	TFO_REQ_L and TM and Dsig!=Lsig	This event occurs when a TFO_REQ_L message is received, the condition TM is fulfilled and the local and distant signatures are different.
22	TFO_ACK_L and (NA_TP | A_TP) and Dsig==Lsig	This event occurs when a TFO_ACK_L message is received, either the condition NA_TP or the condition A_TP is fulfilled, and the local signature is equal to the distant signature.
23	TFO_ACK_L and (NA_TP | A_TP) and Dsig!=Lsig	This event occurs when a TFO_ACK_L message is received, either the condition NA_TP or the condition A_TP is fulfilled, and the local and distant signatures are different.

#	Event	Description
29	TFO_ACK_L and TM and Dsig==?	This event occurs when a TFO_ACK_L message is received and the condition TM is fulfilled. The distant signature is not relevant for this event.
42	TFO_FILL	This event occurs when a TFO_FILL message is received.
43	TFO_NORMAL	This event occurs when a TFO_NORMAL message is received.
49	Distant_Config and (NA_TP | A_TP) and Con_Req & TC	This event occurs when a 3G system (TC) receives a config request from the distant TRAU/TC, the TFO_enable bit is set, and the parameters of this config frame are compatible with the local parameters so that TFO is possible.
50	Distant_Config and TM and Con_Req & TC	This event occurs when 3G system (TC) receives a config request from the distant TRAU/TC, the TFO_enable bit is set, and the parameters of this config frame do not match with the local parameters so that TFO is not possible.
51	Distant_Config and (NA_TP | A_TP) and Con_Ack & TC	This event occurs when a 3G system (TC) receives a config acknowledgement from the distant TRAU/TC, the TFO_enable bit is set, and the parameters of this config frame are compatible with the local parameters so that TFO is possible. This event does not occur when an acknowledgement for a config request indicating Handover_Soon is received.
52	Distant_Config and TM and Con_Ack & TC	This event occurs when 3G system (TC) receives a config acknowledgement from the distant TRAU/TC, the TFO_enable bit is set, and the parameters of this config frame do not match with the local parameters so that TFO is not possible. This event does not occur when an acknowledgement for a config request indicating Handover_Soon is received.
53	Distant_Config and (NA_TP | A_TP) and TRAU	This event occurs when a 2G system (TRAU) receives a config frame (config request or config acknowledgement) from the distant TRAU/TC, the TFO_enable bit is set, and the parameters of this config frame are compatible with the local parameters so that TFO is possible. This event does not occur when an acknowledgement for a config request indicating Handover_Soon is received.
54	Distant_Config and TM and Con_Req & TRAU	This event occurs when a 2G system receives a config request from the distant TRAU/TC, the TFO_enable bit is set, and the parameters of this config frame do not match with the local parameters so that TFO is not possible.
55	Distant_Config and TM and Con_Ack & TRAU	This event occurs when a 2G system receives a config acknowledgement from the distant TRAU/TC, the TFO_enable bit is set, and the parameters of this config frame do not match with the local parameters so that TFO is not possible. This event does not occur when an acknowledgement for a config request indicating Handover_Soon is received.
56	Distant_Disable	This event occurs when a config frame (config request) with a TFO_Enable bit set to zero is received from the distant TRAU/TC, i.e. when the distant side is going to disable TFO.
58	TFO_REQ and Dsig != Lsig and ICO==1	This event occures when a TFO_REQ message is received, the distant signature is different from the local signature, and Immediate Codec Type Optimisation is performed.
59	TFO_ACK and Dsig==Lsig and ICO==1	This event occures when a TFO_ACK message is received, the distant signature is equal to the local signature, and Immediate Codec Type Optimisation is performed
60	New_Local_Codec and ICO==1	This event occurs when the local used codec type changes and Immediate Codec Type Optimisation is performed.
61	New_Local_Config and ICO==1	This event occurs the local codec configuration changes and Immediate Codec Type Optimisation is performed.

…

10.6 Protocol Tables

Table 10.6-3: Most Important Cases, Especially at Call Set-up

Event:	TFO_REQ	TFO_ACK	TFO_ACK	TFO_TRANS	TFO_Frame
Number:	8	9	10	11	12
Condition: & & &	(NA_TP | A_TP) Dsig!=Lsig Dsig!=Old_Sig ICO==0	NA_TP Dsig==Lsig	(NA_TP | A_TP) Dsig!=Lsig	Luc != AMR DCh==LCh	Match_1
Comment: **State:**	Distant REQ Good Signature	Distant ACK Good Signature	Wrong Response Handover?	similar to ACK As response to loc ACK_?	First or second TFO Frame
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;U;ACK; CON; **Typical**	C;U;T;BT;T;T1; KON; **Typical; IPEs!**	C;REQ; FIT;	NoAc; FIT; Wait for Frame	C;U;DUP;RCi; FAT; **1: HO**
COR: Continuous Retry	C;U;ACK; CON; Typical	C;U;T;BT;T;T1; KON; **Typical; IPEs!**	C;REQ; COR;	NoAc; COR; Wait for Frames	C;U;DUP; FAT; 1: Call is back?
PER: Periodic Retry	C;F;ACK; CON; OK, Contact is back	C;F;S;REQ; COR; Rare case, test	C;F;REQ; COR;	NoAc; PER; Wait for Frames	C;DUP; FAT; 1: Call is back?
MON: Monitor	C;F;REQ; FIT; IPEs?	C;F;S;REQ; FIT; Rare case, test	C;F;REQ; FIT;	NoAc; MON Wait for Frames	C;DUP; FAT; 1: Call is back?
MIS: Mismatch	C;F;ACK; CON; Mismatch resolved	C;F;S;REQ; COR; Rare case, test	C;F;REQ; COR;	NoAc; MIS; Wait for Frames	C;DUP; FAT; 1: Call is back?
CON: Contact	C;ACK; CON; **Typical: wait**	C;T;BT;T;T1; KON; **Typical: yes!**	C;REQ; COR;	C;T;BT;T;T1; KON; **yes! Fast way**	C;T;BT;T;T1; KON; Missed TRANS?
FAT: Fast Try	C;REQ;RCm; COR; Safe way	C;REQ;RCm; COR; Safe way	C;REQ;RCm; COR; Safe way	NoAc; FAC; Wait for Frames	NoAc; FAT; **2: Typ. Loc HO**
FAC: Fast Contact	C;REQ;RCm; COR; Safe way	C;REQ;RCm; COR; Safe way	C;REQ;RCm; COR; Safe way	NoAc; FAC; Wait for Frames	C;BT;T;L;T2;AT;B; OPE; **5: Typ. Loc HO**
WRC: Wait_RC	C;RCm;REQ;T1; COR;	---------- ----------	C;RCm;REQ; COR;	---------- ----------	AT WRC;
KON: Konnect	C;RCm;DT;REQ;T1; COR; IPEs transparent!	NoAc; KON; **Typical: wait**	NoAc; KON;	NoAc; KON; **Typical: wait**	RCs;AT;L;T2;B; OPE; **Typ: call set-up**
REK: Re_Konnect	C;RCm;DT;REQ;IT;B;T1; COR; IPEs transparent!	C;DT;REQ;IT;B;T1; COR;	C;DT;RCm;REQ;IT;B; T1 COR;	NoAc; REK; Wait for Frames	AT;L;T2;B; OPE; **5: Typ. Dis HO**
SOS: Sync_Lost	C;RCm;IT;REQ;B;T1; COR; Contact is back	C;IT;REQ;B;T1; COR; Contact is back	C;IT;RCm;REQ;B;T1; COR; Contact is back	NoAc; SOS; Wait for Frames	C;BT;T;L;T2;B; OPE; short Interrupt?
OPE: Operation	---------- ----------	---------- ----------	---------- ----------	NoAc; OPE; Typical in HO	NoAc; OPE; **Main! TFO!**
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------

Table 10.6-4: In Call Modification and Handover

Event: or **New_Local_Codec New_Local_Config**	**New_Local_Codec New_Local_Config**	**TFO_Frame**	**TFO_SYL**	**TFO_DUP**
Number: 13, 14	15, 16	17	18	19
Condition: (NA_TP | A_TP) & ICO==0	TM ICO==0	Match_2		
Comment: In Call Modif. Mismatch resolv	In Call Modif. Mismatch occurs	Three or more TFO Frames	The dist TC lost sync in OPE	The dist TC recognised HO Identical #17
State:				
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	NoAc; WAK;	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;REQ; FIT; Restart	---------- ----------	NoAc; FIT; HO? Ignore	NoAc; FIT; HO? Ignore
COR: Continuous Retry	C;REQ; COR;	---------- ----------	NoAc; COR; Ignore	NoAc; COR; Ignore
PER: Periodic Retry	L1;T5; PER;	---------- ----------	C;F;REQ; COR; Rare case, test	C;F;REQ; COR; Rare case, test
MON: Monitor	NoAc; MON	---------- ----------	C;F;REQ; FIT; Rare case, test	C;F;REQ; FIT; Rare case, test
MIS: Mismatch	C;F;REQ; COR; **Mismatch Res.**	C;L;T2;B; MIS; **Direct info**	---------- ----------	C;F;REQ; COR; Rare case, test
CON: Contact	C;REQ; COR;	C;L;T2;B; MIS;	---------- ----------	C;F;REQ; COR; Rare case, test
FAT: Fast Try	NoAc; FAT;	C;L;T2;B;RCm; MIS;	NoAc; FAC; **3: Typ. Loc HO**	C;F;REQ;RCm; COR; Rare case, test
FAC: Fast Contact	NoAc; FAC;	C;L;T2;B;RCm; MIS;	C;BT;T;L;T2;AT;B;RCs; OPE; assume matching ACS	NoAc; FAC; **4: Typ Loc HO**	C;F;REQ;RCm; COR; rare case, test
WRC: Wait_RC	C;RCm;REQ; COR;	C;RCm;L;T2;B; MIS;	NoAc; WRC;	NoAc; WRC;	NoAc; WRC;
KON: Konnect	C;RCm;DT;REQ; COR;	C;RCm;DT;L;T2;B; MIS;	RCs;AT;L;T2;B; OPE;	NoAc; KON; Wait, short int?	NoAc; KON; Other TC?
REK: Re_Konnect	C;RCm;DT;IT;REQ; COR;	C;RCm;DT;IT;L;T2;B; MIS;	---------- ----------	C;DT;SYL; SOS; IPEs not transp?	NoAc; REK; **4: Typ. Dist HO**
SOS: Sync_Lost	C;RCm;IT;REQ; COR;	C;RCm;IT;L;T2;B; MIS;	---------- ----------	NoAc; SOS; Short Interrupt.?	C;BT;T;T1; REK; **3: typ Dis HO**
OPE: Operation	RCs;L;T2; OPE;	C;RCm;DT;IT;L;T2;B; MIS;	NoAc; OPE; Main! TFO!	NoAc; OPE; Short interrupt?	NoAc; OPE; Typical
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	C;F;REQ; COR;	NoAc; TT;	NoAc; TT;	IT;N; NAC;	NoAc; TT;

Table 10.6-6: TFO Messages with mismatching Codec Type / Configuration

Event:	**TFO_REQ**	**TFO_REQ**	**TFO_ACK**	**TFO_REQ_L**	**TFO_REQ_L**	**TFO_ACK_L**
Number:	24	25	26	27	28	29
Condition: & &	TM Dsig==Lsig	TM Dsig!=Lsig ICO==0	TM Dsig=? ICO==0	TM Dsig==Lsig	TM Dsig!=Lsig	TM Dsig==?
Comment: **State:**	Mismatch Wrong Sig, HO?	Mismatch Good Sig	Mismatch w/wo HO identical #8	Mismatch Codec_List Wrong Sig, HO?	Mismatch Codec_List Identical #20	Mismatch Codec_List Identical #19
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;S;L;T2;B; MIS; Rare	C;U;L;T2;B; MIS; **Typical: Setup**	C;U;L;T2;B; MIS; HO?	C;S;LA;B; MIS; rare	C;U;LA;B; MIS; **Typical: Setup**	C;U;LA;B; MIS; HO?
COR: Continuous Retry	C;S;L;T2;B; MIS;	C;U;L;T2;B; MIS;	C;U;L;T2;B; MIS;	C;S;LA;B; MIS;	C;U;LA;B; MIS;	C;U;LA;B; MIS;
PER: Periodic Retry	C;F;S;L;T2;B; MIS;	C;F;L;T2;B; MIS;	C;F;L;T2;B; MIS;	C;F;S;LA;B; MIS;	C;F;LA;B; MIS;	C;F;LA;B; MIS;
MON: Monitor	C;F;S;L;T2;B; MIS;	C;F;L;T2;B; MIS;	C;F;L;T2;B; MIS;	C;F;S;LA;B; MIS;	C;F;LA;B; MIS;	C;F;LA;B; MIS;
MIS: Mismatch	C;S;L;T2;B; MIS;	C;L;T2;B; MIS;	C;L;T2;B; MIS;	C;S;LA;B; MIS;	C;LA;B; MIS; **Terminate Prot.**	C;LA;B; MIS; **Terminate Prot.**
CON: Contact	C;S;L;T2;B; MIS;	C;L;T2;B; MIS;	C;L;T2;B; MIS;	C;S;LA;B; MIS;	C;LA;B; MIS;	C;LA;B; MIS;
FAT: Fast Try	C;S;L;T2;B;RCm; MIS;	C;L;T2;B;RCm; MIS;	C;L;T2;B;RCm; MIS;	C;S;LA;B;RCm; MIS;	C;LA;B;RCm; MIS;	C;LA;B;RCm; MIS;
FAC: Fast Contact	C;S;L;T2;B;RCm; MIS;	C;L;T2;B;RCm; MIS;	C;L;T2;B;RCm; MIS;	C;S;LA;B;RCm; MIS;	C;LA;B;RCm; MIS;	C;LA;B;RCm; MIS;
WRC: Wait_RC	C;S;RCm;L;T2;B; MIS;	C; RCm;L;T2;B; MIS;	C; RCm;L;T2;B; MIS;	C;S; RCm;LA;B; MIS;	C; RCm;LA;B; MIS;	C; RCm;LA;B; MIS;
KON: Konnect	C;RCm;DT;S;L;T2; B; MIS;	C;RCm;DT;L;T2; B; MIS;	C;RCm;DT;L;T2; B; MIS;	C;RCm;DT;S;LA; B; MIS;	C;RCm;DT;LA;B; MIS;	C;RCm;DT;LA;B; MIS;
REK: Re_Konnect	C;RCm;DT;S;L;T2; IT;B; MIS;	C;RCm;DT;L;T2; IT;B; MIS;	C;RCm;DT;L;T2; IT;B; MIS;	C;RCm;DT;S;LA; IT;B; MIS;	C;RCm;DT;LA;IT ;B; MIS;	C;RCm;DT;LA;IT; B; MIS;
SOS: Sync_Lost	C;RCm;S;L;T2;IT; B; MIS;	C;RCm;L;T2;IT; B; MIS;	C;RCm;L;T2;IT; B; MIS;	C;RCm;S;LA;IT; B; MIS;	C;RCm;LA;IT;B; MIS; **In_Call_Mod**	C;RCm;LA;IT;B; MIS;
OPE: Operation	---------- ----------	---------- ----------	---------- ----------	NoAc; OPE; Trans Error?	NoAc; OPE; Trans Error?	---------- ----------
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	---------- ----------	---------- ----------	---------- ----------	---------- ----------	C;B; TT;	C;B; TT;

Table 10.6-7 AMR and AMR-WB Cases: TFO_TRANS, TFO_ACK, RC_ack

Event:	**TFO_TRANS**	**TFO_ACK**	**RC_ack**
Number:	30	31	32
Condition: & &	Luc == AMR DCh==LCh	A_TP Dsig==Lsig ICO==0	
Comment: **State:**		Good Sig Immediate TFO possible	BTS has steered the mode.
NAC: Not_Active	---------- ----------	---------- ----------	NoAc; NAC;
WAK: Wakeup	---------- ----------	---------- ----------	NoAc; WAK;
FIT: First_Try	NoAc; FIT; Wait for Frame	C;U;RCi;ACK;T1; WRC; **Typical;**	NoAc; FIT;
COR: Continuous Retry	NoAc; COR; Wait for Frames	C;U;RCi;ACK;T1; WRC; **Typical**	NoAc; COR;
PER: Periodic Retry	NoAc; PER; Wait for Frames	C;F;S;REQ; COR; Rare case, test	NoAc; PER;
MON: Monitor	NoAc; MON Wait for Frames	C;F;S;REQ; FIT; Rare case, test	NoAc; MON;
MIS: Mismatch	NoAc; MIS; Wait for Frames	C;F;S;REQ; COR; Rare case, test	NoAc; MIS;
CON: Contact	C;RCi;ACK;T1; WRC; Missed Ack	C;RCi;ACK;T1; WRC; **Typical**	NoAc; CON;
FAT: Fast Try	NoAc; FAC; Wait for Frames	C;REQ;RCm; COR; Safe way	NoAc; FAT;
FAC: Fast Contact	NoAc; FAC; Wait for Frames	C;REQ;RCm; COR; Safe way	NoAc; FAC;
WRC: Wait_RC	NoAc; WRC;	NoAc; WRC;	C; T;BT;T;T1; KON; **Typical**
KON: Konnect	NoAc; KON; **Typical: wait**	NoAc; KON; **Typical: wait**	NoAc; KON;
REK: Re_Konnect	NoAc; REK; Wait for Frames	C;DT;REQ;IT;B;T1 COR;	NoAc; REK;
SOS: Sync_Lost	NoAc; SOS; Wait for Frames	C;IT;REQ;B;T1 COR; Contact is back	NoAc; SOS;
OPE: Operation	NoAc; OPE; Typical in HO	---------- ----------	NoAc; OPE;
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT: TFO_Term	---------- ----------	---------- ----------	NoAc; TT;

Table 10.6-14 Immediate Codec Type Optimisation

Event:	**TFO_REQ**	**TFO_ACK**	**New_Local_Codec** **New_Local_Config**
Number:	58	59	60, 61
Condition: &	Lsig != Dsig ICO==1	Lsig == Dsig ICO==1	ICO==1
Comment: **State:**	Good signature, Immediate Codec Opt.	Good signature, Immediate Codec Opt.	New Config, Immediate Codec Opt.
NAC: Not_Active	---------- ----------	---------- ----------	---------- ----------
WAK: Wakeup	---------- ----------	---------- ----------	---------- ----------
FIT: First_Try	C;U;ACK;B; CON; **enter ICO**	C;U;ACK;B; CON; **enter ICO**	---------- ----------
COR: Continuous Retry	C;U;ACK;B; CON;	C;U;ACK;B; CON;	C;U;ACK;B; CON;
PER: Periodic Retry	C;U;ACK;B; CON;	C;U;ACK;B; CON;	C;U;ACK;B; CON;
MON: Monitor	C;U;ACK;B; CON;	C;U;ACK;B; CON;	C;U;ACK;B; CON;
MIS: Mismatch	C;U;ACK;B; CON;	C;U;ACK;B; CON;	C;U;ACK;B; CON;
CON: Contact	C;ACK; CON; **wait for HO**	NoAc; CON; **wait for HO or Runout**	C;ACK;B; CON;
FAT: Fast Try	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;
FAC: Fast Contact	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;
WRC Wait_RC	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;
KON: Konnect	C;ACK;RCm;B;DT; CON;	C;ACK;RCm;B;DT; CON;	C;ACK;RCm;B;DT; CON;
REK: Re_Konnect	C;ACK;RCm;B;DT;IT; CON;	C;ACK;RCm;B;DT;IT; CON;	C;ACK;RCm;B;DT;IT; CON;
SOS: Sync_Lost	C;ACK;RCm;B;IT; CON;	C;ACK;RCm;B;IT; CON;	C;ACK;RCm;B;IT; CON;
OPE: Operation	C;ACK;RCm;B;DT;IT; CON;	---------- ----------	C;ACK;RCm;B;DT;IT; CON;
FAI: Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
TT : TFO_Term	---------- ----------	---------- ----------	---------- ----------

The next modification is in clause 11

11.7 Immediate Codec Type Optimisation

The Codec Type Optimisation described in the previous section is performed after the exchange of TFO_REQ_L and TFO_ACK_L messages. Because these messages are exchanged in a late phase of the protocol and may require significant time for transmission, the optimisation may be delayed by a significant amount of time. Furthermore, if TFO was already established before optimisation, a switch to the preferred codec type may disturb the ongoing speech call. To avoid these drawbacks, the codec type optimisation can also be performed immediately during TFO establishment, i.e., in a very early stage of the TFO protocol. This option for TFO establishment is termed “Immediate Codec Type Optimisation” and is explained in the following.

The objective of the Immediate Codec Type Optimisation is to switch the codec type at the local and/or the distant side if this results in a preferred TFO configuration. The required information to decide if Immediate Codec Type Optimization shall be performed is included in the TFO_REQ and TFO_ACK messages by means of the TFO_Version Extension_Block (see Clause 7.4.5). This information is equivalent to the Codec_List included in TFO_REQ_L and TFO_ACK_L messages, however, signalled in a different way. If a preferred TFO configuration becomes possible by changing the local and/or the distant codec type, both sides remain in the Contact state as long as the Immediate Codec Type Optimisation is being performed, i.e., until the local and/or the distant side has/have changed the codec type. After the switch, the TFO protocol continues as usual.

Immediate Codec Type Optimisation becomes only effective in TFO version 5 or higher. If either the local or the distant side is using a lower version, no Immediate Codec Type Optimisation is used. Hence, the protocol is compatible with older versions that do not include Immediate Codec Type Optimisation. Note that a switch to a different codec type is always possible using the normal Codec Type Optimisation in the Mismatch state.

The procedure and preference list used for finding the optimal configuration is exactly identical to Clause 11.6. The only difference is that the required information (active codec, codec list, attributes, …) is obtained from TFO_REQ and TFO_ACK messages instead of TFO_REQ_L and TFO_ACK_L messages. Furthermore, the change of codec type is performed in the Contact state instead of the Mismatch or Operation state.

The next modification is in Annex G

G.9 Immediate Codec Type Optimization

The following protocol flow shows an example for Immediate Codec Type Optimation. Both sides start with AMR-NB, but indicate that AMR-WB is also supported. In this case no immediate TFO Setup in AMR-NB is performed because both sides can use better Codec Types and Configurations. No additional optimisation phase is necessary after AMR-WB TFO Setup.

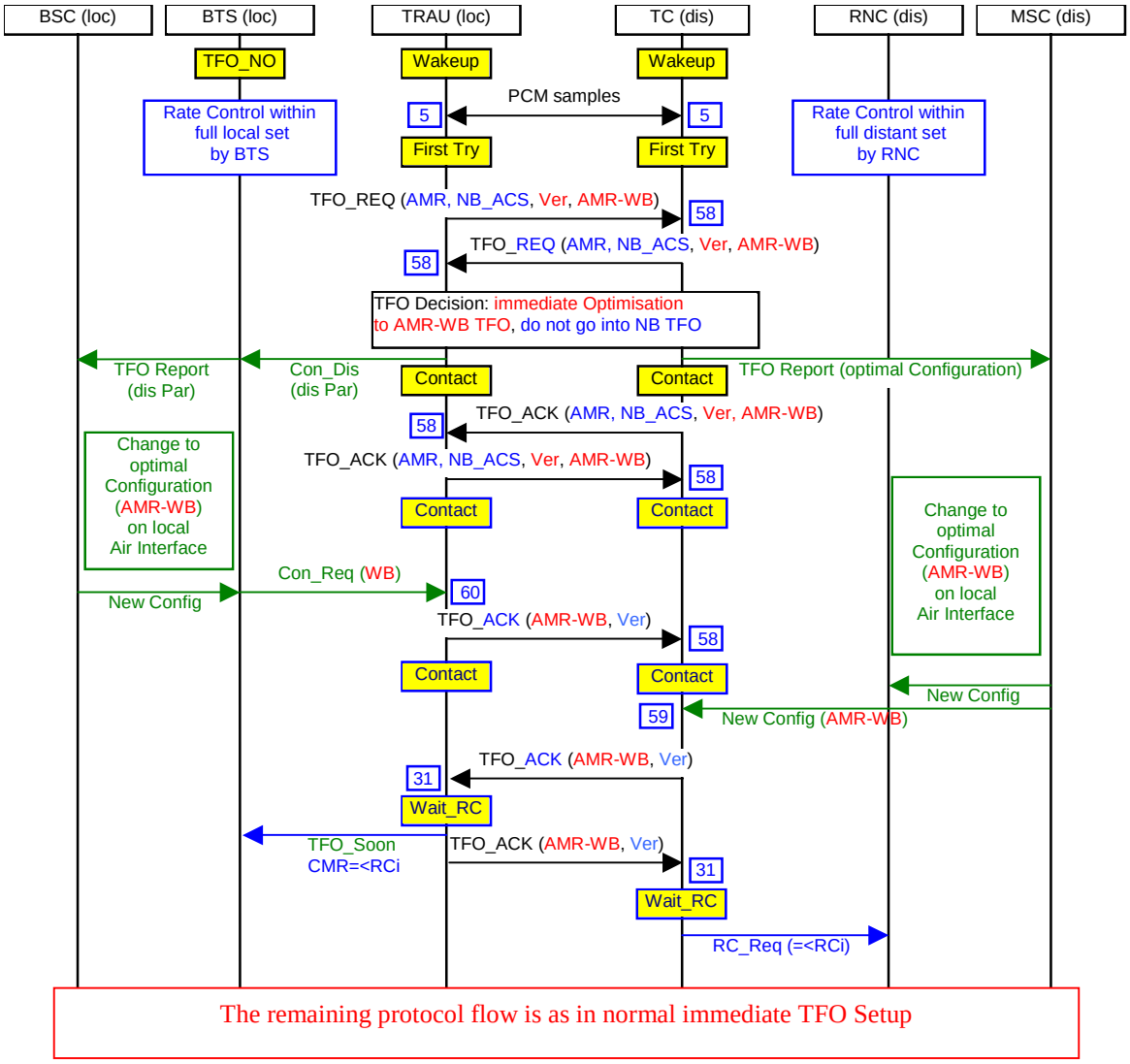


Figure G.9-1: Immediate Codec Type Optimisation for AMR-WB with AMR-NB at call setup

Note: The TFO protocol is kept in the Contact state on both sides as long as contact to the distant side exists and the configurations (local and distant) indicate that TFO setup is possible with a preferred configuration (in this case AMR-WB). The numbers indicate the event number as listed in Table 10.4-1.