

# ETSI TS 128 062 V5.5.0 (2004-12)

*Technical Specification*

**Digital cellular telecommunications system (Phase 2+);  
Universal Mobile Telecommunications System (UMTS);  
Inband Tandem Free Operation (TFO) of speech codecs;  
Service description;  
Stage 3  
(3GPP TS 28.062 version 5.5.0 Release 5)**



---

Reference

RTS/TSGS-0428062v550

---

Keywords

GSM, UMTS

**ETSI**

650 Route des Lucioles  
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C  
Association à but non lucratif enregistrée à la  
Sous-Préfecture de Grasse (06) N° 7803/88

---

**Important notice**

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

[http://portal.etsi.org/chaicor/ETSI\\_support.asp](http://portal.etsi.org/chaicor/ETSI_support.asp)

---

**Copyright Notification**

No part may be reproduced except as authorized by written permission.  
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2004.  
All rights reserved.

DECT™, PLUGTESTS™ and UMTS™ are Trade Marks of ETSI registered for the benefit of its Members.  
TIPHON™ and the TIPHON logo are Trade Marks currently being registered by ETSI for the benefit of its Members.  
3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

---

## Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<http://webapp.etsi.org/IPR/home.asp>).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

---

## Foreword

This Technical Specification (TS) has been produced by the ETSI 3<sup>rd</sup> Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under <http://webapp.etsi.org/key/queryform.asp>.

# Contents

Intellectual Property Rights .....	2
Foreword.....	2
Foreword.....	9
1 Scope .....	10
2 References .....	10
3 Definitions and abbreviations.....	11
3.1 Definitions .....	11
3.2 Abbreviations .....	12
4 General Description.....	14
4.1 Background Information .....	14
4.2 Principle of TFO Operation.....	15
4.2.1 Principle for TFO Operation for Narrow-Band speech codec types .....	15
4.2.2 Principle for TFO Operation for Wide-Band speech codec types (i.e. AMR-WB) .....	18
4.3 TFO Standard Version Handling .....	19
4.4 Document Content.....	20
5 TFO Frame Structure.....	21
5.1 General .....	21
5.2 TFO Frames for 16 kbit/s sub-multiplexing .....	21
5.2.1 TFO Frames for GSM Full Rate and GSM Enhanced Full Rate .....	21
5.2.2 TFO Frames for the Adaptive Multi Rate Family.....	22
5.2.2.1 TFO Frame Format AMR_TFO_16k .....	22
5.2.2.2 TFO Frame Format AMR_TFO_8+8k.....	25
5.2.2.3 TFO Frame Format AMR_WB_TFO_16k and AMR_WB_TFO_32k .....	27
5.2.3 Transmission of the bits of 16 kbit/s TFO Frames.....	29
5.2.3a Transmission of the bits of 32 kbit/s TFO Frames.....	29
5.2.4 Transmission of the bits of AMR_TFO_8+8k Frames .....	30
5.2.5 Optional AMR_TRAU_8+8k Frames.....	30
5.3 TFO Frames for 8 kbit/s sub-multiplexing .....	31
5.3.1 TFO Frame for the GSM Half Rate .....	31
5.3.2 Transmission of the bits of 8 kbit/s TFO frames.....	31
5.4 Void.....	32
5.5 Determination of the TFO Frame format .....	32
6 Elementary Procedures for TFO Operation.....	32
6.1 Pre-synchronisation of IPEs .....	32
6.2 TFO Negotiation .....	32
6.3 Codec Mismatch Resolution.....	33
6.4 TFO Establishment .....	34
6.5 Codec Optimisation .....	34
6.6 TFO Termination.....	34
6.7 TFO Fast Establishment after Local Handover .....	35
7 TFO Messages.....	36
7.1 Extensibility .....	37
7.2 Regular and Embedded TFO Messages.....	38
7.3 Cyclic Redundancy Check .....	38
7.4 TFO_REQ Messages.....	38
7.4.1 Definition of the SIG_LUC Extension_Block .....	39
7.4.2 Definition of the Codec_x Extension_Block .....	40
7.4.3 Definition of the Codec_List_Extension_Block .....	40
7.4.4 Definition of the Codec_Attribute_Head Extension_Block.....	41
7.4.5 Definition of the TFO_Version Extension_Block .....	42
7.4.5.1 Selector for Alternative Codecs .....	42

7.5	TFO_ACK Messages .....	42
7.6	TFO_TRANS Messages .....	43
7.7	TFO_NORMAL Message .....	43
7.8	TFO_FILL Message .....	43
7.9	TFO_DUP Message .....	43
7.10	TFO_SYL Message .....	43
7.11	Specification of the TFO Messages .....	44
7.11.1	Codec_Types .....	44
7.11.2	Codec_List .....	44
7.11.3	Codec_Type Attributes .....	44
7.11.3.1	AMR Codec_Type Attributes .....	44
7.11.3.1.1	AMR Active_Codec_Set Attributes .....	45
7.11.3.1.2	AMR Supported_Codec_Set Attributes .....	46
7.11.3.1.3	AMR specific Codec_Attribute_Head Extension_Block .....	47
7.11.3.1.4	AMR-WB specific Codec_Attribute_Head Extension_Block .....	48
8	Time Alignment of TFO Frames and TFO Messages .....	50
8.1	Alignment of TFO Frames and TFO Messages for GSM .....	50
8.1.1	Time Alignment of TFO Messages in GSM .....	50
8.1.2	Time Alignment of TFO Frames to Uplink TRAU Frames .....	50
8.1.3	Time Alignment of TFO Frames to Downlink TRAU Frames .....	51
8.2	Time Alignment of TFO Frames and TFO Messages for 3G .....	52
8.2.1	Time Alignment of TFO Messages in 3G .....	52
8.2.2	Time Alignment of TFO Frames to Uplink Iu Frames .....	53
8.2.3	Time Alignment of TFO Frames to Downlink Iu Frames .....	53
9	TFO State Machine .....	54
9.1	Initialisation .....	55
9.1.1	Not_Active State .....	55
9.1.2	Wakeup State .....	55
9.2	Establishment .....	55
9.2.1	First_Try State .....	55
9.2.2	Continuous_Retry State .....	55
9.2.3	Periodic_Retry State .....	56
9.2.4	Monitor State .....	56
9.2.5	Mismatch State .....	56
9.3	Contact State .....	56
9.4	Preparation .....	56
9.4.1	Wait_RC State .....	56
9.4.2	Konnnect State .....	57
9.5	Operation State .....	57
9.6	Local Handover .....	57
9.6.1	Fast_Try State .....	57
9.6.2	Fast_Contact State .....	57
9.7	Distant Handover, TFO Interruption .....	57
9.7.1	Sync_Lost State .....	57
9.7.2	Re_Konnnect State .....	57
9.7.3	TFO_Term .....	57
9.8	Failure State .....	58
10	Detailed Description of the TFO Protocol .....	58
10.1	Syntax Used for the TFO_Protocol Description .....	58
10.2	Detailed Description of the Conditions .....	58
10.2.1	Conditions for TFO_REQ, TFO_ACK, TFO_REQ_L, TFO_ACK_L, New_Local_Codec, New_Local_Config, Distant Config .....	58
10.2.2	Conditions for TFO_Frame .....	59
10.3	Abbreviations, Definitions, Notations used in the TFO_Protocol Description .....	60
10.4	Detailed Description of the Events .....	62
10.5	Actions Table .....	64
10.6	Protocol Tables .....	67
11	TFO Decision Algorithm .....	82
11.1	Main TFO Decision Procedure .....	83

11.2	TFO Decision Algorithm for AMR codec types .....	83
11.2.1	Principles .....	83
11.2.2	Available Information at Call Set-up .....	84
11.2.4	Flowchart for AMR-NB TFO Decision .....	85
11.2.5	Annotations to the Flowcharts .....	86
11.3	Immediate TFO Establishment.....	87
11.4	FR – HR – Optimisation (only for AMR-NB).....	87
11.5	Codec Mode Optimisation.....	87
11.6	Codec Type Optimisation and Codec Mismatch Resolution.....	88
11.6.1	Procedure .....	88
11.6.2	Preference List of TFO candidates .....	88
11.7	Immediate Codec Type Optimisation.....	90
11.8	TFO Decision Table for AMR-WB.....	90
12	Determination of the OACS for AMR-NB .....	91
12.1	Principles.....	91
12.2	Algorithm for OACS Determination .....	91
12.2.1	Case 1: No side supports ACS change.....	91
12.2.2	Case 2: Only one side supports ACS change .....	93
12.2.3	Case 3: Both sides support ACS change .....	93
12.3	OACS Selection Rules .....	94
12.3.1	Case 1: No Half Rate Channel is involved .....	94
12.3.2	Case 2: A Half Rate Channel is involved .....	95
12.4	Rules for Contiguous Subset Selection .....	95
12.5	Acceptability Rule for the OACS .....	95
12.6	FR – HR – Matching .....	95
12.7	Contiguity Rule .....	96
12.8	Examples of OACS Computation .....	96
12.8.1	TFO between a full rate channel and a half rate channel.....	96
12.8.2	TFO between two full rate channels with different ACS.....	96
12.8.3	Full Rate Channel with restricted capabilities .....	97
12.8.4	Scenario: Full Rate Channel with MACS == 2.....	97
12.8.5	Scenario: AMR codec type with only one supported mode.....	97
<b>Annex A (normative): In-band Signalling Protocol: Generic Structure.....</b>		<b>98</b>
A.0	Scope of Annex A and Annex B .....	98
A.1	Generic Structure of Inband Signalling Messages .....	98
A.1.1	Frequency and Order of Bit Transmission .....	98
A.1.2	IS_Header.....	99
A.1.3	IS_Command_Block .....	99
A.1.4	IS_Extension_Block(s).....	100
A.2	Detailed Specification of IS Messages.....	100
A.2.1	IS_REQ Message .....	100
A.2.2	IS_ACK Message.....	101
A.2.3	IS_IPE, IS_TRANS and IS_NORMAL Messages .....	101
A.2.4	IS_FILL Message .....	103
A.2.5	IS_DUP Message.....	103
A.2.6	IS_SYL Message .....	103
A.3	Keep_Open_Indication.....	103
A.4	Rules for Sending of IS Messages.....	104
A.5	System Identification and IS_System_Identification_Block.....	105
<b>Annex B (informative): In Path Equipment: Generic Rules and Guidelines.....</b>		<b>107</b>
B.1	Types of In Path Equipment.....	107
B.2	IS_Compliant IPEs .....	108
B.2.1	Typical IPEs are IS_Passive.....	108
B.2.2	IS_Message_Transparency.....	108
B.2.2.1	First IS Message .....	108
B.2.2.2	IS Messages within a Sequence .....	108
B.2.2.3	Isolated IS Message .....	109

B.2.2.4	Check if IS Message is following .....	109
B.3	IPE State Representation .....	109
B.3.1	IPE in Sync_Not_Found.....	111
B.3.2	IPE in Sync_Found.....	111
B.3.3	IPE in Sync_Lost.....	111
B.3.4	IPE in Keep_Open_Sync.....	112
B.3.5	IPE in Keep_Open_Lost.....	112
B.4	IPE Error Handling.....	112
B.5	IPE Transmission Delay .....	112
B.5.1	IPE Transmission Delay in Normal_Mode .....	113
B.5.2	IPE Transmission Delay in Transparent_Mode.....	113
B.6	Compliance to IS Messages .....	113
B.6.1	Compliance to IS_REQ and IS_ACK Messages .....	113
B.6.2	Compliance to IS_NORMAL Message.....	113
B.6.3	Compliance to IS_TRANS_x Messages.....	114
B.6.4	Compliance to IS_TRANS_x_u Messages.....	114
B.6.5	Compliance to IS_FILL Message.....	114
B.6.6	Compliance to IS_DUP Messages.....	115
B.6.7	Compliance to IS_SYL Messages .....	115
<b>Annex C (normative):</b>	<b>Tandem Free Operation in GSM, including AMR-WB .....</b>	<b>116</b>
C.1	Scope .....	116
C.2	Overview .....	116
C.2.1	TFO_TRAU .....	117
C.2.2	TFO_BSC.....	117
C.2.3	TFO_BTS.....	118
C.2.4	Modifications of the Codec Type and/or the Codec Configuration.....	119
C.3	TFO_TRAU .....	120
C.3.1	Rx_TRAU Process .....	121
C.3.2	Tx_TRAU Process .....	121
C.3.2.1	Downlink Speech Transmission and DTX handling if TFO is ON .....	122
C.3.2.1.1	GSM_FR, GSM_EFR and GSM_HR cases .....	123
C.3.2.1.2	AMR and AMR-WB cases.....	123
C.3.2.2	Synchronisation and Bit Errors in Received TFO Frames .....	124
C.3.2.2.1	GSM_FR, GSM_EFR and GSM_HR cases .....	124
C.3.2.2.2	AMR and AMR-WB cases.....	124
C.3.2.2.2.1	No format conversion .....	124
C.3.2.2.2.2	With format conversion.....	124
C.3.2.3	Maximum Rate Control .....	124
C.3.3	Tx_TFO Process.....	125
C.3.3.1	Maximum Rate Control .....	126
C.3.4	Rx_TFO Process .....	126
C.3.4.1	Search for and Monitoring of TFO Synchronization .....	126
C.3.4.2	Errors in TFO Messages and TFO Frames .....	127
C.3.5	TFO_Protocol Process.....	128
C.3.5.1	Messages from Rx_TRAU or local BSC.....	129
C.3.5.2	Messages to Tx_TRAU .....	129
C.3.5.3	Optional Messages to the local BSC.....	129
C.3.5.4	Messages to Tx_TFO.....	130
C.3.5.5	Messages from Rx_TFO .....	130
C.4	TFO_BTS .....	131
C.4.1	TFO_States and Transitions .....	131
C.4.2	Handling of downlink DTX in TFO .....	133
C.4.3	Handling of Errors in Configuration Parameters.....	133
C.4.4	Procedures for Round Trip Delay Measurements .....	133
C.5	TFO_BSC.....	133
C.5.1	Resolution of Codec Type Mismatch and Codec Type Optimization .....	133
C.5.2	Role of the BSC for AMR and AMR-WB TFO .....	134

C.5.2.1	Configuration of the AMR and AMR-WB speech service .....	134
C.5.2.2	Determination and Establishment of the Common ACS .....	134
C.5.2.3	Handovers and the AMR TFO .....	135
C.6	The Dialogue between TFO_TRAU and TFO_BTS .....	135
C.6.1	Configuration Parameters in AMR-NB TRAU/TFO frames .....	135
C.6.1.1	Configuration Protocol Format .....	135
C.6.1.2	Config_Prot field .....	136
C.6.1.3	Message_No Field .....	136
C.6.1.4	Configuration Parameters Fields .....	136
C.6.1.5	Mapping of the Configuration Parameters on 16 and 8 kbit/s TRAU/TFO frames for AMR Configuration .....	138
C.6.2	TFO and Handover Status of the Connection .....	139
C.6.2.1	TFO Status Messages .....	139
C.6.2.2	Notification of Status of Connection .....	139
C.7	The Dialogue between TFO_BTS and TFO_BSC .....	140
C.7.1	BSC to BTS messages .....	140
C.7.2	BTS to BSC messages .....	140
C.8	Configuration Parameter Exchange on Abis/Ater and A Interfaces for AMR and AMR-WB .....	141
C.8.1	Protocol for the Exchange of Configuration Parameters .....	141
C.8.2	Initial Configuration at Resource Allocation .....	142
C.8.3	Distant Configuration before TFO is established .....	142
C.8.4	Optimal TFO configuration .....	142
C.8.5	Configuration Exchange in TFO .....	142
C.8.6	Handover_Complete Notification in TFO .....	143
C.9	Location of the TFO Decision Algorithm .....	144
C.9.1	Immediate TFO Set-up .....	144
C.9.2	Codec Configuration Optimisation .....	144
C.9.3	Codec Type Optimisation .....	144
<b>Annex D (normative):</b>	<b>Tandem Free Operation in 3G including AMR-WB .....</b>	<b>145</b>
D.1	Scope .....	145
D.2	Overview .....	145
D.2.1	TFO_TC .....	147
D.2.2	TFO_MSC .....	147
D.3	TFO_TC .....	147
D.3.1	Rx_IU Process .....	148
D.3.2	Tx_IU Process .....	149
D.3.3	Tx_TFO Process .....	149
D.3.4	Rx_TFO Process .....	150
D.3.4.1	Search for and Monitoring of TFO Synchronization .....	151
D.3.5	TFO_Protocol Process .....	151
D.3.5.1	Messages from the MSC Server to TFO_Protocol .....	151
D.3.5.2	Messages from TFO_Protocol to Tx_IU .....	151
D.3.5.3	Messages from TFO_Protocol to the MSC Server .....	151
D.3.5.4	Messages between TFO_Protocol and Tx_TFO .....	151
D.3.5.5	Messages from Rx_TFO to TFO_Protocol .....	152
D.3.5.6	Messages from Rx_IU to TFO_Protocol .....	153
D.4	TFO in the RNC .....	153
D.5	TFO_MSC .....	153
D.5.1	Status of the Connection .....	154
D.5.2	Change of Codec Type or Configuration .....	154
D.6	Determination of the Optimal Codec Type and Optimal Configuration Parameters .....	154
<b>Annex E (normative):</b>	<b>TFO Decision Algorithm C-Code .....</b>	<b>155</b>
E.1	Brief Description of the Program 'tfo_decision' .....	155
E.1.1	Input .....	155
E.1.2	Output .....	156



<b>Annex F (informative):</b>	<b>Operator's Guide .....</b>	<b>157</b>
F.1	Avoidance of Codec Type Optimisation .....	157
F.2	Earliest possible TFO Establishment .....	157
F.2.1	Avoidance of Codec Mode Optimisation .....	157
F.2.2	Immediate TFO establishment with Codec Mode Optimisation .....	158
F.3	Usage of AMR Tandem compared to TFO with GSM_EFR, GSM_FR, or GSM_HR .....	158
F.4	Balance between Speech Quality and Network Capacity .....	159
<b>Annex G (informative):</b>	<b>Call flows for AMR TFO.....</b>	<b>160</b>
G.1	Typical Initialisation for TRAU, TC and TFO Protocol .....	161
G.2	Re-Initialisation during the Call after TFO_Disable .....	162
G.3	TFO_Disable during Operation.....	163
G.3.1	TFO_Disable – passive partner: UMTS .....	163
G.3.2	TFO_Disable – passive partner: GSM.....	164
G.4	Immediate TFO establishment for AMR.....	165
G.4.1	Con_Req / Con_Ack used on the TFO Interface.....	165
G.4.2	TFO_REQ_L / TFO_ACK_L used on the TFO Interface.....	166
G.5	Configuration Optimisation.....	168
G.6	AMR TFO: Mismatch Case .....	169
G.7	Intra BSC TFO Handover (TRAU remains) .....	170
G.7.1	TRAU-TC TFO connection.....	170
G.7.2	TRAU-TRAU TFO connection.....	171
G.8	Inter BSC TFO Handover (TRAU changes) .....	172
G.9	Immediate Codec Type Optimization .....	173
<b>Annex H (normative):</b>	<b>Definition of the Generic Configuration Frames for TFO.....</b>	<b>174</b>
H.1	Scope.....	174
H.2	Structure for Generic Configuration Frames.....	174
H.2.1	Frame Structure for 8 kBit/s sub-multiplexing.....	174
H.2.2	Frame Structure for 16 kBit/s sub-multiplexing.....	175
H.3	Coding of Generic Configuration Frames .....	175
H.3.1	Generic Configuration Frame Administration Section.....	176
H.3.1.1	Extendability.....	176
H.3.1.2	Version Handling.....	176
H.3.1.3	Configuration Exchange Protocol.....	176
H.3.1.4	System Identification, TFO and DTX control .....	176
H.3.1.5	Specific Section for the Active Codec Type.....	176
H.3.1.5.1	Specific Section for GSM_FR, GSM_HR, GSM_EFR.....	177
H.3.1.5.2	Specific Section for the AMR Narrow Band Family .....	177
H.3.1.5.3	Specific Section for the AMR Wide Band Family .....	177
H.3.1.6	Spare Bits.....	177
H.3.1.7	Error Detection and Error Handling.....	177
H.3.2	Configuration Parameter Section .....	178
H.3.2.1	Mapping for Single Codec Type.....	178
H.3.2.2	Codec List.....	179
<b>Annex I (informative):</b>	<b>Change History .....</b>	<b>180</b>
History .....		181

---

# Foreword

This Technical Specification has been produced by the 3<sup>rd</sup> Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

---

# 1 Scope

The present document contains the service description for the in-band signalling protocol for the support of Tandem Free Operation of the Half Rate, Full Rate, Enhanced Full Rate, Adaptive Multi Rate narrowband and Adaptive Multi Rate Wide-Band speech codec types in GSM and GSM-evolved 3G systems.

---

# 2 References

The following documents contain provisions, which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

- [1] 3GPP TS 42.053: "Digital cellular telecommunication system (Phase 2+); Tandem Free Operation (TFO); Service Description; Stage 1".
- [2] GSM 43.053: "Digital cellular telecommunication system (Phase 2+); Tandem Free Operation (TFO); Service Description; Stage 2".
- [3] 3GPP TS 48.060: "Digital cellular telecommunication system (Phase 2+); Inband control of remote transcoders and rate adaptors for full rate traffic channels".
- [4] 3GPP TS 48.061: "Digital cellular telecommunication system (Phase 2+); In-band Control of Remote Transcoders and Rate Adaptors for half rate traffic channels".
- [5] 3GPP TS 46.010: "Digital cellular telecommunications system (Phase 2+); Full rate speech transcoding".
- [6] 3GPP TS 46.020: "Digital cellular telecommunications system (Phase 2+); Half rate speech transcoding".
- [7] 3GPP TS 46.060: "Digital cellular telecommunications system (Phase 2+); Enhanced Full Rate (EFR) speech transcoding".
- [8] 3GPP TS 26.090: "Mandatory Speech Codec speech processing functions AMR Speech Codec - Transcoding functions".
- [9] 3GPP TS 45.009: "Digital cellular telecommunications system (Phase 2+); Link Adaptation".
- [10] 3GPP TS 48.008: "Digital cellular telecommunications system (Phase 2+); Mobile-services Switching Centre - Base Station System (MSC - BSS) interface; Layer 3 specification".
- [11] 3GPP TS 48.054: "Digital cellular telecommunication system (Phase 2+); Base Station Controller - Base Transceiver Station (BSC - BTS) interface; Layer 1 structure of physical circuits".
- [12] 3GPP TS 48.058: "Digital Cellular telecommunications system (Phase 2+), "Base Station Controller - Base Transceiver Station (BSC - BTS) interface; Layer 3 specification".
- [13] ITU-T Recommendation G.711: "Pulse code modulation (PCM) of voice frequencies".
- [14] GSM 04.18: "Mobile radio interface layer 3 specification; Radio Resource Control Protocol".
- [15] 3GPP TS 23.153: "Out of Band Transcoder Control; Stage 2".

- [16] 3GPP TS 29.232: “Media Gateway Controller (MGC) – Media Gateway (MGW) Interface; Stage 3”
- [17] 3GPP TS 25.415: “UTRAN Iu interface User plane protocols”
- [18] 3GPP TS 26.171: AMR-WB Speech Codec; General Description
- [19] 3GPP TS 26.190: AMR-WB Speech Codec; Transcoding Functions

---

## 3 Definitions and abbreviations

### 3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

**Transcoder:** device that converts the encoding of information from one particular scheme to a different one

NOTE 1: A **Speech Transcoder** in a GSM or 3G system converts the speech encoding usually from G.711 [13] to a format optimised for the transmission over the Air Interface. The new format relates to a specific **speech Codec**.

**Tandem Free Operation:** call configuration where a transcoder device is physically present in the signal path, but the transcoding functions are bypassed

NOTE 2: The transcoding device may perform control and protocol conversion functions.

**Transcoder Free Operation:** call configuration where no transcoder device is physically present and hence no control or conversion or other functions associated with it are activated

**Compressed Speech Samples:** speech samples coded according to one of the Speech Codec Types supported by the TFO specification

**PCM Samples:** speech samples coded according to ITU-T Recommendation G.711 A-Law or  $\mu$ -Law at 64 kbit/s

**Speech Codec Type:** speech Codec among those supported by this TFO specification: GSM\_FR, GSM\_HR, GSM\_EFR, FR\_AMR, HR\_AMR, OHR\_AMR, UMTS\_AMR, UMTS\_AMR\_2, FR\_AMR-WB, UMTS\_AMR-WB, OFR\_AMR-WB, OHR\_AMR-WB

**AMR Speech Codec Type:** one of the following Speech Codec Types: FR\_AMR, HR\_AMR, UMTS\_AMR, UMTS\_AMR\_2, FR\_AMR-WB, UMTS\_AMR-WB, OFR\_AMR-WB, OHR\_AMR-WB

**AMR-NB Speech Codec Type:** one of the following Speech Codec Types: FR\_AMR, HR\_AMR, UMTS\_AMR, UMTS\_AMR\_2, OHR\_AMR.

**AMR-WB speech Codec Type:** one of the following Speech Codec Types: UMTS\_AMR-WB, FR\_AMR-WB, OFR\_AMR-WB, OHR\_AMR-WB

**Non-AMR Speech Codec Type:** one of the following Speech Codec Types: GSM\_FR, GSM\_HR, or GSM\_EFR

**Speech Codec Configuration:** set of parameters defining the operational conditions of a **Speech Codec Type**

EXAMPLE: The Speech Codec Configuration of an AMR Speech Codec Type defines the ACS, the SCS...

**TRAU Frame** or **TRAU Speech Frame:** refer to a Speech Frame carried over the Abis/Ater Interface in a GSM network

**TFO Frame** or **TFO Speech Frame:** refer to the Speech Frames exchanged between the Transcoders when Tandem Free Operation is active

**Abis/Ater:** applies to a GSM network where either the GSM Abis or Ater interfaces are used, depending on the location of the Transcoder and Rate Adaptor Units

Other definitions are contained in [1] and [3].

## 3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

ACS	Active Codec Set
ACT	Active Codec Type
AMR	Adaptive Multi-Rate
ATVN	AMR-TFO Version Number
BSC	Base Station Controller
BSS	Base Station Sub-system
BTS	Base Transceiver Station
CACS	Common Active Codec Set
CSCS	Common Supported Codec Set
DACS	Distant Active Codec Set
DSCS	Distant Supported Codec Set
EFR	Enhanced Full Rate
FQI	Frame Quality Index
FR	Full Rate
HOM	Hand-Over-Mode
HR	Half Rate
IACS	Immediate Active Codec Set
ICM	Initial Codec Mode
IPE	In Path Equipment
LACS	Local Active Codec Set
LSB	Least Significant Bit
LSCS	Local Supported Codec Set
MACS	Maximum number of Codecs Modes in the Active Codec Set
MGw	Media Gateway
MS	Mobile Station
MSB	Most Significant Bit
MSC	Mobile Switching Centre
NB	Narrow-Band
OACS	Optimised Active Codec Set
OD	Optimal or Distant Configuration requested
OM	Optimisation Mode supported
PCM sample	8-bit value representing the A_Law or $\mu$ -Law coded sample of a speech or audio signal; sometimes used to indicate the time interval between two PCM samples (125 $\mu$ s).
PCM	Pulse_Coded_Modulation
PCM_Alaw_Idle	PCM sample with value 0x54
PCM_Alaw_Idle	PCM sample with value 0x54.
PCM_Alaw_Silence	PCM sample with value 0xD5
PCM_Alaw_Silence	PCM sample with value 0xD5.
PCM_Idle	either PCM_Alaw_Idle, or PCM_ $\mu$ Law_Idle, dependent on application
PCM_Silence	either PCM_Alaw_Silence, or PCM_ $\mu$ Law_Silence, dependent on application
PCM_ $\mu$ Law_Idle	PCM sample with value 0x00
PCM_ $\mu$ Law_Idle	PCM sample with value 0x00.
PCM_ $\mu$ Law_Silence	PCM sample with value 0xFF
PCM_ $\mu$ Law_Silence	PCM sample with value 0xFF.
PDU	Packet Data Unit
PLMN	Public Land Mobile Network
RAN	Radio Access Network
RATSCCH	Robust AMR Traffic Synchronised Control Channel
RIF	Request Indication Flag
RNC	Radio Network Controller
SCR	Source Controlled Rate
SCS	Supported Codec Set
T_Bits	Time Alignment Bits
Tbth	Time delay Bad Frame Handling
TC	Transcoder
TCME	TFO Circuit Multiplication Equipment
TFO	Tandem Free Operation

TFO	Tandem Free Operation
TFO_ACK	TFO Acknowledgement Message
TFO_DUP	TFO (Half) Duplex Mode Message
TFO_DUP	TFO (Half) Duplex Mode Message
TFO_FILL	TFO Fill Message
TFO_NORMAL	TFO Normal Mode Message
TFO_REQ	TFO Request Message
TFO_SYL	TFO Sync Lost Message
TFO_TRANS	TFO Transparent Mode Message
TRAU	Transcoder and Rate Adaptor Unit
TrFO	Transcoder Free Operation
TSM	TFO Setup Mode
Tultfo	Time delay UpLink TFO
UE	User Equipment
WB	WideBand

## 4 General Description

### 4.1 Background Information

Tandem Free Operation (TFO) is intended to avoid the traditional double speech encoding/decoding in MS to MS (GSM), MS to UE (GSM/3G) or UE to UE (3G) call configurations. In the following paragraphs the term "MS" is used for MS and UE, the term UE only if a 3G terminal is explicitly addressed.

In a normal MS-MS call configuration the Speech Signal is first encoded in the originating MS, sent over the Air Interface, converted to A-law or  $\mu$ -law ITU-T Recommendation G.711 [13] in the local transcoder, carried over the fixed network, transcoded again in the distant transcoder, sent over the distant Air Interface and finally decoded in the terminating MS (see Figure 4.1-1). In this configuration, the two speech codecs (coder/decoder pairs) are in "Tandem Operation". The key inconvenience of a tandem configuration is the speech quality degradation introduced by the double transcoding. This degradation is usually more noticeable when the speech codecs are operating at low rates.

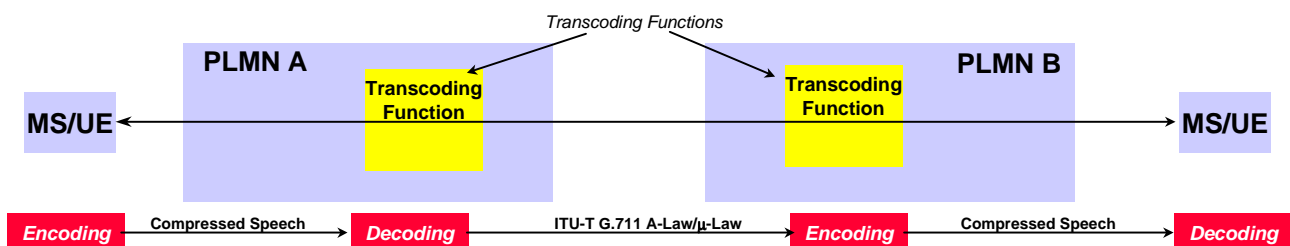


Figure 4.1-1: Typical Speech Codec Tandem Operation

When the originating and terminating connections are using the same speech codec, it is possible to transmit transparently the speech frames received from the originating MS to the terminating MS without activating the transcoding functions in the originating and terminating networks (see figure 4.1-2). In this configuration, "Tandem Free Operation" is on-going.

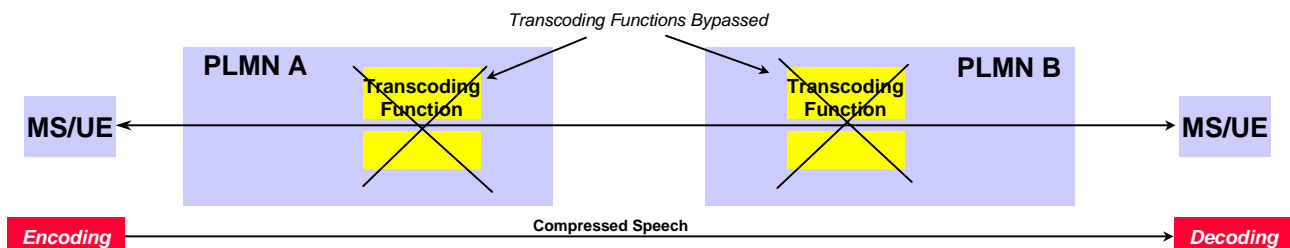


Figure 4.1-2: Tandem Free Operation of Speech Codec

The key advantages of Tandem Free Operation are:

- Improvement in speech quality by avoiding the double transcoding in the network;
- Possible savings on the inter-PLMN transmission links, which are carrying compressed speech compatible with a 32 kbit/s or 16 kbit/s or 8 kbit/s sub-multiplexing scheme, including packet switched transmission;
- Possible savings in processing power in the network equipment since the transcoding functions in the Transcoder Units are bypassed;
- Possible reduction in the end-to-end transmission delay.

The major constraint of Tandem Free Operation is that the inter-PLMN transmission links must be transparent to the compressed speech frames. This means that any device located in the transmission path (IPE: in path equipment) between the originating and the terminating transcoders must be disabled, switched-off, or made aware of the TFO situation to keep unaltered any compressed speech frame sent over the transmission path. Examples of such devices are listed in annex B.

The TFO Protocol defined in the present document provides the following services:

- Establishment of a transparent path between transcoders;
- Provision of an In-band signalling link between transcoders;
- Exchange of information on the active speech codec type and supported speech codec types at both ends of the call configuration;
- Codec Mismatch Resolution;
- Establishment and Maintenance of Tandem Free Operation when identical codec types are used at both ends of the call configuration;
- Fast and seamless fall back to Tandem Operation in case of necessary or unexpected TFO interruption (i.e. activation of supplementary services);
- Support for cost efficient transmission.

The present document defines Tandem Free Operation for the different Speech Codec Types used in GSM and GSM-evolved 3G systems. This includes the GSM\_FR, GSM\_HR, GSM\_EFR and FR\_AMR, HR\_AMR, OHR\_AMR, UMTS\_AMR, UMTS\_AMR\_2, FR\_AMR-WB, UMTS\_AMR-WB, OFR\_AMR-WB, OHR\_AMR-WB codec types. However, the procedures used to establish TFO are considered system independent and could be extended to call configurations involving other systems like ISDN phones, speech servers, IP Multimedia or other wireless systems.

For non-AMR Speech Codec Types (i.e. GSM\_FR, GSM\_EFR and GSM\_HR), Tandem Free Operation is fully compatible with the installed equipment base. The feature is fully supported by the Transcoder Units. The additional processing complexity is small compared to the encoding/decoding functions. Other network elements are not affected and possibly not aware of the establishment of Tandem Free Operation.

For the support of AMR Tandem Free Operation in GSM, the BTS and possibly the BSC may be involved in addition to the TRAU.

The resolution of a possible codec mismatch is defined as an optional feature. A codec mismatch occurs when incompatible speech codecs are used at both ends of the call configuration at call set-up. The resolution consists in finding an optimal speech codec on which TFO may be established. For that purpose, other elements in the Radio Access Network (BSS in GSM or RNC in 3G) might be involved.

## 4.2 Principle of TFO Operation

### 4.2.1 Principle for TFO Operation for Narrow-Band speech codec types

Tandem Free Operation is activated and controlled by the Transcoder Units after the completion of the call set-up phase at both ends of an MS-MS, MS-UE, or UE-UE call configuration. The TFO protocol is fully handled and terminated in the Transcoder Units. For this reason, the Transcoder Units cannot be bypassed in Tandem Free Operation. This is the key difference with the feature called Transcoder Free Operation (TrFO) defined in 3GPP TS 23.153.

In return, the Transcoder Units continuously monitor the normal Tandem Free Operation and can terminate TFO as soon as necessary with limited impact on the speech quality.

Before TFO is activated, the Transcoder Units exchange conventional 64 kbit/s PCM speech samples coded according to the ITU-T Recommendation G.711 [13] A-Law or  $\mu$ -Law. The Transcoders can also exchange TFO messages by stealing the least significant bit in every 16<sup>th</sup> speech sample (see annex A for the specification of the TFO message transmission rule and clauses 6 to 8 for the description of the TFO procedures and messages content).

If compatible Speech Codec Types and Configurations are used at both ends of the MS-MS, MS-UE, or UE-UE call configuration, the Transcoders automatically activate TFO. If incompatible Speech Codec Types and/or Configurations are used at both ends, then a codec mismatch situation exists. TFO cannot be activated until the codec mismatch is resolved. This capability is an optional feature involving other network elements of the Radio Access Network. The rules for finding a common codec type and solve the codec mismatch are defined in clauses 11 and 12.



Once TFO is activated, the Transcoder Units exchange TFO Frames carrying compressed speech and in-band signalling, which structure is derived from the GSM TRAU Frames defined in the 3GPP TS 48.060 and 48.061 (see clause 5). The exchange of TFO messages is still possible while TFO is active. In this case, the stealing process will result in embedding a message in the synchronisation pattern of the TFO Frame.

When TFO is activated between two end connections using the GSM\_HR speech codec, the TFO Frames are carried over 8 kbit/s channels mapped onto the least significant bit (LSB) of the 64 kbit/s PCM speech samples.

When TFO is activated between two end connections using the GSM\_FR or GSM\_EFR speech codecs, the TFO Frames are carried over 16 kbit/s channels mapped onto the two least significant bits of the 64 kbit/s PCM speech samples.

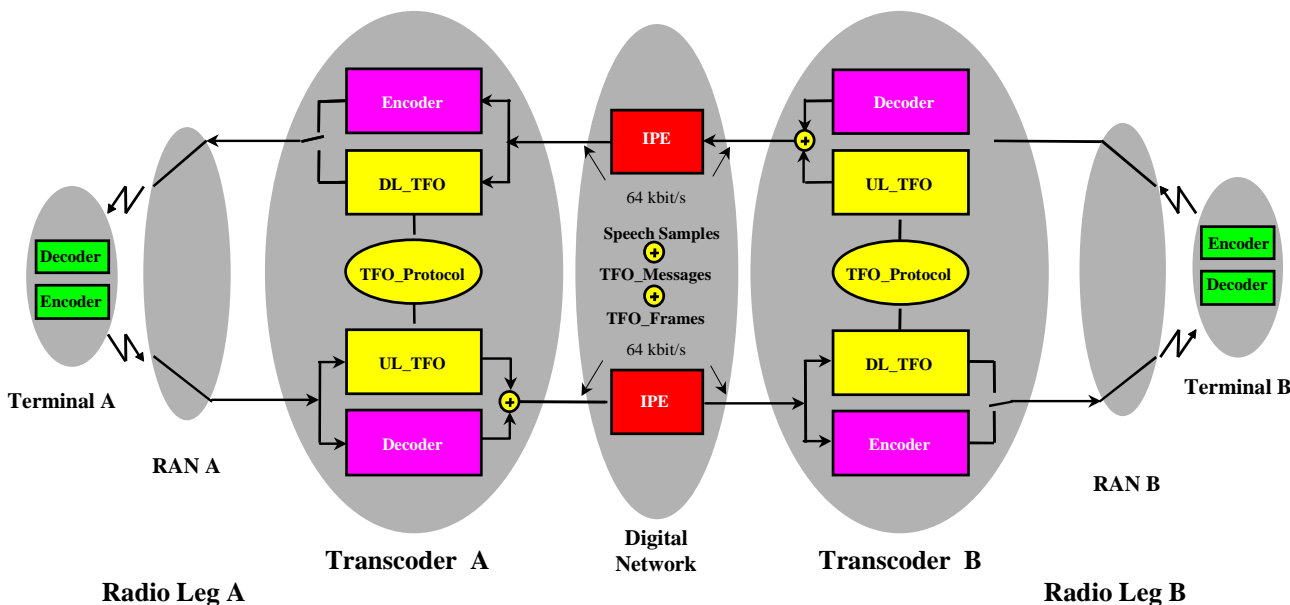
When TFO is activated between two end connections using the AMR speech codec, the TFO Frames are carried over 8 or 16 kbit/s channels mapped onto the least or two least significant bits of the 64 kbit/s PCM speech samples. The format depends on the codec configuration (Optimized Active Codec Set).

To facilitate a seamless TFO interruption, the six or seven MSB of the PCM speech samples (not compressed) are transmitted to the far end unchanged.

Like GSM TRAU Frames, the TFO Frames have a fixed size (and duration) of:

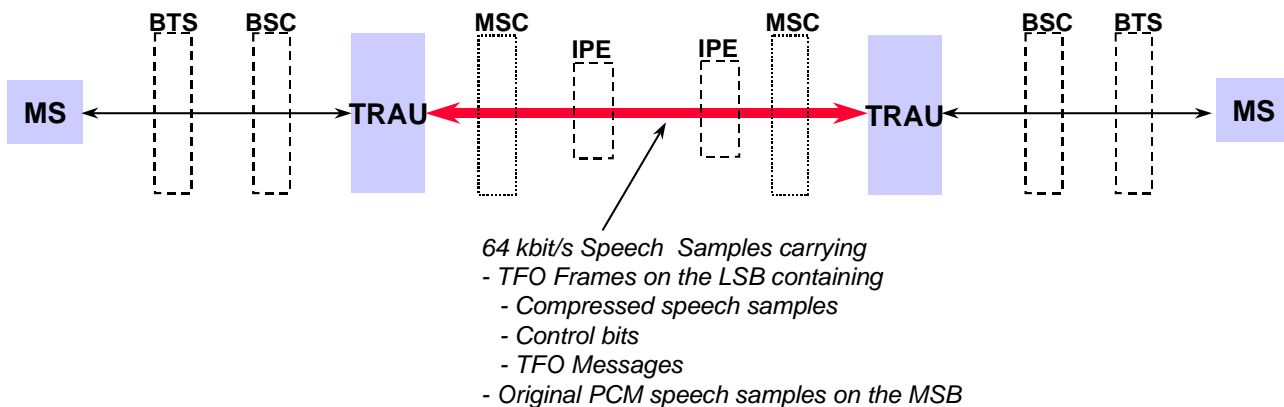
- 160 bits (20 ms) for the 8 kbit/s format;
- 320 bits (20 ms) for the 16 kbit/s format.

Figure 4.2-1 provides a reference model for the functional entities handling Tandem Free Operation. The TFO Protocol is fully described in clauses 9 (State Machine) and 10 (Detailed Protocol).



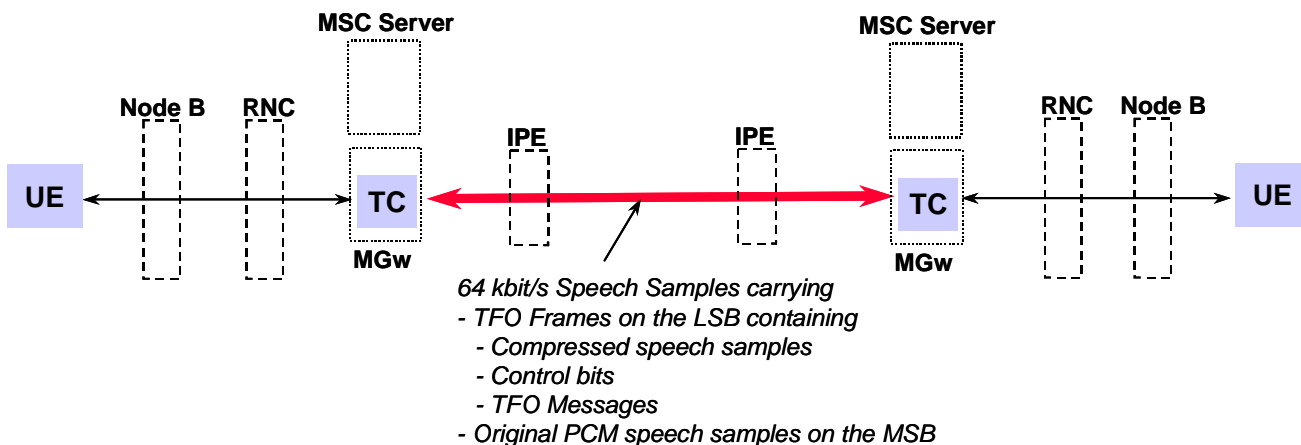
**Figure 4.2.1-1: Functional Entities Handling Tandem Free Operation**

The same TFO protocol and Frame Format is used irrespective of the PLMN types at both ends of the call configuration. Figure 4.2-2 shows a normal TFO configuration involving the same or two different GSM networks.



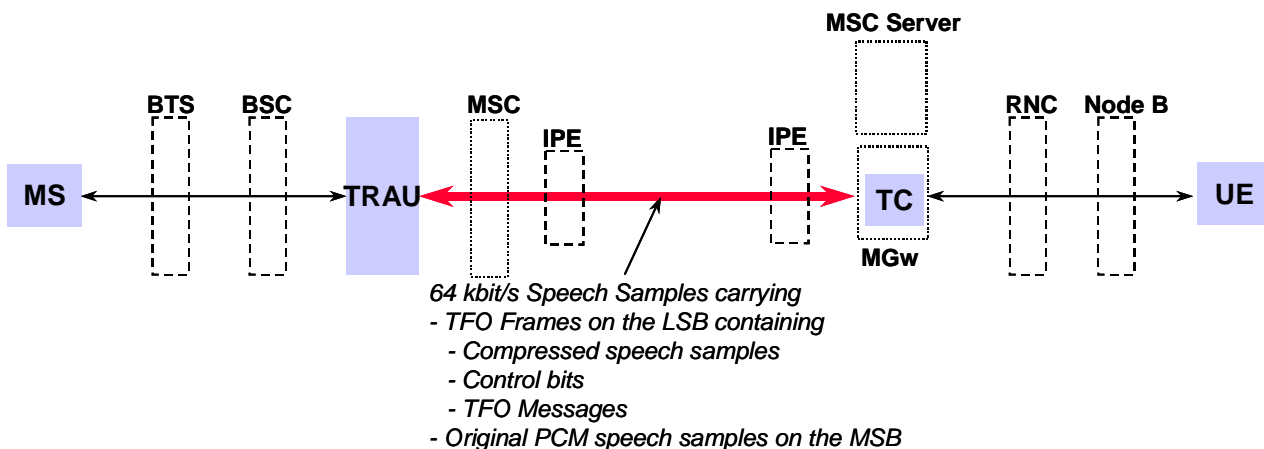
**Figure 4.2.1-2: TFO Configuration between GSM Networks**

Figure 4.2-3 presents a TFO configuration involving two GSM-evolved 3G Networks. Note that the same protocol and Frame Structure are also used irrespective of the type of Transmission Network connecting the two 3G networks (ATM or STM).



**Figure 4.2.1-3: TFO Configuration between 3G Networks**

Finally, figure 4.2-4 presents a TFO configuration involving two different network types (GSM and 3G). Similar configurations could be derived with any network supporting a TFO protocol compatible with the present document.



**Figure 4.2.1-4: TFO Configuration between a GSM and a 3G Network**

## 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 it 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 speech codec types 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 is not desirable to occupy radio resources unnecessarily with wide-band signals, until TFO is operational.
- Figure 4.2.2-2 describes the situation after AMR-WB TFO establishment
- Because of the higher speech signal bandwidth (up to almost 24 kbit/s for AMR-WB) up to the four LSBs must be stolen by TFO frames.
- In case of TFO interruption, the remaining MSBs of the PCM speech samples (not compressed) might not only be less than for narrowband TFO, the transcoded bits carry a different kind of signal: The downsampled signal has narrowband properties (as depicted in figure 4.2.2-3). Because of the significant difference of the narrowband speech signal's impression (possibly even distorted by the stealing of four LSBs) to the wideband signal's quality, AMR-WB TFO interruptions should be avoided as best as possible.

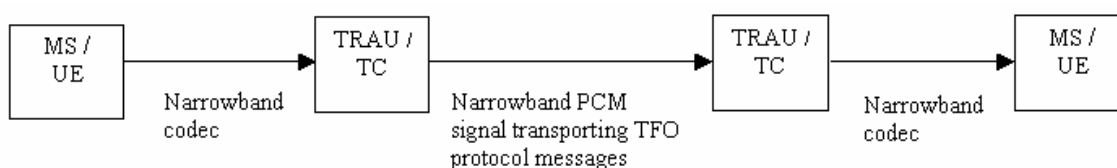


Figure 4.2.2-1: Pre-TFO scenario for AMR-WB (for subsequent codec optimisation)

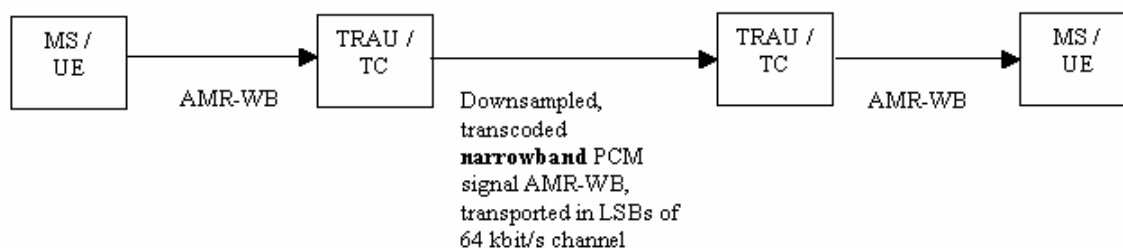


Figure 4.2.2-2: AMR-WB with TFO Ongoing

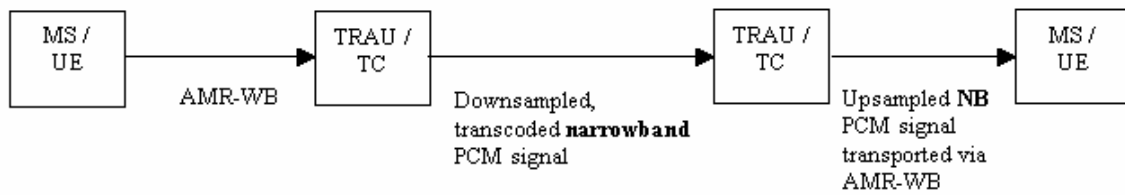


Figure 4.2.2-3: AMR-WB after TFO interruption

## 4.3 TFO Standard Version Handling

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.

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 smallest defined TFO Version number is 0.0. It stands for all TFO Versions before 5.3. All numbers between 0.0 and 5.3 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 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	<b>Optional.</b> The TFO Version extension block need not to be sent. If not contained in TFO Messages, or is lower than 5.3, then Pre-REL-5 handling shall apply	<b>Mandatory,</b> if TFO Version is 5.3 or higher.	If the TFO Version is lower than 5.3 then Generic Configuration Frames shall <u>not</u> be used. Only TFO_REQ_L and (TFO_ACK_L) shall be used.  If the TFO Version is 5.3 or higher, then Generic Configuration Frames shall be used. TFO_REQ_L and TFO_ACK_L shall <u>not</u> be used embedded into TFO Frames.
FR_AMR HR_AMR UMTS_AMR UMTS_AMR2 OHR_AMR	<b>Optional.</b> The TFO Version extension block need not to be sent. If not contained in TFO Messages, or is lower than 5.3, then Pre-REL-5 handling shall apply	<b>Mandatory,</b> if TFO Version is 5.3 or higher.	If the TFO Version is lower than 5.3, then Generic Configuration Frames shall <u>not</u> be used. If the TFO Version is 5.3 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 <u>not</u> be used embedded into TFO Frames.
FR_AMR-WB UMTS_AMR-WB OFR_AMR-WB OHR_AMR-WB	<b>Mandatory.</b> The TFO Version extension block shall always be sent.	<b>Mandatory.</b>	Generic Configuration Frames shall be used. TFO_REQ_L and TFO_ACK_L shall <u>not</u> be used embedded into TFO Frames.

## 4.4 Document Content

In the following, clause 5 defines the structure of the TFO Frames exchanged between the Transcoder Units. The TFO Frames carry the compressed speech (payload) and some control bits for the inter-transcoder in-band signalling. Clause 6 introduces the elementary procedures used for the establishment and maintenance of Tandem Free Operation. Clause 7 defines the detailed content of the TFO messages associated with the TFO procedures. The TFO Message Structure follows the generic format defined in Annex A. Clause 8 defines how the TFO messages are mapped onto the TFO Frames. Clause 9 defines the TFO State Machine. Clause 10 contains the detailed TFO protocol. Clause 11 and 12 specify the TFO Decision algorithm and the optional Codec Mismatch Resolution.

Annex B is an informative annex defining the expected behaviour of In-Path Equipment (IPE) for compatibility with Tandem Free Operation.

Annex C and Annex D define specific TFO processes for GSM and 3G systems.

Annex E contains a reference implementation for the TFO decision algorithm (C-code) described in clause 11 and 12.

Annex F is an informative Implementer's Guide containing recommendations in the implementation and introduction of AMR TFO.

Annex G provides basic Message Flow sequences for the TFO protocol.

## 5 TFO Frame Structure

### 5.1 General

TFO Frame formats are defined for the following Speech Codec Types:

- GSM Full Rate (GSM\_FR);
- GSM Half Rate (GSM\_HR);
- GSM Enhanced Full Rate (GSM\_EFR);
- Adaptive Multi Rate Family (FR\_AMR, HR\_AMR, UMTS\_AMR, UMTS\_AMR\_2, OHR\_AMR);
- WB Adaptive Multi Rate Family (UMTS\_AMR-WB, FR\_AMR-WB, OFR\_AMR-WB, OHR\_AMR-WB)

TFO Frame formats for 8 kbit/s, 16kbit/s and 32 kbit/s sub-multiplexing are defined in the following clauses.

### 5.2 TFO Frames for 16 kbit/s sub-multiplexing

#### 5.2.1 TFO Frames for GSM Full Rate and GSM Enhanced Full Rate

The TFO Frames for GSM\_FR and GSM\_EFR are derived from the **uplink** TRAU Frames as defined in the 3GPP TS 48.060. Table 5.2.1-1 defines the coding of the Control Bits for these TFO Frames.

**Table 5.2.1-1: Control Bits in TFO Frames for GSM\_FR and GSM\_EFR**

Control Bit	Description	Comment
C1 - C4	<b>Frame Type</b>	copied from uplink TRAU Frames
0.0.0.1	GSM_FR	All other code words are reserved.
1.1.0.1	GSM_EFR	
C5	<b>EMBED</b>	Indicates the presence of an embedded TFO Message
C6 - C11	Spare	(is Time Alignment in TRAU Frame) set to Spare by TRAU
C12	<b>BFI</b>	Copied from the uplink TRAU Frame
C13 - C14	<b>SID</b>	Copied from the uplink TRAU Frame
C15	<b>TAF</b>	Copied from the uplink TRAU Frame
C16	Spare	set to Spare by TRAU
C17	<b>DTXd</b>	Copied from the uplink TRAU Frame
C18 - C21	Spare	set to Spare by TRAU

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

The **Synchronisation Pattern** is similar to the Synchronisation Pattern in the 3GPP TS 48.060, with some exceptions depending on 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.

For the coding of the **Time Alignment Bits** (T\_Bits, T1.. T4) see 3GPP TS 48.060. The T\_Bits normally correspond to the T\_Bits received in the up-link TRAU Frame.

## 5.2.2 TFO Frames for the Adaptive Multi Rate Family

The TFO Frames for any narrow-band AMR Codec Type use always 16 kbit/s sub-multiplexing on the A-Interface, regardless which sub-multiplexing is used on the Abis-Interfaces. Two different AMR\_TFO Frame formats exist. One, called AMR\_TFO\_16k, is based on the TRAU Frame format for 16 kBit/s sub-multiplexing, as described in 3GPP TS 48.060. The other one, called AMR\_TFO\_8+8k, is based on the TRAU Frame format for 8 kbit/s sub-multiplexing, as described in 3GPP TS 48.061, with an added synchronisation pattern, to improve transmission and synchronisation quality on the A-Interface.

Optionally the TRAU frame format AMR\_TRAU\_8+8k may be used on the Abis-Interface for 16 kBit/s sub-multiplexing, when a TFO connection with HR\_AMR on the distant side is established.

Additionally, a frame format using 16 kbit/s sub-multiplexing is defined for wide band AMR codec types, called AMR\_WB\_TFO\_16k. It is based on the TRAU frame format for 16 kBit/s sub-multiplexing, as described in 3GPP TS 48.060.

### 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.

Octet no.	Bit number							
	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)	<b>Frame_Type / Codec Type</b> (GSM_FR) <b>FR_AMR</b> <b>HR_AMR</b> <b>UMTS_AMR</b> <b>UMTS_AMR_2</b> (FR_AMR-WB) (UMTS_AMR-WB) <b>OHR_AMR</b> (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	<b>EMBED</b> 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)	<b>Codec Mode Request (CMR)</b>	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 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	<b>RIF (Request or Indication Flag)</b>	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 lu Frames, are monitored. Whenever the Codec Mode changes, the RIF bit is set to "0". The next frames are then alternately marked with RIF = "1", "0", "1" and so on.	
C13	Spare (set to 1)		C13 is spare in UL TRAU frames.	
C14 – C16	<b>Config_Prot</b>		Coding defined in Annex C.	
C17 C18	<b>Mess No</b>		Coding defined in Annex C.	
C19	DTXd (see note)		Copied from uplink TRAU Frame in GSM	
C20 0 1	<b>TFOE</b> 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	<b>Frame_Classification</b> "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_Classificatio n == 0.0)	<b>Codec Mode Indication (CMI)</b> ; (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	<b>Time Alignment Bits</b>		In GSM copied from the uplink TRAU Frame In 3G, generated by the TC (UMTS) based on lu 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 the FR\_AMR, HR\_AMR, UMTS\_AMR\_2 and OHR\_AMR 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 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 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.

**Table 5.2.2.1-3: Conversion between Generic AMR Frames and AMR\_TFO\_16k Frames**

Generic AMR Frame			AMR_TFO_16k Frame				
Frame Quality Indicator	Frame Type Index	TX_TYPE or RX_TYPE (see 3GPP TS 26.101)		Frame Classification C21 - C22	CMI or CMR C23 - C25	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) 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.

### 5.2.2.2 TFO Frame Format AMR\_TFO\_8+8k

The AMR\_TFO\_8+8k Frame formats are derived from the GSM Adaptive Multi-Rate 8 kbit/s TRAU Frame formats defined in 3GPP TS 48.061. AMR Codec Modes with rates up to 7,40 kbit/s can be used with these AMR\_TFO\_8+8k Frame formats. The AMR\_TFO\_8+8k is described in an 8 kbit/s frame structure for the second LSB of the PCM samples and an 8 kbit/s synchronisation pattern for the LSB. The TFO Frame structures for the second LSB are illustrated in Figures 5.2.2.2-1 to 5.2.2.2-3, using the same notations as in 3GPP TS 48.061. Figure 5.2.2.2-4 defines the additional Synchronisation pattern for the LSB. Both frames shall be exactly synchronised on the A-Interface. This additional Synchronisation Pattern is sometimes modified by embedding of TFO Messages, indicated by the value of the **EMBED** bit:

**EMBED equal "0":** the Synchronisation Pattern is exactly as described in Figure 5.2.2.2-4;  
**EMBED equal "1":** the Synchronisation Pattern contains an embedded TFO Message.

	Bit number							
Octet no	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	0
2	1	D1	D2	D3	D4	D5	D6	D7
3	1	C1	C2	C3	C4	C5	D8	D9
4	0	1	D10	D11	D12	D13	D14	D15
5...19	1							
20	1	D121	D122	D123	D124	D125	D126	T

**Figure 5.2.2.2-1: AMR\_TFO\_8+8k Frame Structure, second LSB:  
 NO\_SPEECH frames and SPEECH frames for Codec Modes 4,75, 5,15 and 5,90 kbit/s**

	Bit number							
Octet no	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	0
2	1	D1	D2	D3	D4	D5	D5	D7
3	1	C1	C2	C3	D8	D9	D10	D11
4...19								
20	D130	D	D	D	D	D	D	D137

**Figure 5.2.2.2-2: AMR\_TFO\_8+8k Frame Structure, second LSB:  
 Speech frame for Codec Mode 6,70 kbit/s**

	Bit number							
Octet no	1	2	3	4	5	6	7	8
1	0	0	1	D1	D2	D3	D4	D5
2	0	D6	D7	D8	D9	D10	D11	D12
3	1	C1	C2	C3	D13	D14	D15	D16
4	0	D17	D18	D19	D20	D21	D22	D23
5	D24	D	D	D	D	D	D	D31
6 ... 19								
20	D144	D145	D146	D147	D148	D149	D150	D151

**Figure 5.2.2.2-3: AMR\_TFO\_8+8k Frame Structure, second LSB:  
 Speech frame for Codec Mode 7,40 kbit/s**

Octet no	Bit number							
	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	0
2	1	EMBED	EXTEND					
3...6	1							1
7	0	1						
8 ... 19	1							
20	1						1	1

**Figure 5.2.2.2-4: AMR\_TFO\_8+8k Frame Structure, LSB: Additional Synchronisation Pattern**

- EXTEND equal "0": The bits not defined in the Synchronisation Pattern described in Figure 5.2.2.2-4 are "spare" (equal 1). In AMR\_TFO\_8+8k frames these undefined bit positions shall leave the original bits of the PCM coded speech unaltered. In TRAU\_8+8k frames these undefined bits shall be set to "1" (spare).
- EXTEND equal "1": The bits not defined in the Synchronisation Pattern described in Figure 5.2.2.2-4 transport other parameters (tbd).

Table 5.2.2.2-1 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.

**Table 5.2.2.2-1: The coding of the Control Bits (C1 .. C5) for AMR\_TFO\_8+8k Frames**

Control Bit	Description	No_Speech frames and Speech frames for 4,75, 5,15 and 5,9 kbit/s Codec Modes	6,7 + 7,4 kbit/s Codec Mode
C1 – C3	see 3GPP TS 48.061	<ul style="list-style-type: none"> <li>- For the low rates frame types, these bits jointly define the CMI, CMR and RIF.</li> <li>- For the No_Speech frame type, they define the RIF.</li> <li>- Copied from the uplink TRAU Frame in GSM.</li> <li>- Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.2.2.2-2 below)</li> </ul>	<ul style="list-style-type: none"> <li>- For the 6,70 and 7,40 kbit/s speech frame, these bits jointly provide the CMR, RIF, and the Frame Classification.</li> <li>- Copied from the uplink TRAU Frame in GSM.</li> <li>- Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.2.2.2-2 below)</li> </ul>
C4 - C5	<b>Frame Classification</b> (No_Speech and low rates modes only)	<ul style="list-style-type: none"> <li>- Copied from the uplink TRAU Frame in GSM</li> <li>- Derived from the Frame Quality Indicator and Frame Type for 3G systems (see Table 5.3.2-2 below)</li> </ul>	The Frame_Classification is defined by bits C1-C3 in 6,70 and 7,40 kbit/s TFO Frames C4..C5 are not existent for this codec modes
1 1	"Speech_Good"		
1 0	"Speech_Degraded"		
0 1	"Speech_Bad"		
0 0	"No_Speech"		

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

The coding of the **Data Bits** is described in 3GPP TS 48.061 [4].

For 3G systems, Table 5.2.2.2-2 provides the conversion rules between the generic AMR Frames as defined in 3GPP TS 26.101 and the AMR\_TFO\_8+8k Frames. In this table, the arrows in the fourth column indicate the direction for which the conversion applies. The Transcoder shall autonomously and internally generate a RIF alternating between the binary "0" and "1" values (see Annex D).

**Table 5.2.2.2-2: Conversion between Generic AMR Frames and AMR\_TFO\_8+8k Frames**

Generic AMR Frame			TFO Frame for 8 kbit/s submultiplexing					
Frame Quality Indicator	Frame Type Index	TX_TYPE or RX_TYPE (see 3GPP TS 26.101)		Bits C1 .. C3	Bits C4 – C5	Data bits in No_Speech frames D8 .. D10	Equivalent Frame Type in 3GPP TS 48.061	Frame Type
1	0-2	SPEECH_GOOD	< >	as 3GPP TS 48.061	1 1	-	Speech_Good	4,75 kbit/s, 5,15 kbit/s, 5,90 kbit/s Modes
1	0-2	SPEECH_GOOD	<	as 3GPP TS 48.061	1 0	-	Speech_Degraded	
0	0-2	SPEECH_BAD	< >	as 3GPP TS 48.061	0 1	-	Speech_Bad	
1	3-4	SPEECH_GOOD	< >	as 3GPP TS 48.061	Speech bits	-	Speech_Good	6,70 kbit/s, 7,40 kbit/s Modes
1	3-4	SPEECH_GOOD	<	as 3GPP TS 48.061	Speech bits	-	Speech_Degraded	
0	3-4	SPEECH_BAD	< >	as 3GPP TS 48.061	Speech bits	-	Speech_Bad	
1	8	SID_FIRST	< >	as 3GPP TS 48.061	0 0	SID_First	No_Speech	No Speech
1	15	NO_DATA	<	as 3GPP TS 48.061	0 0	Onset	No_Speech	
1	8	SID_UPDATE	< >	as 3GPP TS 48.061	0 0	SID_Update	No_Speech	
0	8	SID_BAD	< >	as 3GPP TS 48.061	0 0	SID_Bad	No_Speech	
1	15	NO_DATA	< >	as 3GPP TS 48.061	0 0	No_Data	No_Speech	

The **Synchronisation Pattern** in the second LSB of the PCM samples is identical to the Synchronisation Pattern in 3GPP TS 48.061. Embedding of TFO Messages has no influence on this synchronisation pattern.

For the coding of the **Time Alignment Bit** (T Bit) for all modes below 5,9 kbit/s and the No\_Speech Frame, see 3GPP TS 48.061. The T-Bit in a TFO Frame normally corresponds to the T\_Bit received in the up-link TRAU Frame.

**5.2.2.3 TFO Frame Format 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 Adaptive Multi-Rate Wide Band as defined in the 3GPP TS 48.060. The AMR\_WB\_TFO\_16k Frame structure is illustrated in Table 5.2.2.3-1 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 synchronisation 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.

**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**

Octet no.	Bit number							
	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**

Octet no.	Bit number							
	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
<b>C1 - C4</b> (0.0.0.1) (0.0.1.1) (0.1.0.0) (0.1.0.1) (0.1.1.0)	<b>Frame_Type / Codec Type</b> (GSM_FR) (FR_AMR) (HR_AMR) (UMTS_AMR) (UMTS_AMR_2)	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.
<b>1.0.0.1</b> <b>1.0.1.0</b> (1.0.1.1)	<b>FR_AMR-WB</b> <b>UMTS_AMR-WB</b> (OHR_AMR)	Note: Codec Types in (brackets) are not supported by this TFO Frame format. They are listed to show their coding for convenience.
<b>1.1.0.0</b> <b>0.0.1.0</b> (1.1.0.1)	<b>OFR_AMR-WB</b> <b>OHR_AMR-WB</b> (GSM_EFR)	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.

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

The CRC covering also the control bits C1..C25 shall be recomputed in the transcoder, 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 26.201. 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 26.201), 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.

3G systems using the UMTS\_AMR-WB 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 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.

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.

For the coding of the **Time Alignment Bits** (T\_Bits, T1 .. T4) see 3GPP TS 48.060. 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.

#### 5.2.3a 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 5.2.2.3-1 and 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 **fourth 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.

## 5.2.4 Transmission of the bits of AMR\_TFO\_8+8k Frames

For the purpose of this description the 160+160 bits of one AMR\_TFO\_8+8k frame are arranged in 20 rows (1..20), with 8 bit each (1..8: one octet) as shown in Figures 5.2.2.2-1 to 5.2.2.2-4.

The bits of AMR\_TFO\_8+8k frames are transmitted in the following order:

Bit  $m$  of octet  $n$  of the **additional synchronisation pattern** described in Figure 5.2.2.2-4 shall be transmitted in the **Least Significant Bit** of the

PCM sample  $k = (n-1)*8+m$ ; with  $m = (1..8)$  and  $n = (1..20)$ .

Bit  $m$  of octet  $n$  of the **No\_Speech and Speech frames** as described in Figures 5.2.2.2-1 to 5.2.2.2-3 shall be transmitted in the **Second Least Significant Bit** of the

PCM sample  $k = (n-1)*8+m$ ; with  $m = (1..8)$  and  $n = (1..20)$ .

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.

## 5.2.5 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 48.061 (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:

- the additional synchronisation pattern shall be transmitted in the Second LSBs of the 16 kbit/s sub-multiplexed channel, while the TRAU\_8k frames shall be transmitted in the LSBs;
- no embedded TFO Messages shall exist in TRAU\_8+8k frames;
- the EMBED bit shall be set to "0";
- the EXTEND bit shall be set to "0";
- undefined bits in Figure 5.2.2.2-4 shall be set to "1" (spare) in TRAU\_8+8k frames.

The potential transition from regular TRAU\_16k frames to AMR\_TRAU\_8+8k frames shall be triggered by the FR\_TRAU with TFO\_Soon and Dis\_Req (including the distant Codec Type: HR\_AMR) in downlink direction.

If the BTS applies the optional AMR\_TRAU\_8+8k format, then the BTS shall respond with the acknowledging TFO\_Soon in the first AMR\_TRAU\_8+8k frame in uplink. This will result in a small additional delay for the decoded PCM samples, which the TRAU shall handle by proper concealment techniques. The delay for TFO Messages and TFO Frames is, however, not increased: since no format conversion is necessary in the TRAU the delay for AMR\_TFO\_8+8k frames is minimised. After TFO has been established the TRAU shall change from TRAU\_16k to AMR\_TRAU\_8+8k in downlink with the reception of the first AMR\_TFO\_8+8k frame.

If the BTS does not apply the AMR\_TRAU\_8+8k frame format in uplink, the TRAU shall also not use this in downlink. The TRAU shall perform format conversion in uplink from TRAU\_16k format to AMR\_TFO\_8+8k format and in downlink from AMR\_TFO\_8+8k format to TRAU\_16k format. This will cause an additional delay of TFO Messages and TFO Frames, which shall be handled by inserting the necessary number of T\_Bits. This format conversion causes also an additional delay in downlink, which the BTS shall handle by proper buffering technique.

## 5.3 TFO Frames for 8 kbit/s sub-multiplexing

### 5.3.1 TFO Frame for the GSM Half Rate

The GSM Half Rate (GSM\_HR) TFO Frames are always based on the **uplink** GSM Half Rate TRAU Frames for **8 kbit/s** submultiplexing scheme, as defined in the **3GPP** TS 48.061.

If GSM\_HR TRAU Frames with 16 kbit/s submultiplexing are used on the Abis/Ater interface, then the Control and Extended Control Bits for the 8 kbit/s TFO Frame need to be generated on basis of the received Control Bits from the TRAU Frame.

The coding of the **Control Bits** (C1 .. C9) is defined by the following Table 5.3.1-1:

**Table 5.3.1-1: Coding of the Control Bits (C1 .. C9) for the GSM\_HR**

Control Bit	Description	Comment
C1 - C4 0.0.0.1	Frame Type GSM_HR	All other code words are reserved.
C5	EMBED	Indicates the presence of an embedded TFO Message
C7 - C8	spare	
C9	DTXd	Copied from the uplink TRAU Frame

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

The **Synchronisation Pattern** is similar to the Synchronisation Pattern in the 3GPP TS 48.061, with some exceptions depending on the value of the EMBED Bit:

EMBED equal "0": the Synchronisation Pattern is exactly as described in the 3GPP TS 48.061;  
EMBED equal "1": the Synchronisation Pattern contains an embedded TFO Message.

Coding of the **Extended Control Bits (XC1 .. XC6)**:

**XC1** is copied from the uplink TRAU Frame.  
**XC2 .. XC6**: These bits are normally copied from the 8 kbit/s TRAU Frame.  
All other codes are reserved.

For the coding of the **Data Bits** see 3GPP TS 48.061.

For the coding of the **Time Alignment Bits** see 3GPP TS 48.061. The T\_Bits normally correspond to the T\_Bits received in the up-link TRAU Frame.

### 5.3.2 Transmission of the bits of 8 kbit/s TFO frames

For the purpose of this description the 160 bits of one frame are arranged in 20 rows (1..20), with 8 bit each (1..8: one octet) as in 3GPP TS 48.061.

The bits of 8 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

$$\text{PCM sample } k = (n-1)*8+m; \quad \text{with } m = (1..8) \text{ and } n = (1..20).$$

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.



## 5.4 Void

## 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 12,65 kbit/s, the TFO frame format shall be AMR\_WB\_TFO\_16k.

For all other AMR WB TFO connections, the TFO frame format shall be AMR\_WB\_TFO\_32k.

---

# 6 Elementary Procedures for TFO Operation

This clause provides a simplified overview of the elementary procedures of the Tandem Free Operation Protocol. The complete, binding specification of the TFO Protocol is provided in clause 10.

## 6.1 Pre-synchronisation of IPEs

As soon as the local transcoder receives and sends speech samples and TFO is enabled, it initiates the TFO negotiation by sending **TFO\_FILL** messages, in order to pre-synchronise potential IPEs quickly. The IPEs will then let further TFO messages pass transparently (see Annex B for guidelines for In-Path Equipment behaviour).

The distant TC may initiate the same procedure at the same time.

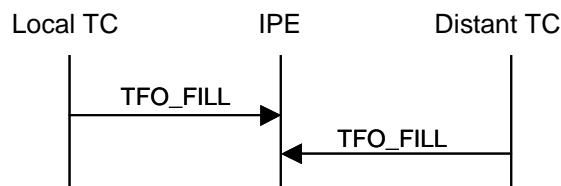


Figure 6.1

If the IPE does not support TFO, i.e. if it is not transparent for the TFO Messages and TFO Frames, it is perceived by the local transcoder in the same way as if the distant transcoder does not answer (see clause 6.2).

## 6.2 TFO Negotiation

The transcoder sends **TFO\_REQ** messages, indicating its System Identification (3G, GSM...) and the Speech Codec Type used with its main characteristics (ACS for AMR). If the distant transcoder supports TFO, it will answer by a **TFO\_ACK** message. The distant transcoder may initiate the same procedure at the same time.

If the local and distant transcoders use compatible Speech Codec Types (or compatible configurations of the same Speech Codec Type), see clause 11, they will go into TFO. Otherwise, a Codec Mismatch Resolution may be initiated, if supported by the transcoder.

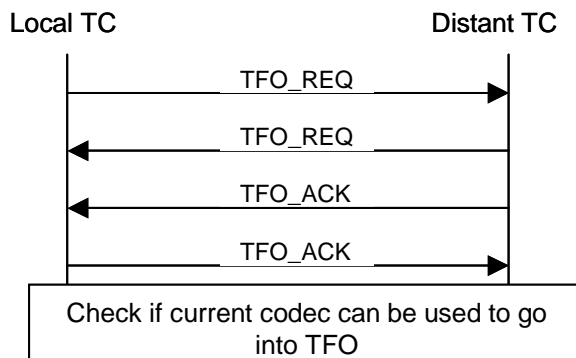


Figure 6.2

In some rare cases, the transcoders might also go into TFO even if both ends use different Speech Codec Types or different Configurations of the same Speech Codec Types. Typical examples of this situation occur when both ends use AMR Speech Codec Types with a substantial subset of identical Codec Modes. The conditions and rules related to this situation are defined in clause 11.

The distant transcoder may not answer for following reasons (the list is not exhaustive):

- The call is connected to PSTN (and then there is no distant transcoder!);
- The distant transcoder does not support TFO or TFO is disabled there;
- The path between the transcoders is not transparent.

In these cases, the local transcoder sends several TFO\_REQ and returns to normal mode. However, it continues to monitor if there are TFO messages inserted in the PCM samples.

### 6.3 Codec Mismatch Resolution

If the optional Codec Mismatch Resolution is supported, the transcoders shall exchange their full codec capabilities (Supported Codec List, with the full range of parameters for these codecs) by sending **TFO\_REQ\_L messages** or **Con\_Req frames**. These are acknowledged by **TFO\_ACK\_L** messages, respectively **Con\_Ack** frames. The same procedure may be initiated by the distant transcoder.

The same algorithm is then run at both extremities to determine a Common Speech Codec Type and its configuration to go into TFO. If no Common Speech Codec Type exists, the transcoders give up TFO. Any Speech Codec Type or Configuration listed in the Supported Codec Set is a candidate for TFO establishment. If a Codec Type configuration is undesirable, e.g. Full Rate Codec Type when operating on a Half Rate Channel, it should not be listed in the Supported Codec List.

Once the Common Speech Codec Type/Configuration is defined, each side must modify its Local Used Speech Codec Type and/or Configuration to the Common Speech Codec Type, if necessary. This operation may involve other network elements (BSS/RAN) and is out of the scope of the present document. Once the Speech Codec Type is set to the Common Speech Codec Type, the transcoder shall re-initialise the TFO Negotiation as defined in clause 6.2.

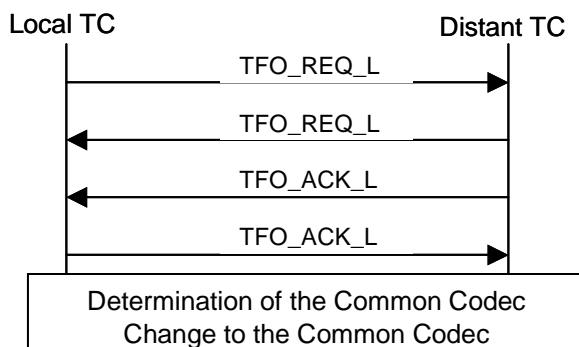


Figure 6.3

If the Codec Mismatch Resolution is not supported, the List of Supported Codec Types shall be restricted to the Local Active Codec Type and its Configuration (Active Speech Codec Mode/s in use).

## 6.4 TFO Establishment

To establish TFO, the transcoders send a **TFO\_TRANS** message to indicate to the IPEs that TFO frames follow, and begins to send **TFO frames**. The TFO\_TRANS message also defines the bandwidth occupied by the TFO frames (8 kbit/s or 16 kbit/s or 32 kbit/s).

Once both transcoders send and receive TFO frames, encoded with the Common Speech Codec Type, TFO is established.

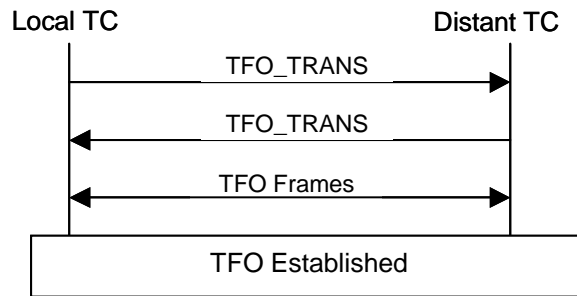


Figure 6.4

## 6.5 Codec Optimisation

Once TFO is established, the transcoders shall exchange their capabilities available for Optimisation by sending a TFO\_REQ\_L message or a Configuration frame. The TFO\_REQ\_L message is acknowledged by TFO\_ACK\_L messages, the Configuration Request by an Configuration Acknowledgement. This may trigger a Codec Optimisation. The TFO Decision Algorithm will determine, if another Common Speech Codec Type/Configuration exists with the potential to provide better speech quality while operating in TFO.

If the Optimisation leads to a new Common Speech Codec Type and/or Configuration, both ends shall switch to the new Common Speech Codec following the same procedure as in clause 6.3 Codec Mismatch Resolution.

The Codec Optimisation may temporarily break TFO while the Speech Codec is switched to the new Optimised Codec Type/Configuration.

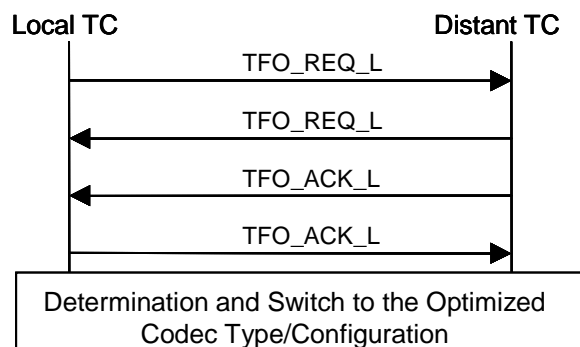


Figure 6.5

## 6.6 TFO Termination

TFO may be terminated for the following reasons (the list is not exhaustive):

- TFO is disabled in one of the transcoders;
- The call is released;
- An in-call modification from speech to data is initiated;
- A handover moves the call to a transcoder that does not support TFO, or where TFO is disabled;
- A handover moves the call to a cell where no common codec can be found with the distant side.

The transcoder which is still in TFO shall stop sending TFO frames, go back to normal operation and send a **TFO\_NORMAL** message to indicate to the IPEs that TFO has ended.

## 6.7 TFO Fast Establishment after Local Handover

While TFO is established, if the local side is handed over, the distant side may not detect the loss of synchronisation immediately and continue to send TFO Frames.

Once the handover is performed, the new local transcoder receives TFO Frames, while TFO is not yet re-established. If the Speech Codec Types on both sides match, the local TC sends a **TFO\_DUP** message to indicate the situation to the distant TC. Meanwhile, the distant transcoder may have detected a loss of synchronisation, which it signals by sending a **TFO\_SYL** message. If further TFO Frames and especially if a **TFO\_SYL** message are received, the new local transcoder sends TFO Frames and goes into TFO.

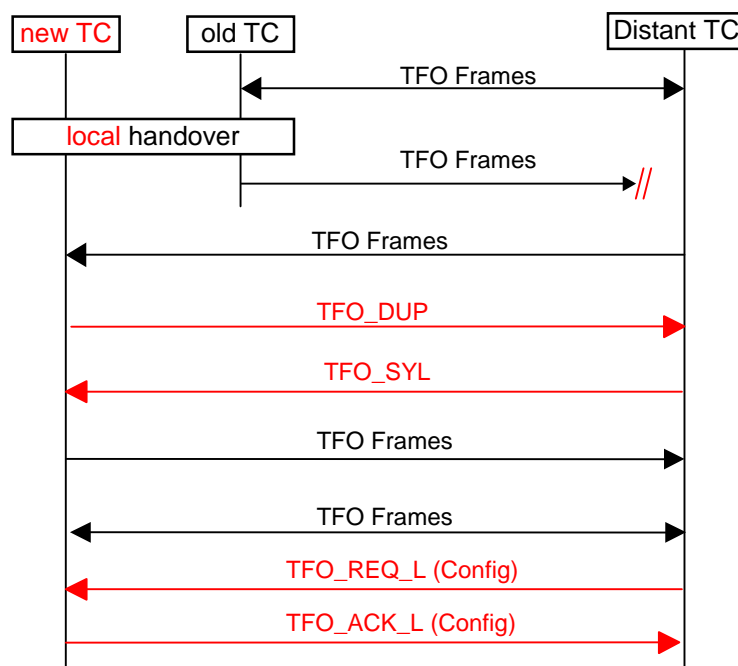


Figure 6.6

The same procedure applies if the new local Transcoder operates an AMR Speech Codec Type and receives acceptable TFO Frames (AMR TFO Frames for one of the Codec Modes in the ACS) after a local handover. The local Transcoder assumes that the ACS was not changed during the Handover and sends TFO Frames to the distant Transcoder. The local and distant Transcoders should then confirm that they are operating on the same or compatible ACS by exchanging TFO\_REQ\_L messages (or Configuration Frames, see example below) and by running the TFO Decision Algorithm.

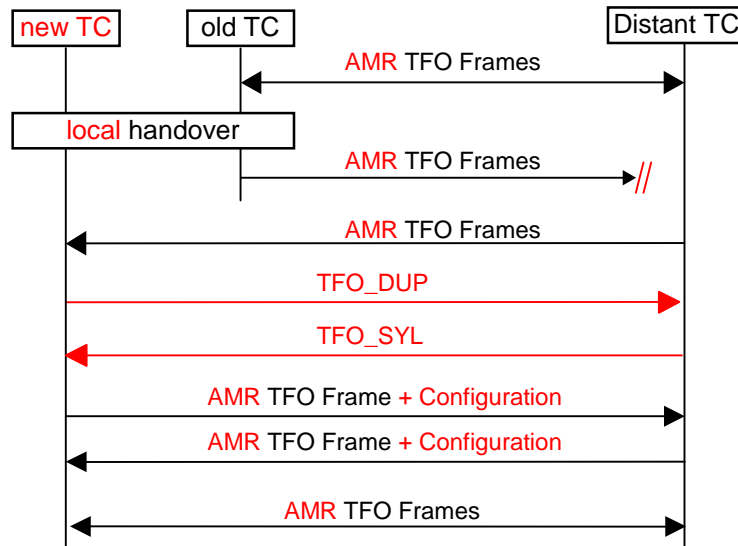


Figure 6.7

## 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:

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

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:

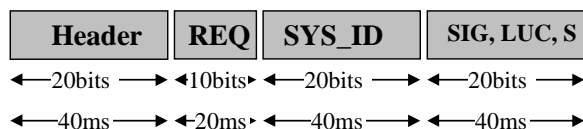
TFO\_REQ (Sys\_Id, LSig, Local\_Used\_Codec\_Type[, Used\_Codec\_Attributes] )

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**

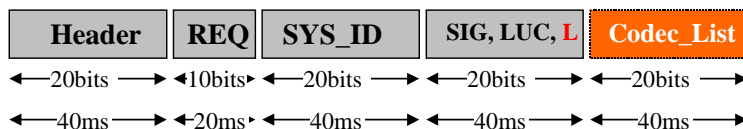


Figure 7.4-2: Construction of the shortest possible TFO\_REQ\_L Message

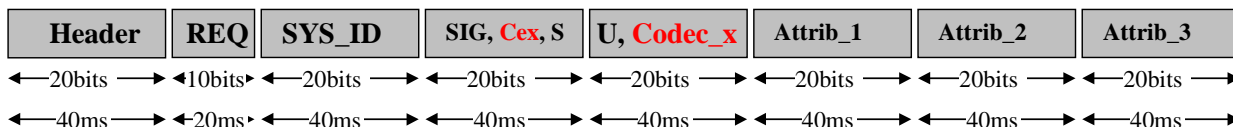


Figure 7.4-3: Example of a TFO\_REQ Message with a Codec with an index higher than 15 and with three Attribute Extension\_Blocks (300 ms length)

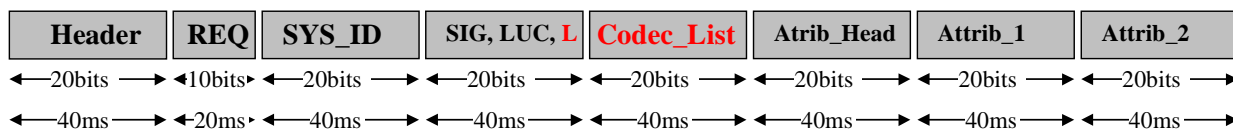


Figure 7.4-4: Example of a TFO\_REQ\_L Message with Codec\_List and one alternative Codec with two Attribute Extension\_Blocks (300 ms length)

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.

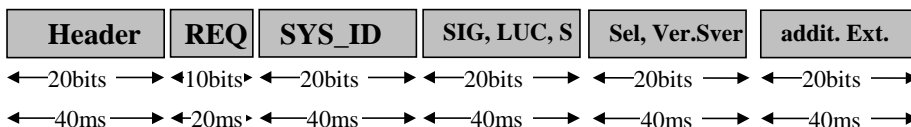


Figure 7.4-5: Construction of a TFO\_REQ Message with Selector, TFO\_Version.Subversion and one additional Extension\_Block

### 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	<b>Selector</b>	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	<b>Ver</b>	This field contains the TFO_Version number as specified in Annex H
Bit 11	"0"	normal IS-Message Sync Bit, constant
Bit 12.. 15:	<b>Sver</b>	This field contains the TFO_Subversion number as specified in Annex H
Bit 16..18:	<b>CRC</b>	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	<b>EX</b>	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, RSign, Local\_Used\_Codec\_Type [, Used\_Codec\_Attributes] )

TFO\_ACK\_L (Sys\_Id, RSign, 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 unexpectedly, 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, OHR\_AMR) and the Adaptive Multi-Rate Wideband Codec\_Types (FR\_AMR-WB, UMTS\_AMR-WB, OFR\_AMR-WB, OHR\_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). These attributes are signalled differently for the AMR Codec\_Types and AMR-WB Codec\_Types, resulting in a different construction of TFO messages.

The ACS is related to the Local\_Used\_Codec\_Type and is part of the Used\_Codec\_Attributes. One and exactly one ACS\_Extension\_Block shall be sent in all cases where the Local\_Used\_Codec\_Type is an AMR Codec\_Type. In all cases where the Local\_Used\_Codec\_Type is an AMR-WB Codec\_Type the ACS is signalled within the AMR-WB specific Attribute\_Head\_Extension\_Block. In the former case, the 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 codec modes 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. In the case of an AMR Codec\_Type the corresponding ACS\_Extension\_Block is following next. In the case of an AMR-WB Codec\_Type no ACS\_Extension\_Block is following since the ACS is already defined within the Attribute\_Head.

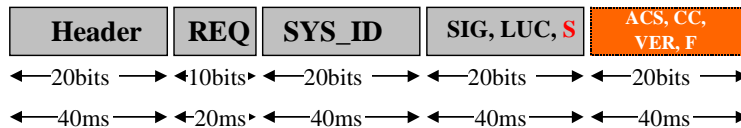
The SCS shall be sent in TFO\_REQ or TFO\_ACK only if the ACS\_Extension\_Block indicates that the sending side does not support the full set of AMR codec modes, but a subset (Full\_Sub flag). In this case the SCS\_Extension\_Block shall follow immediately after the ACS\_Extension\_Block.

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 an AMR Codec\_Type. In addition, one SCS\_Extension\_Block is needed for each AMR

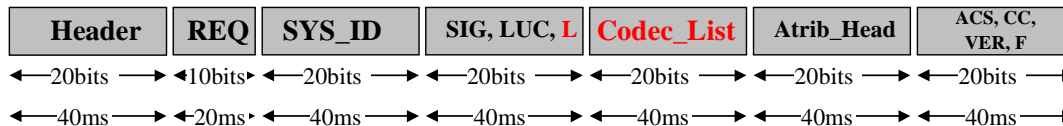
Codec\_Type flagged in the Local\_Codec\_List. In that case an Attribute\_Head\_Extension\_Block shall follow after the Local\_Codec\_List, indicating the Codec\_Type it specifies, followed by the corresponding SCS\_Extension\_Block. If multiple AMR\_Codec\_Types are flagged, then multiple Attribute\_Heads 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). For each AMR-WB Codec\_Type flagged in the Local\_Codec\_List, one Attribute\_Head Extension\_Block shall follow after the Local\_Codec\_List. Since the AMR-WB specific Attribute\_Head fully defines the SCS no further SCS Extension\_Block is following.

The following figures give the examples for the full-set AMR TFO Messages. Note that an additional TFO\_Version Extension\_Block shall follow if the TFO version is equal or greater than 5.



**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 an AMR Codec\_Type, 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.

For example, assume that the Local\_Used\_Codec\_Type is FR\_AMR and the supported codecs which are flagged in the Codec\_List are FR\_AMR, HR\_AMR, and FR\_AMR-WB. Then, if neither FR\_AMR nor HR\_AMR support the full set of AMR modes, the following six Extension\_Blocks follow the Codec\_List: Atrib\_Head(FR\_AMR) – ACS(FR\_AMR) – SCS(FR\_AMR) – Atrib\_Head(HR\_AMR) – SCS(HR\_AMR) – Atrib\_Head(FR\_AMR-WB).

**7.11.3.1.1 AMR Active\_Codec\_Set Attributes**

One AMR\_ACS Extension\_Block shall be added in the TFO\_REQ and TFO\_ACK messages after the SIG\_LUC Extension\_Block if an AMR 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	<b>Active Codec Set (NB_ACS)</b>	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	<b>Full_Sub (NB_F/S)</b>	0: Full Set supported, NB_SCS is not following 1: Subset only supported, NB_SCS is following immediately
Bit 11	"0"	Normal IS-Message Sync Bit, constant
Bit 12	spare	set to "1"
Bit 13	<b>Optimisation Mode (NB_OM)</b>	ACS Optimisation Mode 0 No ACS Change supported 1 ACS change supported
Bit 14 & 15	<b>NB_Ver</b>	<b>Version Number of the AMR-NB TFO Scheme</b> Bit 15 is equivalent to the ATVN in Configuration Frames, see Annex C
Bit 16..18	<b>CRC</b>	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	<b>EX</b>	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 Extension\_Block contains the information on the AMR 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).

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	<b>Supported Codec Set (NB_SCS)</b>	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	<b>NB_MACS MSB</b>	See comment for Bit 12...13
Bit 11	"0"	normal IS-Message Sync Bit, constant
Bit 12...13	<b>NB_MACS LSBs</b>	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	<b>NB_Ver</b>	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	<b>CRC</b>	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19 20	<b>EX</b>	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	<b>PAR_Sel</b>	Differentiates this Extension_Block <b>0</b> : Parameters included in <b>PAR</b> field: Simple Codec_List_Extension <b>1</b> : Length Indicator ( <b>LI</b> ) included: Parameters follow in subsequent Extension_Blocks
Bit 3..10	<b>CoID = HR_AMR or FR_AMR or UMTS_AMR or UMTS_AMR2 or OHR_AMR</b>	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:	<b>LI / PAR</b>	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:	<b>CRC</b>	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	<b>EX</b>	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.  
 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 shall ignore it.

If PAR\_Sel is set to "0", then one of 15 possible AMR Configurations is indicated in the PAR field and no additional Codec Attribute Extension\_Blocks do follow.  
 The coding for PAR (bits 12.13.14.15) is defined in Table 7.11.3.1.3-2 (Config-NB-Code):

**Table 7.11.3.1.3-2: Preferred Configurations for the Adaptive Multi-Rate Codec Types**

Configuration → (Config-NB-Code) ↓ Codec Mode	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
12,20								1							1	1
10,20							1						1	1		
7,95						1									1	1
7,40					1						1	1				
6,70				1						1	1	1	1	1		
5,90			1						1	1	1	1	1	1	1	1
5,15																
4,75	1								1	1	1	1	1	1	1	1
OM	F		F	F	F	F	F	F	F	F	F	A	F	A	F	A
HR_AMR	Y		Y	Y	Y	Y			Y	Y	Y					
FR_AMR, OHR_AMR, UMTS_AMR, UMTS_AMR_2	Y		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

The "1" in the table indicates that the Codec Mode is included in the Active Codec Set of the Configuration.

The parameter "OM" (Optimisation Mode) defines whether the indicated Configuration can be changed to any of the other Allowed ones (OM == A) or if the change is Forbidden (OM == F). For the three "A" configurations (11, 13 and 15) the TFO Decision algorithm shall consider the SCS {1, 1, 1, 1, 1, 0, 1}, i.e. all AMR modes except the 5.15 kbps shall be treated as supported and the OM shall be assumed to be "Optimisation of the ACS supported". For the other "F" configurations the ACS and SCS shall be assumed to be identical and as shown in the configuration table. The OM shall be assumed to be "Optimisation of the ACS not supported".

A change via Maximum Rate Control is always possible (e.g. from configurations 10, 11, 12, 13, 14, 15 to 9 and 8).

The "Y" in the table indicates, which Configuration is defined for which Codec Type.

7.11.3.1.4 AMR-WB specific Codec\_Attribute\_Head Extension\_Block

The AMR-WB specific Codec\_Attribute\_Head Extension\_Block is defined in Table 7.11.3.1.4-1.

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	<b>PAR_Sel</b>	Differentiates this Extension_Block <b>0</b> : Parameters included in <b>PAR</b> field: Simple Codec_List_Extension <b>1</b> : undefined
Bit 3..10	<b>CoID = FR_AMR-WB or UMTS_AMR-WB or OHR_AMR-WB or OFR_AMR-WB</b>	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:	<b>PAR</b>	AMR-WB configuration as defined in TS 26.103, table 5.7-1 (Config-WB-Code)
Bit 16..18:	<b>CRC</b>	3 CRC bits protecting Bits 2 to 10 and 12 to 15
Bit 19..20:	<b>EX</b>	The normal 2 bits for IS_Message Extension: 00: No other extension block follows 11: An other extension block follows

## 8 Time Alignment of TFO Frames and TFO Messages

### 8.1 Alignment of TFO Frames and TFO Messages for GSM

The relative TRAU Frame phase positions of the two TRAU's using TFO across the A interface are arbitrary and depend on the local timing structure of the relevant BTSs. These BTSs are typically not synchronised. The TFO Protocol can not and does not change this. The clock systems of the transmission channels are typically also not synchronised and octet slips may occur.

TFO Frames and embedded TFO Messages are always exactly aligned with each other and follow the uplink TRAU Frames with a small, negligible, constant delay ( $T_{ultfo}$ : some PCM samples).

For the Codec Types GSM\_FR, GSM\_HR and GSM\_EFR the time alignment procedures for the **downlink** TRAU Frames, as specified in 3GPP TS 48.060 (full rate traffic) and 3GPP TS 48.061 (half rate traffic) on the Abis/Ater interface, are not affected by the TFO procedures on the A interface. For these Codec Types the TRAU shall buffer the received TFO Frames until they fit into the downlink timing as commanded by the local BTS.

For the Codec Types FR\_AMR, HR\_AMR and FR\_AMR-WB the phase of the downlink TRAU Frame depends on the phase of the received TFO Frames. An AMR/AMR-WB TRAU does not follow the Time Alignment procedure, when TFO is established, but sends the received TFO Frames as soon as possible in downlink as TRAU Frames. Therefore the local BTS has to buffer the TRAU Frames accordingly until they fit for the transmission on the air interface.

#### 8.1.1 Time Alignment of TFO Messages in GSM

At start up of the TFO Protocol the first regular TFO Message is aligned to an uplink TRAU Frame in the same way as a TFO Frame or an embedded TFO Message would be aligned (see clause 8.1.2). Then, after that, all regular TFO Messages follow contiguously, without any phase shift in time alignment, until the first TFO Frame needs to be sent (in general after the TFO\_TRANS Message). Then, the required number of T\_Bits is inserted before the first TFO Frame, see clause 8.1.2. Consequently, all following embedded TFO Messages are always aligned with the TFO Frames in a way, that the first bit of any TFO Messages is placed into the LSB of the first sample of a TFO Frame. Due to this definition, embedded TFO Messages only modify some of the synchronisation bits of the TFO Frames and the EMBED bit.

#### 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 ( $T_{ultfo}$ ) 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  $T_{ultfo}$  is defined or need to be defined.

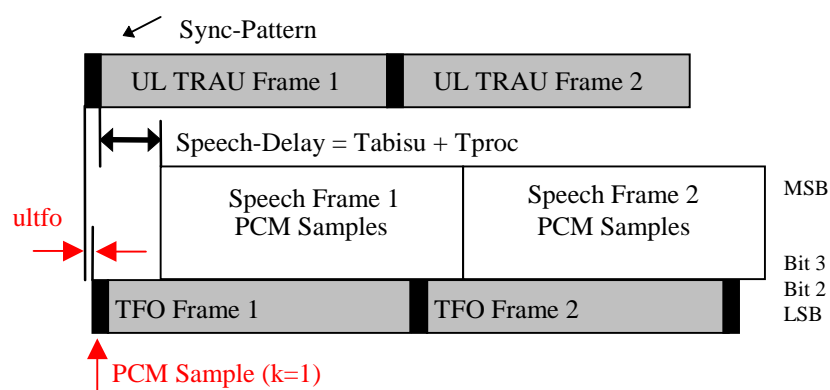


Figure 8.1.2-1: Uplink TFO Frame Time Alignment in GSM

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.

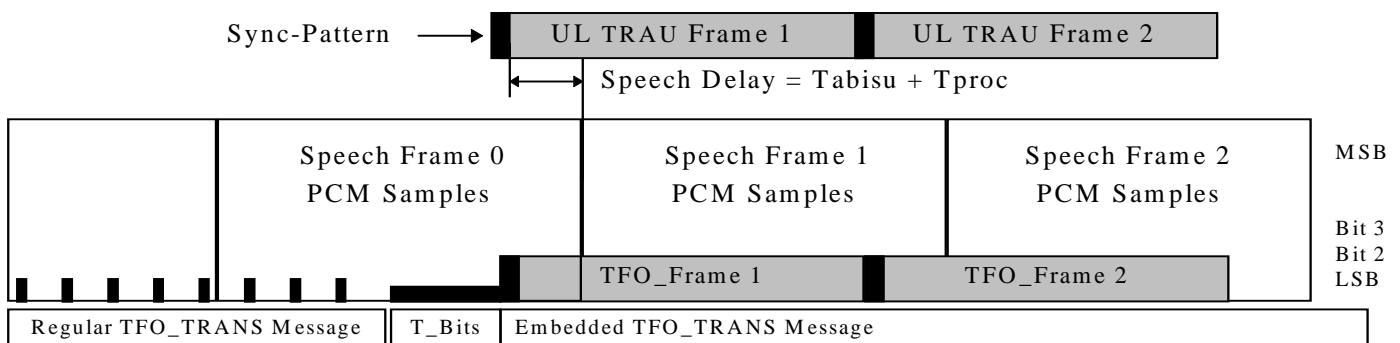
On the transition between the sending of regular TFO Messages and the first TFO Frame, a sufficient number (up to a maximum of 159) of Time Alignment Bits, also called "T\_Bits", are inserted into the LSBs of the PCM samples to align the TFO Frame as described above.

This insertion of Time Alignment Bits (if necessary) is started exactly with the 16<sup>th</sup> PCM sample after the last bit of the last regular TFO Message (i.e. the TFO\_TRANS Message).

Whenever, in a later stage, the phase of the uplink TRAU Frame changes, then again T\_Bits need to be inserted between two consecutive TFO Frames or deleted from the tail of the last TFO Frame to ensure proper alignment.

The insertion of T\_Bits as a result of timing changes shall occur *between* TFO Frames and not within TFO Frames.

If the time alignment is necessary while a TFO Message is embedded into a series of TFO Frames, then the TFO Message may be cut into two parts with the T\_Bits in between. Therefore, whenever an adjustment of the phase of the TFO Frames is necessary, then one additional TFO Message shall be embedded into the next TFO Frames (after the possibly ongoing TFO Message). If nothing else is to be transmitted, then the TFO\_FILL Message shall be used. One TFO\_TRANS Message is *always* embedded into the first TFO Frames. See the following Figure 8.1.2-2:



**Figure 8.1.2-2: Time Alignment by inserting T\_Bits and embedding one TFO\_TRANS Message**

### 8.1.3 Time Alignment of TFO Frames to Downlink TRAU Frames

For the Codec Types GSM\_FR, GSM\_HR and GSM\_EFR the TFO Protocol does not affect the phase position of the downlink TRAU frames.

The phase difference between the received TFO Frames and the downlink TRAU Frames is in general constant, but arbitrary between 0 and 159 PCM samples. The time alignment of the TFO Frames to the downlink TRAU Frames must therefore be managed by buffering the TFO Frames within the receiving downlink TRAU. This can be done in one of two methods:

**Method 1:** The received TFO Frame is buffered for a period between 0 to 159 PCM samples in addition to the processing delay (*T<sub>b<sub>fh</sub></sub>*) required to perform a suitable *Bad Frame Handling on parameter level*. Transmission of the downlink TRAU Frame may in this case begin *prior* to receipt of the complete TFO Frame.

NOTE 1: In this first method the overall one way signal delay will be between 30 ms and 10 ms lower than the delay in normal tandem connections.

**Method 2:** Alternatively the received TFO Frame is buffered for a period between 160 to 319 PCM samples in addition to the processing delay required to perform a suitable *Bad Frame Handling on parameter level* (*T<sub>b<sub>fh</sub></sub>*). Transmission of the downlink TRAU Frame will in this case always begin *after* the receipt of the complete TFO Frame.

NOTE 2: In this second method the overall one way signal delay will always be up to 10ms lower or up to 10 ms higher than the delay in normal tandem connections.

NOTE 3: The two methods differ in one way signal delay always by exactly 20 ms. Figure 8.1.3-1 highlights the relations for an arbitrarily selected relative phase difference between TFO and TRAU Frames of 80 samples (10 ms). *Tbfh* is in the order of some PCM samples only, if error concealment is done "in advance" based on the parameters of the previous TFO Frame, before the actual TFO Frame is even received.

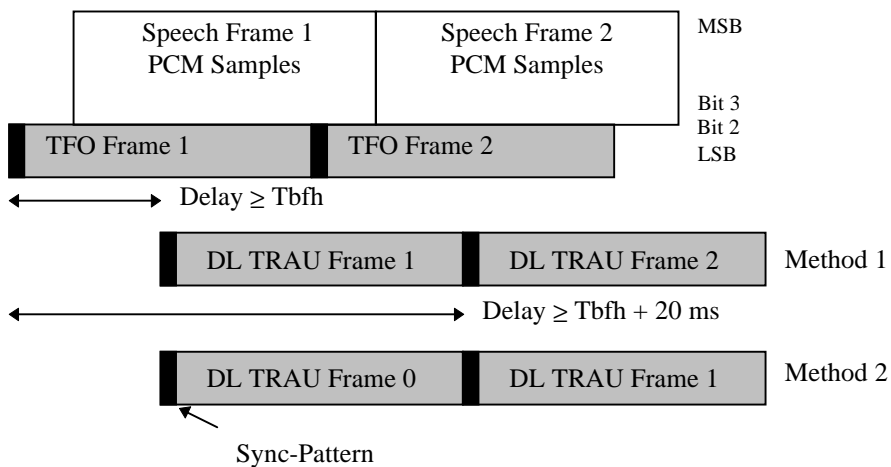


Figure 8.1.3-1: Downlink Time Alignment of TFO Frames in GSM

For the Codec Types FR\_AMR, HR\_AMR and FR\_AMR-WB no error concealment is necessary within the downlink TRAU. The received TFO Frames are passed as soon as possible downlink as TRAU Frames, without considering the previous phase of the TRAU Frames.

**General:** TRAU Frames shall always be sent as complete TRAU Frames.

The transition from normal Tandem Operation to Tandem Free Operation shall be done by inserting the necessary number of T-Bits between the previous - time aligned TRAU Frame - and the new - TFO aligned TRAU Frame. By this the BTS does not loose synchronisation. The signal delay within the TRAU is kept at minimum. The BTS has to buffer the received TRAU Frames until they fit for transmission on the air interface. Time Alignment and phase alignment are discontinued as long as the BTS is in States TFO\_MAYBE, TFO\_YES or TFO\_TERM, see Annex C.

In case TFO is terminated the transition from TFO aligned TRAU Frames back to time aligned TRAU Frames shall be done in the following way: The first TRAU Frames after TFO is terminated shall be sent in exactly the same phase as the TFO aligned TRAU Frames. Then the BTS will re-start the time alignment procedure and command time and phase alignments. Then the necessary number of T-Bits shall be inserted between the TFO aligned TRAU Frames and the time aligned TRAU Frames.

## 8.2 Time Alignment of TFO Frames and TFO Messages for 3G

There is no requirement for the Time Alignment of TFO Frames and the Iu User Plane. However, all implementation should minimise the transmission delay between Iu User Plane PDUs and TFO Frames in the uplink and the downlink directions.

TFO Frames and embedded TFO Messages shall always be exactly aligned with each other and follow the uplink with minimal delay.

### 8.2.1 Time Alignment of TFO Messages in 3G

At start up of the TFO Protocol the first regular TFO Message is aligned to the uplink Iu frames in the same way as a TFO Frame or an embedded TFO Message would be aligned (see clause 8.2.2). Subsequently, all regular TFO Messages follow contiguously, without any phase shift in time alignment, until the first TFO Frame needs to be sent (in general after the TFO\_TRANS Message). Then, the required number of T\_Bits is inserted before the first TFO Frame, see clause 8.2.2.

Consequently, all following embedded TFO Messages are always aligned with the TFO Frames in a way, that the first bit of any TFO Messages is placed into the LSB of the first sample of a TFO Frame. Due to this definition, embedded TFO Messages only affect some of the synchronisation bits of the TFO Frames and the EMBED bit.

## 8.2.2 Time Alignment of TFO Frames to Uplink Iu Frames

The contents of the Uplink Iu User Plane PDU undergo a variable delay (Tultfo) required to perform the generation of the necessary framing bits (control and Sync) and also to ensure the continuous flow of TFO Frames. It is important that this is optimised to remove the jitter from the uplink Iu frame reception to ensure a constant and continuous play-out of TFO Frames to the distant partner.

On the transition between the sending of regular TFO Messages and the first TFO Frame, a sufficient number (up to a maximum of 159) of Time Alignment Bits, also called "T\_Bits", are inserted into the LSBs of the PCM samples to align the TFO Frame as described above.

This insertion of Time Alignment Bits (if necessary) is started exactly with the 16<sup>th</sup> PCM sample after the last bit of the last regular TFO Message (i.e. the TFO\_TRANS Message).

Whenever, in a later stage, it is necessary to alter the play-out timing, then again T\_Bits need to be inserted *between* two consecutive TFO Frames or deleted from the tail of the last TFO Frame to ensure proper alignment.

If the adjustment is necessary while a TFO Message is embedded into a series of TFO Frames, then the TFO Message may be cut into two parts with the T\_Bits in between. Therefore, whenever an adjustment of the phase of the TFO Frames is necessary, then one additional TFO Message shall be embedded into the next TFO Frames (after the possibly on-going TFO Message). If nothing else is to be transmitted, then the TFO\_Fill Message shall be used. One TFO\_TRANS Message is *always* embedded into the first TFO Frames.

## 8.2.3 Time Alignment of TFO Frames to Downlink Iu Frames

The Transcoder should wait for the complete reception of a TFO Frame and send a corresponding Iu UP PDU with the minimum buffering delay to perform the required conversion between TFO Frames and Iu UP Frames as defined in clause 5.

# 9 TFO State Machine

A State Machine, consisting of 17 States can describe the TFO\_Protocol Process, see the following figure.

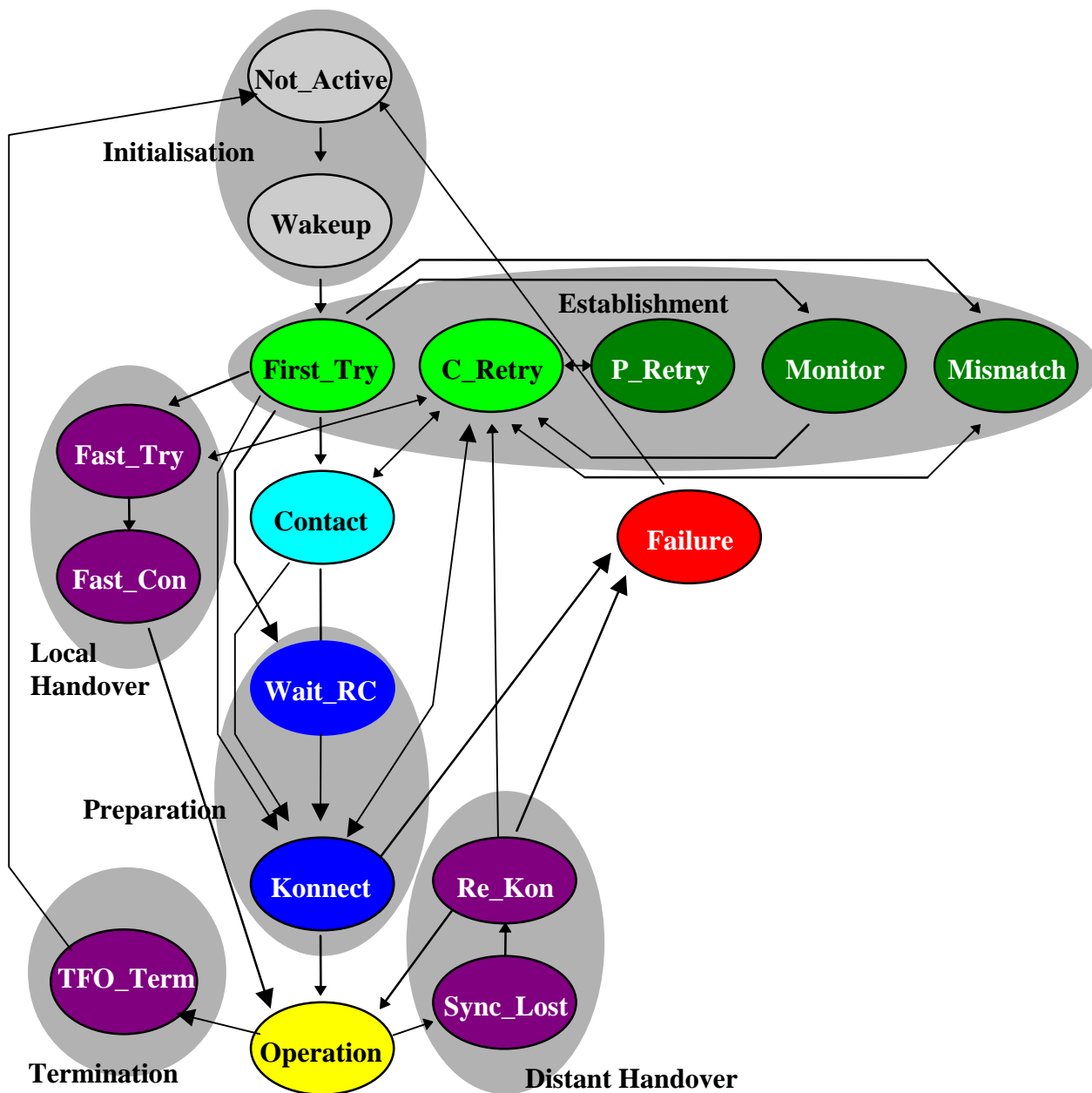


Figure 9-1: TFO\_Protocol State Machine with most important transitions

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.

## 9.1 Initialisation

### 9.1.1 Not\_Active State

The Not\_Active state shall be the initial state of the TFO\_Protocol. In this state the TFO\_Protocol is not active and the TRAU/TC works in a conventional way. The state Not\_Active is left and a transition to the Wakeup state is performed when a new speech call is set up or/and when TFO gets enabled.

### 9.1.2 Wakeup State

In the Wakeup state the TFO\_Protocol waits until PCM speech samples are received that are different from PCM\_Idle. Then a transition to the First\_Try state is performed and three TFO\_FILL messages and some TFO\_REQ messages are initiated.

## 9.2 Establishment

### 9.2.1 First\_Try State

The TC enters the First\_Try state from the Wakeup state if TFO was enabled and PCM\_Non\_Idle speech samples are received. Regular TFO\_REQ Messages are sent continuously for a certain maximum time. After that, if no TFO Partner answers before a Runout of TFO Messages, TFO\_Protocol enters automatically into the Monitor State.

If TFO\_REQ Messages are received with the same, own Signature, then a circuit loop back is assumed, i.e. the call is still not through connected. The TC selects a new Signature and continues sending TFO\_REQ Messages, until a different Signature is received or a TFO\_ACK is received. Since loop back delays may be substantial in some cases, the TC has to remember and compare also the previously selected own Signature. Care must be taken that the Signature selection contains a true random element to avoid that two different TCs select by coincidence identical signatures again and again.

When the TC receives a TFO\_REQ with an appropriate signature and TFO is possible, it enters the Contact State.

If the TC receives a TFO\_ACK to a previously sent TFO\_REQ, TFO\_Protocol enters the Mismatch State, if immediate TFO establishment is not possible.

If immediate TFO establishment is possible, TFO\_Protocol enters directly the Konnect State in the case of Non\_AMR Codec Types. If immediate TFO establishment is possible in case of an AMR or AMR-WB Codec Type, the TFO\_Protocol enters the Wait\_RC State, before it goes on to the Konnect State.

If the TC receives TFO Frames in the First\_Try State, it should assume that a TFO might have been established previously and was recently broken because of a local handover. The TC should then enter the Fast\_Try State.

### 9.2.2 Continuous\_Retry State

In this state, TFO Contact has existed either by TFO Messages or by TFO Frames, but was interrupted and sync was lost. The TC sends a maximum number of regular TFO\_REQ Messages continuously in order to test, if TFO could be re-established. In case of Runout of TFO messages, the TFO\_Protocol enters the Periodic\_Retry State.



### 9.2.3 Periodic\_Retry State

The Periodic\_Retry state is typically entered from Continuous\_Retry in the case of Runout of TFO messages. The TFO\_Protocol tests from time to time by sending a single TFO\_REQ\_L message, if TFO could be re-established. As soon as a TFO Message is received, TFO\_Protocol leaves this State.

NOTE: Since no contiguous transmission of TFO Messages is ongoing, possible IPEs may be unsynchronised.

### 9.2.4 Monitor State

In this state the TC monitors the PCM samples for TFO messages or TFO Frames, but it does not send any TFO messages or TFO frames. As soon as a TFO message has been received from a distant partner, the TC knows that a TFO Partner exists. Moreover, it knows that the transmission path from the distant partner is digitally transparent. The TC may already now see, whether TFO is possible, but it must ensure that all IPEs are synchronised. It therefore transits into the Continuous\_Retry state. If no TFO is possible, the TFO\_Protocol informs its local BSS/RAN and transits into the Mismatch state by sending back TFO\_REQ\_L messages.

NOTE: Since no contiguous transmission of TFO Messages is ongoing, possible IPEs may be unsynchronised.

### 9.2.5 Mismatch State

In this state it is obvious from a previous contact that a distant TFO Partner exists, but TFO establishment was not possible because of incompatible codec types or codec configurations. The TC waits without sending TFO messages or TFO frames until the mismatch situation is resolved.

NOTE: Since no contiguous transmission of TFO Messages is ongoing, possible IPEs may be unsynchronised.

## 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.

## 9.4 Preparation

### 9.4.1 Wait\_RC State

This State exists only when the local used Codec Type is an AMR or AMR-WB Codec. For all other Codec Types this State is not entered and all transitions go instead directly into Konnect State.

The state WAIT\_RC is typically entered when a TFO\_ACK message is received in Contact State. Rate control is done. In GSM, a TFO\_Soon message is sent to the BTS. In 3G a Rate Control command is sent to the RNC.

In this Wait\_RC State the TFO\_Protocol waits for the acknowledgement from the BTS / RNC that the Rate Control Command has been received and executed. Then the TC sends TFO\_TRANS to bypass the IPEs, starts sending TFO Frames and TFO\_Protocol transits into the Konnect State.

## 9.4.2 Konnect State

In the Konnect state the TC sends TFO Frames and possibly embedded TFO Messages as long as it receives correct TFO Messages.

The first received TFO Frame causes the transition into the Operation State.

If no TFO Frames are received within a certain period, the TC transits to the Failure State.

## 9.5 Operation State

In this State - the Main State of TFO\_Protocol - the TC sends and receives TFO Frames, thus the TFO Connection is fully operating. TFO Messages may occur embedded into TFO Frames.

## 9.6 Local Handover

### 9.6.1 Fast\_Try State

When the TC is in First\_Try and suddenly receives TFO Frames and the Codecs do match, then there is a high probability that a local handover has initialised the TC into an existing TFO connection and a fast TFO establishment is likely. The TFO\_Protocol has still to check, whether the link to the distant TFO Partner is (already) transparent. This is done by the specific TFO\_DUP Message.

Since the handover must have been a local handover, i.e. close to the (new) TC, it can be assumed that the possibly existing IPEs are still in transparent mode and TFO Messages therefore pass through directly.

### 9.6.2 Fast\_Contact State

This State is entered from First\_Try via Fast\_Try, if TFO Frames and then TFO\_SYL Messages are received. The TC continues to send TFO\_DUP Messages, until TFO Frames are received again. Then it immediately starts to send TFO Frames, with a TFO\_TRANS embedded into the first TFO Frames. The TC transits directly to Operation State.

## 9.7 Distant Handover, TFO Interruption

### 9.7.1 Sync\_Lost State

If the TC was in Operation State and suddenly the TFO Frame synchronisation is lost, then the TC enters the Sync\_Lost State for a short while, before it transits to Continuous\_Retry.

If synchronisation was lost due to a distant handover, then a fast TFO establishment might be possible and the TC enters Operation State soon again. In Sync\_Lost it expects TFO\_DUP Message as confirmation of the distant handover. Then it transits to Re\_Konnect.

### 9.7.2 Re\_Konnect State

This State is entered from Operation via Sync\_Lost, if TFO\_DUP Messages are received. The TC starts immediately to send TFO Frames again, with a TFO\_TRANS embedded into the first TFO Frames. The TC transits back to Operation State, as soon as TFO Frames are received, again.

### 9.7.3 TFO\_Term

This State is entered when TFO is disabled by either the local or distant TRAU/TC. The TRAU/TC stops then sending TFO frames but still accepts receiving TFO frames and messages sent by the distant TRAU/TC.

When the TFO termination has been initiated locally the TRAU/TC transits through this state before entering to Not\_Active state after the TFO termination has been acknowledged by the distant side.

When the TFO termination has been initiated by the distant TRAU/TC, the TRAU/TC enters in MONITOR state when TFO frames are no more received.

## 9.8 Failure State

This State is entered when the distant partner shows an incorrect behaviour. The TC then sends pure PCM samples and waits for the failure to disappear. It does not send TFO Frames or TFO Messages.

---

# 10 Detailed Description of the TFO Protocol

## 10.1 Syntax Used for the TFO\_Protocol Description

The TFO\_Protocol Process is always in one of the states defined in clause 9. It is fully described by the set of Tables in clause 10.6 defining the required **Actions** and state **Transitions** triggered by all relevant **Events**. The syntax used for this description is showed in Table 10.1-1. The **Events** are the Column entries, while the different states are listed as Rows entries.

**Table 10.1-1: Definition of the Syntax for the State Machine Description**

Event: Or Number	<Received Message> <Received Message> <running number>	...	<Other Event> <Other Event> <running number>
Condition: &	[<Condition>] [<Condition>]		[<Condition>] [<Condition>]
Comment:	[<Comment>] [<Comment>]		[<Comment>] [<Comment>]
<b>State:</b> <Actual State>:	<Action Name>;[<Action Name>;] <Next State>; [<Comment>]		<Action Name>;[<Action name>;] <Next State>; [<Comment>]
...			
<Actual State>:	<Action Name>;[<Action Name>;] <Next State>; [<Comment>]		<Action Name>;[<Action Name>;] <Next State>; [<Comment>]

## 10.2 Detailed Description of the Conditions

For a short notation the following abbreviations are used in the conditions row of the TFO protocol tables:

### 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 5.3 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.

## 10.2.2 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 ( $Duc \neq Luc$ ) 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 ( $Duc \neq Luc$ ) 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  $n > 2$ .

## 10.3 Abbreviations, Definitions, Notations used in the TFO\_Protocol Description

The following Abbreviations and Definitions are used in the TFO\_Protocol Tables.

**Local\_Used\_Codec** (short form: **Luc**) refers to the Speech Codec Type used in the local transcoder and RAN (e.g. GSM\_FR, GSM\_EFR, GSM\_HR, FR\_AMR, HR\_AMR, OHR\_AMR, UMTS\_AMR or UMTS\_AMR\_2, FR-AMR-WB, UMTS\_AMR-WB, OFR\_AMR-WB or OHR\_AMR-WB).

**Distant\_Used\_Codec (Duc)** refers to the Speech Codec Type used by the distant partner, as reported in TFO\_REQ... or TFO\_ACK (e.g. GSM\_FR, GSM\_EFR, GSM\_HR, FR\_AMR, HR\_AMR, UMTS\_AMR, OHR\_AMR, UMTS\_AMR\_2, FR-AMR-WB, UMTS\_AMR-WB, OFR\_AMR-WB, OHR\_AMR-WB).

All these variables are initialised to **UNKNOWN**, which means that the content of the variables is not defined.

**Local\_Signature (Lsig)** refers to the 8-bit random number in TFO\_REQ, which identifies the local TFO\_REQ Messages. It is also used in TFO\_REQ\_L.

**Distant\_Signature (Dsig)** refers to the 8-bit random number as received in TFO\_REQ, TFO\_REQ\_L, TFO\_ACK and TFO\_ACK\_L. If received in TFO\_REQ or TFO\_REQ\_L, it should be different from the Local\_Signature, otherwise loop back must be assumed (exceptions exist). If received in TFO\_ACK or TFO\_ACK\_L, then it should be identical to the Local\_Signature, otherwise the TFO\_ACK is not a response to an own TFO\_REQ, but was possibly created during an handover situation.

**Local Channel Type (LCh)** and **Distant Channel Type (DCh)** refer to the 8 or 16 kbit/s or 32kbit/s transparent channel used by the local Transmission process or received through the distant TFO\_TRANS.

**Error protection** and error handling: It is assumed that the defined error protection is strong enough for the error rates encountered on typical transmission links. The few occurring errors are usually all detected and possibly corrected by Rx\_TFO, before reported to TFO\_Protocol. Therefore TFO\_Protocol can rely on the correctness of the received Events. The protocol is, however, "self healing" and will handle the unlikely erroneous Events.

**Fast Handover** handling: The defined protocol assumes that the new Transcoder, to which the handover is performed, is already in State Wakeup before the A-Interface is switched to that Transcoder. Only then, the TFO Frames can be received and fast handover handling is possible.

**Timing:** If two Events occur by coincidence at the same time, then they shall be processed in the order given by the tables 10.6-1 to 10.6-13 (left to right). TFO Messages arrive always some time before the embedding TFO Frame and shall be handled therefore first.

## 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 lu 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, 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, 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, the local codec configuration changes, 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, the local codec configuration changes, 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.
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, 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, 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, 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, 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, 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.
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 occurs 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 occurs 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 when the local codec configuration changes and Immediate Codec Type Optimisation is performed.

## 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 **First-In First-Out** 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  $T_c$ ).

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

**Table 10.6-1: Enabling/Disabling/New\_Speech\_Call/TRAU\_Idle**

Event: or Number:	TFO_Enable New_Speech_Call	TFO_Disable TRAU_Idle
Condition: &		
Comment:	TFO gets active.	Local disable.
<b>State:</b>		
<b>NAC:</b> Not_Active	C;S;IT;RCm; WAK;	NoAc; NAC;
<b>WAK:</b> Wakeup	NoAc; WAK;	NoAc; NAC;
<b>FIT:</b> First_Try	----- -----	C;N; NAC;
<b>COR:</b> Continuous Retry	----- -----	C;N; NAC;
<b>PER:</b> Periodic Retry	----- -----	C;N; NAC;
<b>MON:</b> Monitor	----- -----	C;N; NAC;
<b>MIS:</b> Mismatch	----- -----	C;N; NAC;
<b>CON:</b> Contact	----- -----	C;N; NAC;
<b>FAT:</b> Fast Try	----- -----	C;N;RCm; NAC;
<b>FAC:</b> Fast Contact	----- -----	C;N;RCm; NAC;
<b>WRC:</b> Wait_RC	----- -----	C;N;RCm; NAC;
<b>KON:</b> Konnnect	----- -----	C;RCm;CR;DT;N;T1; TT;
<b>REK:</b> Re_Konnnect	----- -----	C;RCm;CR;DT;N;T1; TT;
<b>SOS:</b> Sync_Lost	----- -----	C;RCm;IT;N; NAC;
<b>OPE:</b> Operation	----- -----	C;RCm;CR;DT;N;T1; TT;
<b>FAI:</b> Failure	----- -----	C; NAC; <b>Exit from FAI</b>
<b>TT:</b> TFO_Term	----- -----	NoAc; 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:	Occurs only at the beginning	Loopback (LB) or distant handover (HO)? wrong Sig	Loopback (LB) or distant handover (HO)?
<b>State:</b>			
<b>NAC:</b> Not_Active	----- -----	----- -----	----- -----
<b>WAK:</b> Wakeup	C;F;REQ; FIT; <b>Typ 2<sup>nd</sup> Event</b>	----- -----	----- -----
<b>FIT:</b> First_Try	----- -----	C;SO;REQ; FIT; <b>LB!</b>	NoAc; FIT; <b>Ignore LB</b>
<b>COR:</b> Continuous Retry	----- -----	C;SO;REQ; COR; LB!?	NoAc; COR; <b>Ignore LB</b>
<b>PER:</b> Periodic Retry	----- -----	C;F;S;ACK; CON; Dist HO!	----- -----
<b>MON:</b> Monitor	----- -----	C;F;S;REQ; FIT; Dist HO!	----- -----
<b>MIS:</b> Mismatch	----- -----	C;F;S;ACK; CON; Dist HO!	----- -----
<b>CON:</b> Contact	----- -----	C;SO;REQ; COR; Safe way	----- -----
<b>FAT:</b> Fast Try	----- -----	C;SO;REQ;RCm; COR; Safe way	----- -----
<b>FAC:</b> Fast Contact	----- -----	C;SO;REQ;RCm; COR; Safe way	----- -----
<b>WRC:</b> Wait_RC	----- -----	C;SO;RCm;REQ; COR;	----- -----
<b>KON:</b> Konnnect	----- -----	C;DT;SO;RCm;REQ;T1; COR; IPEs transparent!	----- -----
<b>REK:</b> Re_Konnnect	----- -----	C;DT;SO;RCm;REQ;IT;B;T1; COR; IPEs transparent!	----- -----
<b>SOS:</b> Sync_Lost	----- -----	C;IT;S;RCm;REQ;B;T1; COR; Contact is back	----- -----
<b>OPE:</b> Operation	----- -----	----- -----	----- -----
<b>FAI:</b> Failure	----- -----	NoAc; FAI;	----- -----
<b>TT:</b> 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 ICO==0	NA_TP Dsig==Lsig	(NA_TP   A_TP) Dsig!=Lsig	Luc != AMR DCh==LCh	Match_1
Comment:	Distant REQ Good Signature	Distant ACK Good Signature	Wrong Response Handover?	similar to ACK As response to loc ACK_?	First or second TFO Frame
<b>State:</b>					
<b>NAC:</b> Not_Active	----- -----	----- -----	----- -----	----- -----	----- -----
<b>WAK:</b> Wakeup	----- -----	----- -----	----- -----	----- -----	----- -----
<b>FIT:</b> First_Try	C;U;ACK; CON; <b>Typical</b>	C;U;T;BT;T;T1; KON; <b>Typical; IPES!</b>	C;REQ; FIT;	NoAc; FIT; Wait for Frame	C;U;DUP;RCi; FAT; <b>1: HO</b>
<b>COR:</b> Continuous Retry	C;U;ACK; CON; Typical	C;U;T;BT;T;T1; KON; <b>Typical; IPES!</b>	C;REQ; COR;	NoAc; COR; Wait for Frames	C;U;DUP; FAT; 1: Call is back?
<b>PER:</b> 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?
<b>MON:</b> 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?
<b>MIS:</b> 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?
<b>CON:</b> Contact	C;ACK; CON; <b>Typical: wait</b>	C;T;BT;T;T1; KON; <b>Typical: yes!</b>	C;REQ; COR;	C;T;BT;T;T1; KON; <b>yes! Fast way</b>	C;T;BT;T;T1; KON; Missed TRANS?
<b>FAT:</b> 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; <b>2: Typ. Loc HO</b>
<b>FAC:</b> 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; <b>5: Typ. Loc HO</b>
<b>WRC:</b> Wait_RC	C;RCm;REQ;T1; COR;	----- -----	C;RCm;REQ; COR;	----- -----	AT; WRC;
<b>KON:</b> Konnnect	C;RCm;DT;REQ;T1; COR; IPES transparent!	NoAc; KON; <b>Typical: wait</b>	NoAc; KON;	NoAc; KON; <b>Typical: wait</b>	RCs;AT;L;T2;B; OPE; <b>Typ: call set-up</b>
<b>REK:</b> Re_Konnnect	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; <b>5: Typ. Dis HO</b>
<b>SOS:</b> 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?
<b>OPE:</b> Operation	----- -----	----- -----	----- -----	NoAc; OPE; Typical in HO	NoAc; OPE; <b>Main! TFO!</b>
<b>FAI:</b> Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
<b>TT:</b> TFO_Term	----- -----	----- -----	----- -----	----- -----	----- -----

Table 10.6-4: In Call Modification and Handover

Event: or Number:	New_Local_Codec New_Local_Config	New_Local_Codec New_Local_Config	TFO_Frame	TFO_SYL	TFO_DUP
13, 14		15, 16	17	18	19
Condition: &	(NA_TP   A_TP) ICO==0	TM ICO==0	Match_2		
Comment: <b>State:</b>	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
<b>NAC:</b> Not_Active	----- -----	----- -----	----- -----	----- -----	----- -----
<b>WAK:</b> Wakeup	NoAc; WAK;	NoAc; WAK;	----- -----	----- -----	----- -----
<b>FIT:</b> First_Try	C;REQ; FIT; Restart	C;REQ; FIT; Restart	----- -----	NoAc; FIT; HO? Ignore	NoAc; FIT; HO? Ignore
<b>COR:</b> Continuous Retry	C;REQ; COR;	C;REQ; COR;	----- -----	NoAc; COR; Ignore	NoAc; COR; Ignore
<b>PER:</b> Periodic Retry	L1;T5; PER;	L1;T5; PER;	----- -----	C;F;REQ; COR; Rare case, test	C;F;REQ; COR; Rare case, test
<b>MON:</b> Monitor	NoAc; MON;	NoAc; MON;	----- -----	C;F;REQ; FIT; Rare case, test	C;F;REQ; FIT; Rare case, test
<b>MIS:</b> Mismatch	C;F;REQ; COR; <b>Mismatch Res.</b>	C;L;T2;B; MIS; <b>Direct info</b>	----- -----	C;F;REQ; COR; Rare case, test	C;F;REQ; COR; Rare case, test
<b>CON:</b> Contact	C;REQ; COR;	C;L;T2;B; MIS;	----- -----	C;F;REQ; COR; Rare case, test	C;F;REQ; COR; Rare case, test
<b>FAT:</b> Fast Try	NoAc; FAT;	C;L;T2;B;RCm; MIS;	NoAc; FAC;	NoAc; FAC; <b>3: Typ. Loc HO</b>	C;F;REQ;RCm; COR; Rare case, test
<b>FAC:</b> Fast Contact	NoAc; FAC;	C;L;T2;B;RCm; MIS;	C;BT;T;L;T2;AT;B;RCs; OPE; assume matching ACS	NoAc; FAC; <b>4: Typ Loc HO</b>	C;F;REQ;RCm; COR; rare case, test
<b>WRC:</b> Wait_RC	C;RCm;REQ; COR;	C;RCm;L;T2;B; MIS;	NoAc; WRC;	NoAc; WRC;	NoAc; WRC;
<b>KON:</b> 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?
<b>REK:</b> 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; <b>4: Typ. Dist HO</b>
<b>SOS:</b> Sync_Lost	C;RCm;IT;REQ; COR;	C;RCm;IT;L;T2;B; MIS;	----- -----	NoAc; SOS; Short Interrupt.?	C;BT;T;T1; REK; <b>3: typ Dis HO</b>
<b>OPE:</b> Operation	RCs;L;T2; OPE;	C;RCm;DT;IT;L;T2;B; MIS;	NoAc; OPE; Main! TFO!	NoAc; OPE; Short interrupt?	NoAc; OPE; Typical
<b>FAI:</b> Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
<b>TT:</b> 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: &	(NA_TP   A_TP) Dsig==Lsig	(NA_TP   A_TP) Dsig!=Lsig	(NA_TP   A_TP) Dsig==Lsig	(NA_TP   A_TP) Dsig!=Lsig
Comment:	Only sent in MIS/OPE/PER HO? Loop?	Only sent in MIS/OPE/PER Codec_List	Only sent in MIS; HO?	HO?
<b>State:</b>				
<b>NAC:</b> Not_Active	----- -----	----- -----	----- -----	----- -----
<b>WAK:</b> Wakeup	----- -----	----- -----	----- -----	----- -----
<b>FIT:</b> First_Try	NoAc; FIT; Ignore	NoAc; FIT; Ignore	NoAc; FIT; Ignore	NoAc; FIT; Ignore
<b>COR:</b> Continuous Retry	NoAc; COR; Ignore	NoAc; COR; Ignore	NoAc; COR; Ignore	NoAc; COR; Ignore
<b>PER:</b> 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
<b>MON:</b> Monitor	C;F;S;REQ; FIT; Test	C;F;REQ; FIT; Test	C;F;S;REQ; FIT; Test	C;F;REQ; FIT; Test
<b>MIS:</b> Mismatch	C;F;S;REQ; COR; Test	C;F;REQ; COR; Test	C;F;S;REQ; COR; Test	C;F;REQ; COR; Test
<b>CON:</b> Contact	C;S;REQ; COR; Safe way!	C;REQ; COR; Safe way!	C;S;REQ; COR; Safe way!	C;REQ; COR; Safe way!
<b>FAT:</b> 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!
<b>FAC:</b> 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!
<b>WRC:</b> Wait_RC	C;S;RCm;REQ; COR;	C;RCm;REQ; COR;	C;S;RCm;REQ; COR;	C;RCm;REQ; COR;
<b>KON:</b> Konnnect	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!
<b>REK:</b> Re_Konnnect	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!
<b>SOS:</b> 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!
<b>OPE:</b> 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
<b>FAI:</b> Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
<b>TT:</b> 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 ICO==0	TM Dsig=? ICO==0	TM Dsig==Lsig	TM Dsig!=Lsig	TM Dsig==?
Comment: <b>State:</b>	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
<b>NAC:</b> Not_Active	----- -----	----- -----	----- -----	----- -----	----- -----	----- -----
<b>WAK:</b> Wakeup	----- -----	----- -----	----- -----	----- -----	----- -----	----- -----
<b>FIT:</b> First_Try	C;S;L;T2;B; MIS; Rare	C;U;L;T2;B; MIS; <b>Typical: Setup</b>	C;U;L;T2;B; MIS; HO?	C;S;LA;B; MIS; rare	C;U;LA;B; MIS; <b>Typical: Setup</b>	C;U;LA;B; MIS; HO?
<b>COR:</b> 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;
<b>PER:</b> 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;
<b>MON:</b> 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;
<b>MIS:</b> 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; <b>Terminate Prot.</b>	C;LA;B; MIS; <b>Terminate Prot.</b>
<b>CON:</b> 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;
<b>FAT:</b> 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;
<b>FAC:</b> 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;
<b>WRC:</b> 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;
<b>KON:</b> 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;
<b>REK:</b> 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;
<b>SOS:</b> 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; <b>In_Call Mod</b>	C;RCm;LA;IT;B; MIS;
<b>OPE:</b> Operation	----- -----	----- -----	----- -----	NoAc; OPE; Trans Error?	NoAc; OPE; Trans Error?	----- -----
<b>FAI:</b> Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
<b>TT:</b> 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:		Good Sig Immediate TFO possible	BTS has steered the mode.
<b>State:</b>			
<b>NAC:</b> Not_Active	----- -----	----- -----	NoAc; NAC;
<b>WAK:</b> Wakeup	----- -----	----- -----	NoAc; WAK;
<b>FIT:</b> First_Try	NoAc; FIT; Wait for Frame	C;U;RCi;ACK;T1; WRC; <b>Typical;</b>	NoAc; FIT;
<b>COR:</b> Continuous Retry	NoAc; COR; Wait for Frames	C;U;RCi;ACK;T1; WRC; <b>Typical</b>	NoAc; COR;
<b>PER:</b> Periodic Retry	NoAc; PER; Wait for Frames	C;F;S;REQ; COR; Rare case, test	NoAc; PER;
<b>MON:</b> Monitor	NoAc; MON; Wait for Frames	C;F;S;REQ; FIT; Rare case, test	NoAc; MON;
<b>MIS:</b> Mismatch	NoAc; MIS; Wait for Frames	C;F;S;REQ; COR; Rare case, test	NoAc; MIS;
<b>CON:</b> Contact	C;RCi;ACK;T1; WRC; Missed Ack	C;RCi;ACK;T1; WRC; <b>Typical</b>	NoAc; CON;
<b>FAT:</b> Fast Try	NoAc; FAC; Wait for Frames	C;REQ;RCm; COR; Safe way	NoAc; FAT;
<b>FAC:</b> Fast Contact	NoAc; FAC; Wait for Frames	C;REQ;RCm; COR; Safe way	NoAc; FAC;
<b>WRC:</b> Wait_RC	NoAc; WRC;	NoAc; WRC;	C; T;BT;T;T1; KON; <b>Typical</b>
<b>KON:</b> Konnect	NoAc; KON; <b>Typical: wait</b>	NoAc; KON; <b>Typical: wait</b>	NoAc; KON;
<b>REK:</b> Re_Konnect	NoAc; REK; Wait for Frames	C;DT;REQ;IT;B;T1; COR;	NoAc; REK;
<b>SOS:</b> Sync_Lost	NoAc; SOS; Wait for Frames	C;IT;REQ;B;T1; COR; Contact is back	NoAc; SOS;
<b>OPE:</b> Operation	NoAc; OPE; Typical in HO	----- -----	NoAc; OPE;
<b>FAI:</b> Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
<b>TT:</b> 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:	Local hand-over future parameters	Local hand-over future parameters
<b>State:</b>		
<b>NAC:</b> Not_Active	----- -----	----- -----
<b>WAK:</b> Wakeup	----- -----	----- -----
<b>FIT:</b> First_Try	C; NAC;	C; NAC;
<b>COR:</b> Continuous Retry	C; NAC;	C; NAC;
<b>PER:</b> Periodic Retry	C; NAC;	C; NAC;
<b>MON:</b> Monitor	C; NAC;	C; NAC;
<b>MIS:</b> Mismatch	C; NAC;	C; NAC;
<b>CON:</b> Contact	C; NAC;	C; NAC;
<b>FAT:</b> Fast Try	C;RCm; NAC;	C;RCm; NAC;
<b>FAC:</b> Fast Contact	C;RCm; NAC;	C;RCm; NAC;
<b>WRC:</b> Wait_RC	C;RCm; NAC;	C;RCm; NAC;
<b>KON:</b> Konnect	RCh; KON;	C;RCm;DT; NAC;
<b>REK:</b> Re_Konnect	RCh; REK;	C;RCm;DT;IT; NAC;
<b>SOS:</b> Sync_Lost	RCh; SOS;	C;RCm;IT; NAC;
<b>OPE:</b> Operation	RCh; OPE;	C;RCm;DT;T1; TT;
<b>FAI:</b> Failure	----- -----	----- -----
<b>TT:</b> 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:	Mismatch of channel type	Mismatch for one or two TFO Frames	Continued Mismatch
<b>State:</b>			
<b>NAC:</b> Not_Active	----- -----	----- -----	----- -----
<b>WAK:</b> Wakeup	----- -----	----- -----	----- -----
<b>FIT:</b> First_Try	C;U;L;T2;B; MIS; HO?	NoAc; FIT; HO? be tolerant	C;U;L;T2;B; MIS; <b>Typical in HO</b>
<b>COR:</b> Continuous Retry	C;U;L;T2;B; MIS;	NoAc; COR; Call Forw?	C;U;L;T2;B; MIS;
<b>PER:</b> Periodic Retry	C;F;L;T2;B; MIS;	NoAc; PER; Call Forw?	C;F;L;T2;B; MIS;
<b>MON:</b> Monitor	C;F;L;T2;B; MIS;	NoAc; MON; Call Forw?	C;F;L;T2;B; MIS;
<b>MIS:</b> Mismatch	C;L;T2;B; MIS;	NoAc; MIS; Call Forw?	C;L;T2;B; MIS;
<b>CON:</b> Contact	C;L;T2;B; MIS;	NoAc; CON;	C;L;T2;B; MIS;
<b>FAT:</b> Fast Try	C;L;T2;B;RCm; MIS;	NoAc; FAT;	C;L;T2;B;RCm; MIS;
<b>FAC:</b> Fast Contact	C;L;T2;B;RCm; MIS;	NoAc; FAC;	C;L;T2;B;RCm; MIS;
<b>WRC:</b> Wait_RC	C;RCm;L;T2;B; MIS;	NoAc; WRC;	C;RCm;L;T2;B; MIS;
<b>KON:</b> Konnect	C;RCm;DT;L;T2;B; MIS;	NoAc; KON;	C;RCm;DT;L;T2;B; MIS;
<b>REK:</b> Re_Konnect	C;RCm;DT;L;T2;IT;B; MIS;	NoAc; REK;	C;RCm;DT;L;T2;IT;B; MIS;
<b>SOS:</b> Sync_Lost	C;RCm;L;T2;IT;B; MIS;	NoAc; SOS;	C;RCm;L;T2;IT;B; MIS;
<b>OPE:</b> Operation	NoAc; OPE; Ignore?	NoAc; OPE; Hard HO?	C;RCm;DT;L;T2;IT;B; MIS; Hard HO into TFO
<b>FAI:</b> Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
<b>TT:</b> 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:	From RAN	In Call Modif. Stop TFO (see TFO_Disable)	Ignore is just Filler	Ignore alternative: Soft Reset
<b>State:</b>				
<b>NAC:</b> Not_Active	NoAc; NAC;	NoAc; NAC;	----- -----	----- -----
<b>WAK:</b> Wakeup	NoAc; WAK;	NoAc; NAC;	----- -----	----- -----
<b>FIT:</b> First_Try	NoAc; FIT; Update loc. Par.	C;N; NAC;	NoAc; FIT;	NoAc; FIT;
<b>COR:</b> Continuous Retry	NoAc; COR;	C;N; NAC;	NoAc; COR;	NoAc; COR;
<b>PER:</b> Periodic Retry	NoAc; PER;	C;N; NAC;	NoAc; PER;	NoAc; PER;
<b>MON:</b> Monitor	NoAc; MON;	C;N; NAC;	NoAc; MON;	NoAc; MON;
<b>MIS:</b> Mismatch	C;L;T2; MIS; <b>direct info</b>	C;N; NAC;	NoAc; MIS;	NoAc; MIS;
<b>CON:</b> Contact	NoAc; CON;	C;N; NAC;	NoAc; CON;	NoAc; CON;
<b>FAT:</b> Fast Try	NoAc; FAT;	C;N;RCm; NAC;	NoAc; FAT;	NoAc; FAT;
<b>FAC:</b> Fast Contact	NoAc; FAC;	C;N;RCm; NAC;	NoAc; FAC;	NoAc; FAC;
<b>WRC:</b> Wait_RC	NoAc; WRC;	C;N; NAC;	NoAc; WRC;	NoAc; WRC;
<b>KON:</b> Konnnect	NoAc; KON;	C;DT;N; NAC;	NoAc; KON;	NoAc; KON;
<b>REK:</b> Re_Konnnect	NoAc; REK;	C;DT;IT;N; NAC;	NoAc; REK;	NoAc; REK;
<b>SOS:</b> Sync_Lost	NoAc; SOS;	C;IT;N; NAC;	NoAc; SOS;	NoAc; SOS;
<b>OPE:</b> Operation	L;T2; OPE; <b>direct info</b>	C;DT;IT;N; NAC;	NoAc; OPE;	NoAc; OPE;
<b>FAI:</b> Failure	NoAc; FAI;	C; NAC; <b>exit from FAI</b>	NoAc; FAI;	NoAc; FAI;
<b>TT:</b> 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 TFO_Disabled	
Comment:	IPEs may become unsynchronised	Time-Out	start to send SYL already	Stop TFO Frames if 3 Frames missing	
<b>State:</b>					
<b>NAC:</b> Not_Active	----- -----	----- -----	----- -----	----- -----	----- -----
<b>WAK:</b> Wakeup	----- -----	----- -----	----- -----	----- -----	----- -----
<b>FIT:</b> First_Try	U;N; MON; <b>PSTN Call</b>	----- -----	----- -----	----- -----	NoAc; FIT;
<b>COR:</b> Continuous Retry	U;L1;T5; PER; <b>at end of COR</b>	C;N;REQ; COR; Reset IPEs	----- -----	----- -----	NoAc; COR;
<b>PER:</b> Periodic Retry	NoAc; PER;	L1;T5; PER; <b>Periodic Test</b>	----- -----	----- -----	NoAc; PER;
<b>MON:</b> Monitor	----- -----	C;N; MON;	----- -----	----- -----	----- -----
<b>MIS:</b> Mismatch	NoAc; MIS; <b>typ Final state</b>	N;B; MIS; List not Ack_ed!	NoAc; MIS;	NoAc; MIS;	NoAc; MIS;
<b>CON:</b> Contact	REQ; COR; can this occur?	----- -----	----- -----	----- -----	C;REQ; COR;
<b>FAT:</b> 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
<b>FAC:</b> 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
<b>WRC:</b> Wait_RC	C;RCm; FAI; Missing RC_Ack	C;RCm; FAI; Missing RC_Ack	NoAc; WRC;	IT; WRC;	C;RCm;REQ; COR;
<b>KON:</b> Konnect	NoAc; KON; may happen	C;RCm;DT;N; FAI; Misbehaviour!	----- -----	----- -----	C;RCm;DT;REQ;T1; COR; after Timeout: N
<b>REK:</b> 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
<b>SOS:</b> 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
<b>OPE:</b> Operation	NoAc; OPE; <b>typ Final event</b>	B; OPE; List not Ack_ed!	SYL1; OPE; <b>1: Alarm, go on</b>	C;DT;SYL; SOS; <b>2: Alarm, stop!</b>	NoAc; OPE; <b>Typ Final event</b>
<b>FAI:</b> Failure	NoAc; FAI; typical	----- -----	----- -----	----- -----	NoAc; FAI; don't trust!
<b>TT:</b> 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:	Stop TFO Frames if 3 Frames missing
<b>State:</b>	
<b>NAC:</b> Not_Active	----- -----
<b>WAK:</b> Wakeup	----- -----
<b>FIT:</b> First_Try	----- -----
<b>COR:</b> Continuous Retry	----- -----
<b>PER:</b> Periodic Retry	----- -----
<b>MON:</b> Monitor	----- -----
<b>MIS:</b> Mismatch	NoAc; MIS;
<b>CON:</b> Contact	----- -----
<b>FAT:</b> Fast Try	NoAc; FAT; typical in HO
<b>FAC:</b> Fast Contact	NoAc; FAC; typical in HO
<b>WRC:</b> Wait_RC	IT; WRC;
<b>KON:</b> Konnect	----- -----
<b>REK:</b> Re_Konnect	----- -----
<b>SOS:</b> Sync_Lost	NoAc; SOS; wait for Runout
<b>OPE:</b> Operation	C;DT;SYL; SOS; <b>2: Alarm, stop!</b>
<b>FAI:</b> Failure	----- -----
<b>TT:</b> 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:	Config request Matching parameters	Config request TFO Mismatch	Config acknowledgement Matching parameters	Config acknowledgement TFO Mismatch
State:				
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; 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:	Config req or Config ack Matching parameters	Config request TFO Mismatch	Config acknowledgement TFO Mismatch	Distant side has disabled TFO
State:				
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;

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:	Good signature, Immediate Codec Opt.	Good signature, Immediate Codec Opt.	New Config, Immediate Codec Opt.
<b>State:</b>			
<b>NAC:</b> Not_Active	----- -----	----- -----	----- -----
<b>WAK:</b> Wakeup	----- -----	----- -----	----- -----
<b>FIT:</b> First_Try	C;U;ACK;B; CON; <b>enter ICO</b>	C;U;ACK;B; CON; <b>enter ICO</b>	----- -----
<b>COR:</b> Continuous Retry	C;U;ACK;B; CON;	C;U;ACK;B; CON;	C;U;ACK;B; CON;
<b>PER:</b> Periodic Retry	C;U;ACK;B; CON;	C;U;ACK;B; CON;	C;U;ACK;B; CON;
<b>MON:</b> Monitor	C;U;ACK;B; CON;	C;U;ACK;B; CON;	C;U;ACK;B; CON;
<b>MIS:</b> Mismatch	C;U;ACK;B; CON;	C;U;ACK;B; CON;	C;U;ACK;B; CON;
<b>CON:</b> Contact	C;ACK; CON; <b>wait for HO</b>	NoAc; CON; <b>wait for HO or Runout</b>	C;ACK;B; CON;
<b>FAT:</b> Fast Try	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;
<b>FAC:</b> Fast Contact	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;
<b>WRC</b> Wait_RC	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;	C;ACK;RCm;B; CON;
<b>KON:</b> Konnect	C;ACK;RCm;B;DT; CON;	C;ACK;RCm;B;DT; CON;	C;ACK;RCm;B;DT; CON;
<b>REK:</b> Re_Konnect	C;ACK;RCm;B;DT;IT; CON;	C;ACK;RCm;B;DT;IT; CON;	C;ACK;RCm;B;DT;IT; CON;
<b>SOS:</b> Sync_Lost	C;ACK;RCm;B;IT; CON;	C;ACK;RCm;B;IT; CON;	C;ACK;RCm;B;IT; CON;
<b>OPE:</b> Operation	C;ACK;RCm;B;DT;IT; CON;	----- -----	C;ACK;RCm;B;DT;IT; CON;
<b>FAI:</b> Failure	NoAc; FAI;	NoAc; FAI;	NoAc; FAI;
<b>TT :</b> TFO_Term	----- -----	----- -----	----- -----

## 11 TFO Decision Algorithm

The TFO decision algorithm defines the processes invoked in both transcoders in order to examine the possibility for TFO establishment. Codec Types are in general only compatible to itself.

All members of the AMR-NB Codec Type family, except UMTS\_AMR, are compatible, when both codec types use compatible multi-mode ACSs. In any multi-mode configuration the UMTS\_AMR shall be regarded as only compatible to itself, not to any other AMR-NB Codec Type, to avoid incompatibilities in TFO-TrFO-TFO interworking scenarios. In single mode configuration, UMTS\_AMR and UMTS\_AMR\_2 are compatible, when both Codec Types use the same single rate ACS. The UMTS\_AMR\_2 is the preferred AMR-NB Codec Type for 3G systems.

All members of the AMR-WB Codec Type family are compatible, when both codec types use compatible multi-mode ACSs.

For the AMR Codec Type family the following tables 11-1 and 11-2 illustrate the compatible combinations (Table 11-1 for AMR-NB codec types, table 11-2 for AMR-WB codec types):

**Table 11-1: Compatibility of AMR-NB Codec Types**

distant → ↓ local	UMTS_AMR_2	UMTS_AMR	FR_AMR	HR_AMR	OHR_AMR
UMTS_AMR_2	compatible	compatible (Note)	compatible	compatible	compatible
UMTS_AMR	compatible (Note)	compatible	-	-	-
FR_AMR	compatible	-	compatible	compatible	compatible
HR_AMR	compatible	-	compatible	compatible	compatible
OHR_AMR	compatible	-	compatible	compatible	compatible

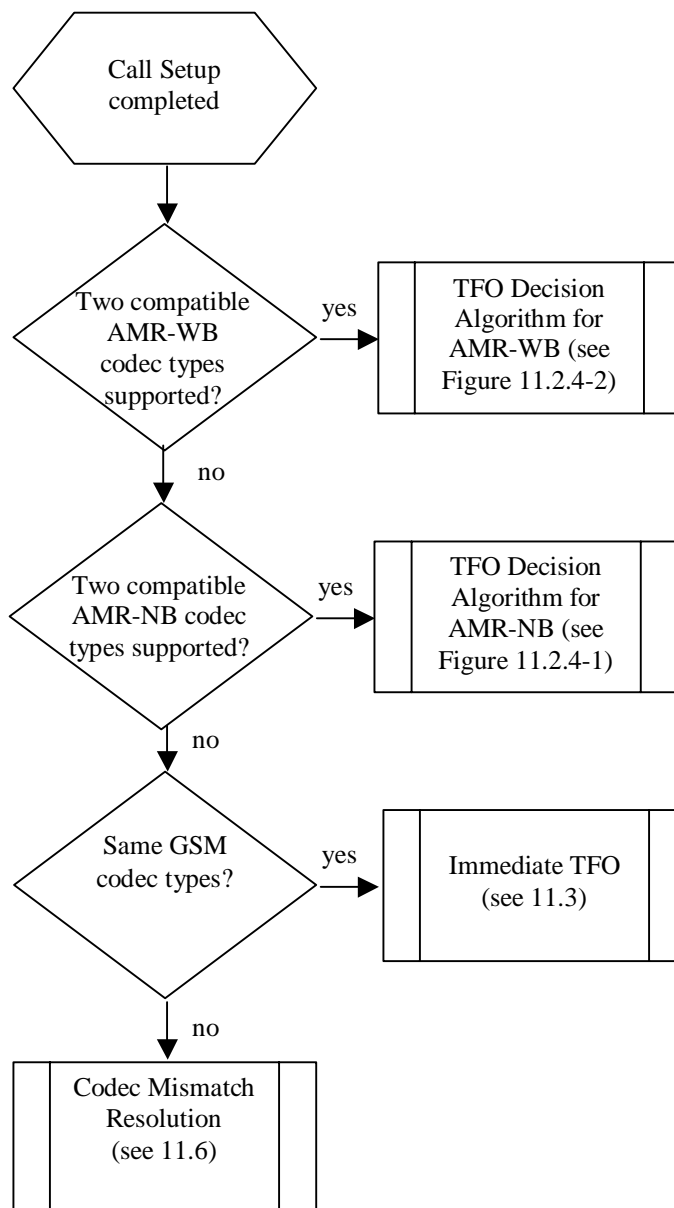
Note: only for single mode ACSs.

**Table 11-2 Compatibility of AMR-WB Codec Types**

distant → ↓ local	FR_AMR-WB	UMTS_AMR-WB	OFR_AMR-WB	OHR_AMR-WB
FR_AMR-WB	compatible	compatible	compatible	compatible
UMTS_AMR-WB	compatible	compatible	compatible	compatible
OFR_AMR-WB	compatible	compatible	compatible	compatible
OHR_AMR-WB	compatible	compatible	compatible	compatible

## 11.1 Main TFO Decision Procedure

The main TFO decision procedure is depicted in figure 11.1-1.



**Figure 11.1-1: Main TFO Decision Algorithm**

## 11.2 TFO Decision Algorithm for AMR codec types

The TFO Decision Algorithm for AMR codec types defines the processes that are invoked in order to examine the possibility for a TFO establishment if both radio legs use compatible AMR codec types.

### 11.2.1 Principles

In order to yield high speech quality the following items are underlying principles of the TFO decision algorithm for AMR codec types:

- Avoid immediate TFO establishment with a following codec optimisation that has to interrupt the TFO connection.

- Go into immediate TFO if this is possible with a good configuration, otherwise do codec optimisation.
- Only do codec mode optimisation if the ongoing TFO connection is established on a contiguous subset of the ACS and if this ongoing TFO connection need not be interrupted.

## 11.2.2 Available Information at Call Set-up

After the exchange of TFO\_REQ and TFO\_ACK messages the following information is available at the transcoders on both sides:

- Local / distant codec type (FR\_AMR, HR\_AMR, UMTS\_AMR, UMTS\_AMR\_2, OHR\_AMR, FR\_AMR-WB, UMTS\_AMR-WB, OHR\_AMR-WB, OFR\_AMR-WB)
- Local / distant supported codec set (LSCS / DSCS)
- Local / distant ACS (LACS / DACS)
- Local / distant MACS
- Local / distant ACS optimisation mode (OM)
- Local / distant version number (Ver)

With this information the following can be calculated:

- Common ACS (CACCS)
- Common supported codec set (CSCS)
- Common MACS (CMACS)
- Optimised ACS (OACS)

Furthermore, additional information on supported codecs may become available when the Codec List is received. There are several possibilities for receiving this information: 1) by TFO\_REQ and TFO\_ACK messages including optional Extension\_Blocks, 2) by TFO\_REQ\_L and TFO\_ACK\_L messages, and 3) by Configuration Frames (Con\_Req, Con\_Ack). In any case, the following information is available for each supported codec in the Codec List:

- Local / distant supported codec type
- Local / distant supported codec set (LSCS / DSCS)
- Local / distant MACS

In the case that Generic Configuration Frames are used to transmit the Codec List the following information is also available for each codec in the Codec List:

- Local / distant intended ACS (LACS/ DACS)
- Local / distant ACS optimisation mode (OM)

In all other cases (no Generic Configuration Frames are used) the OM bit shall be assumed to be set ("ACS optimisation allowed").

With this information the following can be calculated for each compatible codec combination in the local and distant Codec List:

- Common supported codec set (CSCS)
- Common MACS (CMACS)
- Optimised ACS (OACS)

### 11.2.3 Void

### 11.2.4 Flowchart for AMR-NB TFO Decision

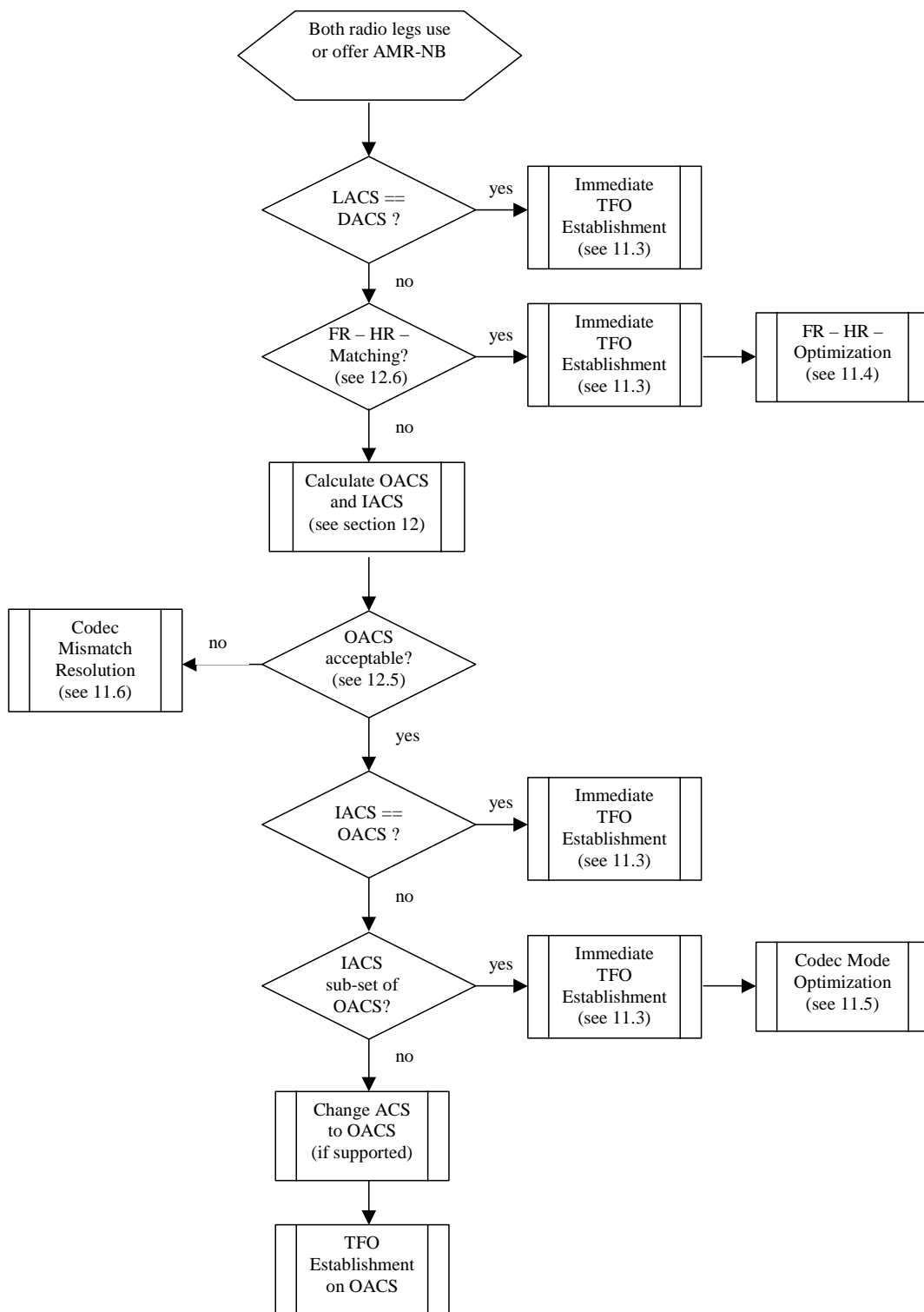


Figure 11.2.4-1: Flowchart for AMR-NB TFO Establishment at Call Set-Up

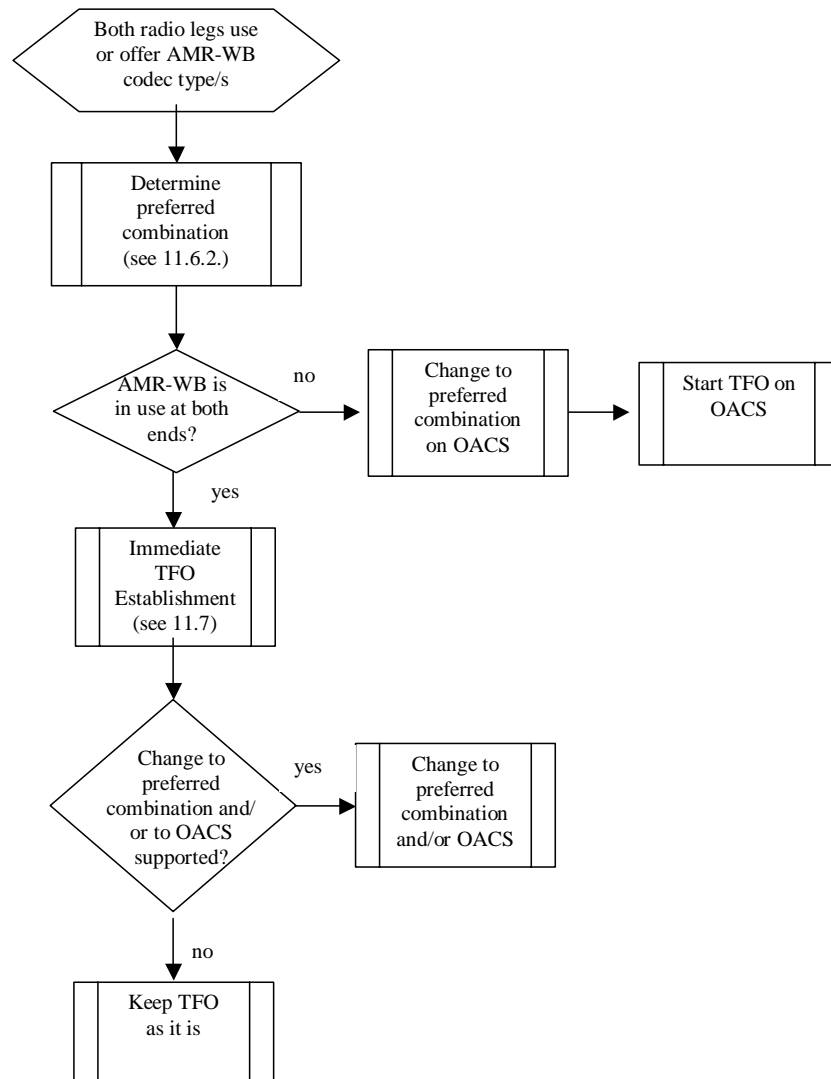


Figure 11.2.4-2: Flowchart for AMR-WB TFO Establishment at Call Set-Up

### 11.2.5 Annotations to the Flowcharts

- LACS == DACS:  
Establish immediate TFO if the local and distant ACS are identical.  
Example: Enable immediate TFO establishment within one operator's homogenous network. The operator's choice is always acceptable and needs no optimisation.
- FR – HR Matching  
The rules for FR – HR – Matching are stated in clause 12.6.  
Goal: Enable immediate TFO between 3G channels and 2G FR and 2G HR channels.
- FR – HR – Optimisation  
The rules for FR – HR – Optimisation are stated in clause 11.4.
- Calculate OACS and IACS:  
The calculation of the OACS is described in clause 12.  
The Immediate ACS (IACS) is given by the common ACS (CACS) if it is contiguous.
- OACS acceptable:  
The acceptability rules for the OACS are stated in clause 12.5.

- IACS == OACS  
If the immediate ACS is already optimal, establish immediate TFO.
- IACS subset of OACS:  
Immediate TFO is established on a contiguous subset of the OACS. Afterwards, a codec mode optimisation is performed without interrupting the TFO connection.
- Change ACS to OACS  
If immediate TFO cannot be established, both sides must change their ACS to the OACS in order to enable TFO. If one side doesn't support an ACS change (ACS Optimisation Mode), the OACS determination rules ensure that the OACS is a contiguous subset of the fix ACS. So a TFO connection can be established without the need for an ACS change on that side.
- Codec Mismatch Resolution  
A TFO connection with the currently used AMR codec types will not be possible, but the remaining codec types have to be investigated.

## 11.3 Immediate TFO Establishment

Immediate TFO establishment shall take place if

- both radio legs use the same codec type that is different from an AMR codec type; or
- the local ACS is equal to the distant ACS in the case of two compatible AMR codec types; or
- the CACS is equal to the OACS and the CACS fulfils the contiguity rule in the case of two compatible AMR codec types; or
- the rules for FR – HR – matching are fulfilled in the case of two compatible AMR-NB codec types; or
- the CACS is a contiguous subset of the OACS in the case of two compatible AMR codec types and Codec Mode Optimisation is supported and will be done after immediate TFO establishment.

If both radio legs use the same codec type that is different from an AMR codec type, immediate TFO shall be established on this common codec type. If both radio legs use compatible AMR codec types and immediate TFO can be established, each side keeps its own AMR codec type (e. g. FR\_AMR, HR\_AMR, UMTS\_AMR, UMTS\_AMR\_2) and Active Codec Set (ACS).

If immediate TFO is possible on a currently used codec type, but also on a supported codec type with a higher preference level (see 11.6.2), then TFO shall not be started on the used codec type, but only after switching to the preferred codec. For details on this Immediate Codec Type Optimisation see 11.7.

## 11.4 FR – HR – Optimisation (only for AMR-NB)

FR – HR – Optimisation takes place after immediate TFO establishment in the case of FR – HR – Matching. The FR\_AMR-side adopts the ACS of the HR\_AMR-side, if this ACS is supported and the optimisation mode allows an ACS change.

This ACS change can be done without interrupting the TFO connection that is established on a contiguous subset.

## 11.5 Codec Mode Optimisation

After an immediate TFO establishment with compatible AMR codec types, a codec mode optimisation shall be invoked if the optimisation can be done without interrupting the TFO connection, i.e. without degradation of speech quality. Codec Mode Optimisation takes place in the following situations:

- After immediate TFO establishment on a IACS that is a contiguous subset of the OACS.



## 11.6 Codec Type Optimisation and Codec Mismatch Resolution

The objective of the Codec Mismatch Resolution and the Codec Type Optimisation is to find the optimised TFO codec type and configuration for a TFO connection. Codec Mismatch Resolution is invoked if a TFO establishment is not possible on the currently used codec types. Codec Type Optimisation may happen while a TFO connection is ongoing and the capabilities of one radio leg have changed (e. g. after a hand-over, other reasons).

Codec Mismatch Resolution and Codec Type Optimisation are optional features. If one radio leg doesn't support these features, the codec list sent in the TFO\_REQ\_L and TFO\_ACK\_L messages (or Con\_Req and Con\_Ack frames) shall be restricted to the local used codec. If supported, the Codec Type Mismatch Resolution or the Codec Type Optimisation shall be performed every time a new codec list is sent or received by TFO\_REQ\_L or TFO\_ACK\_L (or Con\_Req and Con\_Ack frames) messages.

The determination of the local codec list (list of all codec types supported by the local radio leg, consisting of the local UE and the local RAN) is outside the scope of the present document. Similarly, the determination of the attributes of all locally supported codec types (e.g. LSCS for AMR codec types) is also outside the scope of the present document. Only codec types that are real alternatives, considering all resources (UE, RAN, TC, radio interface, cell capacity, interference), shall be reported within the local codec list. Only codec type Attributes that can be considered shall be indicated with the codec list as well. This means that if a TFO configuration is not desirable, it should not be listed in the TFO\_REQ\_L or TFO\_ACK\_L messages (or Con\_Req and Con\_Ack frames).

### 11.6.1 Procedure

1. The transcoders shall exchange their lists of supported codec types (codec list) and their associated attributes. This is done either by the exchange of TFO\_REQ\_L and/or TFO\_ACK\_L messages or Con\_Req and Con\_Ack frames.
2. Each side shall identify all candidate TFO configurations involving compatible codec types supported by both radio legs.
3. Each side shall calculate the OACS in the case of an AMR TFO candidate. If the OACS is not acceptable, this candidate shall be removed from the list of candidate TFO configurations.
4. The candidate TFO configuration with the highest preference level shall define the optimised codec type and the optimised codec configuration.
5. Each side shall switch its operation to the optimised codec type and the optimised codec configuration. If no acceptable TFO candidate was found, TFO is not possible.

### 11.6.2 Preference List of TFO candidates

The preference list of TFO candidates orders all possible TFO combinations according to the speech quality they provide.

**Table 11.6.2-1: Codec Type Combination Preference List, Part 1**

distant → ↓ local	OFR_AMR-WB	UMTS_AMR-WB	FR_AMR-WB	OHR_AMR-WB
OFR_AMR-WB	1	2	4	7
UMTS_AMR-WB	symmetric	3	5	8
FR_AMR-WB	symmetric	symmetric	6	9
OHR_AMR-WB	symmetric	symmetric	symmetric	10

For AMR-WB the preference is determined by the OACS: A combination with the highest mode in the OACS has preference. If the highest mode in OACSs for at least two combinations is identical, then the preference level as given in Table 11.6.2-1 shall decide.

Examples:

The configuration (OFR\_AMR-WB, UMTS\_AMR-WB, OACS={6,60, 8,85, 12,65, 23,85}) is preferred to (OFR\_AMR-WB, OFR\_AMR-WB, OACS={6,60, 8,85, 12,65, 15,85}).

The configuration (OFR\_AMR-WB, OFR\_AMR-WB, OACS={6,60, 8,85, 12,65}) is preferred to (OFR\_AMR-WB, UMTS\_AMR-WB, OACS={6,60, 8,85, 12,65}).

**Table 11.6.2-2 Codec Type Combination Preference List, Part 2**

distant → ↓ local	UMTS_AMR_2	FR_AMR	UMTS_AMR	OHR_AMR	HR_AMR
UMTS_AMR_2	11	12	15 (Note)	17	20
FR_AMR	symmetric	13	Not compatible	18	21
UMTS_AMR	symmetric	Not compatible	14	Not compatible	Not compatible
OHR_AMR	symmetric	symmetric	Not compatible	19	22
HR_AMR	symmetric	symmetric	Not compatible	symmetric	23

Note: only for single mode ACSs

**Table 11.6.2-3 Codec Type Combination Preference List, Part 3**

distant → ↓ local	GSM_EFR	GSM_FR	GSM_HR
GSM_EFR	16	Not compatible	Not compatible
GSM_FR	Not compatible	24	Not compatible
GSM_HR	Not compatible	Not compatible	25

All other possible codec type combinations not listed in these table 11.6.2.3-1/2/3 are not compatible.

The codec type FR\_AMR-WB is preferred to the AMR-NB codec types, because it still provides significantly better speech quality.

The two equivalent combinations FR\_AMR-WB ⇔ UMTS\_AMR-WB and UMTS\_AMR-WB ⇔ FR\_AMR-WB should not exist in parallel, because these two AMR-WB codec types are not offered by one side simultaneously.

The speech quality of some AMR-WB codec type combinations involving FR\_AMR-WB, UMTS\_AMR-WB and OHR\_AMR-WB are very similar. Therefore within category 1 the OACSs of the possible combinations are evaluated. For details on this evaluation see clause 12.3.2.2 .

The codec type UMTS\_AMR\_2 is the most preferred AMR-NB codec type, because it is compatible with all other AMR codec types.

The codec type FR\_AMR is preferred to UMTS\_AMR because UMTS\_AMR is not compatible with FR\_AMR and HR\_AMR.

If the two equivalent AMR-NB combinations like FR\_AMR  $\Leftrightarrow$  HR\_AMR and HR\_AMR  $\Leftrightarrow$  FR\_AMR or UMTS\_AMR\_2  $\Leftrightarrow$  HR\_AMR and HR\_AMR  $\Leftrightarrow$  UMTS\_AMR\_2 etc. exist in parallel, then they shall be ranked according to the following rules:

1. The combination with the highest number of modes shall be selected.
2. If they have the same number of modes, then the combination with the widest spread shall be selected. The spread is the difference between the highest and the lowest mode indexes.
3. If the spreads are identical, then the combination with the highest mode in the OACS shall be selected.
4. If the highest modes are identical, repeat 3 with the second highest mode. If the second highest are identical, then repeat 3 with the third highest, etc.

## 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 Optimisation 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.

## 11.8 TFO Decision Table for AMR-WB

For AMR-WB only a limited set of configurations is allowed. Table 12.8-1.gives the effective ACS for all combinations of these allowed configurations.

**Table 11.8-1: Effective ACS for AMR-WB TFO**

Local scenario	<b>0:</b> ACS:=A=SCS	<b>1:</b> ACS:=A, SCS:=D 23,85 15,85	<b>2:</b> ACS:=B=SCS 15,85	<b>3:</b> ACS:=A, SCS=D 15,85	<b>4:</b> ACS:=C=SCS 23,85	<b>5:</b> ACS=C, SCS=D 23,85
	12,65	12,65 12,65	12,65	12,65	12,65	12,65
	8,85	8,85 8,85	8,85	8,85	8,85	8,85
	6,60	6,60 6,60	6,60	6,60	6,60	6,60
Distant	OM=F	OM=A	OM=F	OM=A	OM=F	OM=A

scenario	MACS=3	MACS=4	MACS=4	MACS=4	MACS=4	MACS=4
<b>0</b>	A	A	A	A	A	A
<b>1</b>	Symmetrical	A	B (config. 1 changes to 3)	B (config. 1 changes to 3)	C (config. 1 changes to 5)	C (config. 1 changes to 5)
<b>2</b>	Symmetrical	Symmetrical	B	B	A	B (config. 5 changes to 3)
<b>3</b>	Symmetrical	Symmetrical	Symmetrical	B	C (config. 3 changes to 5)	C (config. 3 changes to 5)
<b>4</b>	Symmetrical	Symmetrical	Symmetrical	Symmetrical	C	C
<b>5</b>	Symmetrical	Symmetrical	Symmetrical	Symmetrical	Symmetrical	C

Immediate TFO (see 11.3) is always possible for each combination of the allowed configurations. In some cases a Codec Mode Optimisation (see 11.5) is invoked after TFO establishment. For these cases the changing configuration is also specified in the table 12.8-1. Remark: For all combinations where one side changes the configuration, immediate TFO is established on ACS A.

All final (and immediate) ACSs as listed in table 12.8-1 are contiguous and acceptable, by design. No check – as for an AMR-NB OACS (see clause 12.4, 12.5 and 12.7) – is needed.

---

## 12 Determination of the OACS for AMR-NB

In case of inconsistencies between the TFO decision C-Code in Annex E and this clause the C-Code shall take precedence.

### 12.1 Principles

The determination of the OACS shall be done considering the available information (see 11.2.2).

The common MACS is defined as the minimum value of the local and distant MACS.

The determination of the OACS shall depend on the local and distant optimisation mode (LOM / DOM).

### 12.2 Algorithm for OACS Determination

#### 12.2.1 Case 1: No side supports ACS change

If neither the local side nor the distant side supports an ACS change, the OACS is equal to the CACS if it fulfils the contiguity rule. Otherwise, the rules for contiguous subset selection are applied to the CACS in order to obtain the OACS.

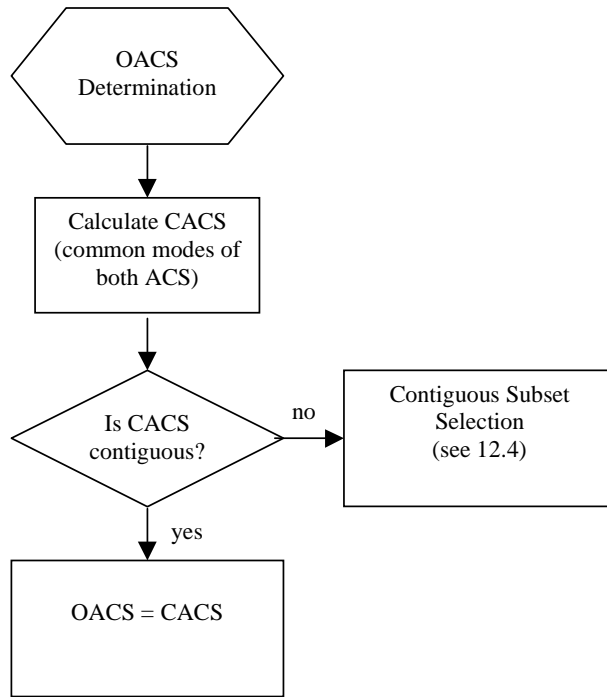
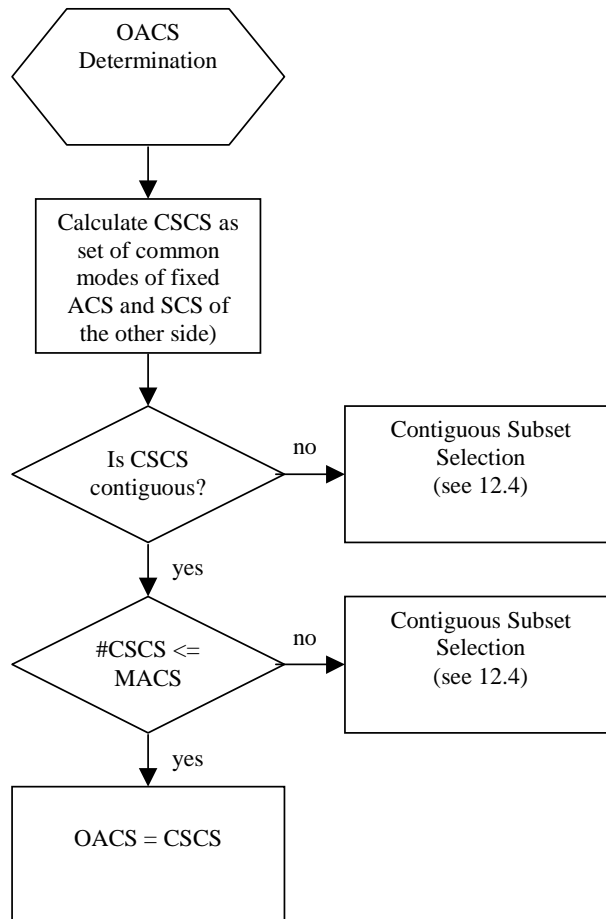


Figure 12.2.1-1: OACS Determination when No side supports ACS Change

## 12.2.2 Case 2: Only one side supports ACS change

If only one side supports an ACS change, the CSCS is built with the common modes of the SCS of the flexible side and the unchangeable ACS.

If the CSCS doesn't fulfil the contiguity rule or the common MACS is lower than the number of modes in the CSCS, the OACS is obtained by applying the rules for contiguous subset selection. Otherwise, the OACS is equal to the CSCS.



**Figure 12.2.2-1: OACS Determination when only one side supports ACS Change**

## 12.2.3 Case 3: Both sides support ACS change

If both sides support ACS change, the CSCS is built with the common modes of both SCS.

The Optimised Active Codec Set (OACS) is equal to the Common Supported Codec Set (CSCS) if the number of modes in the CSCS is equal or lower than the common MACS.

If the number of modes in the CSCS is higher than the common MACS, the OACS shall be defined as a subset of the CSCS using the OACS selection rules.

If the CSCS is not empty, then a Optimised Active Codec Set (OACS) exists.

The existence of an OACS doesn't mean the OACS is acceptable. To check this, the acceptability rules for the OACS have to be applied.

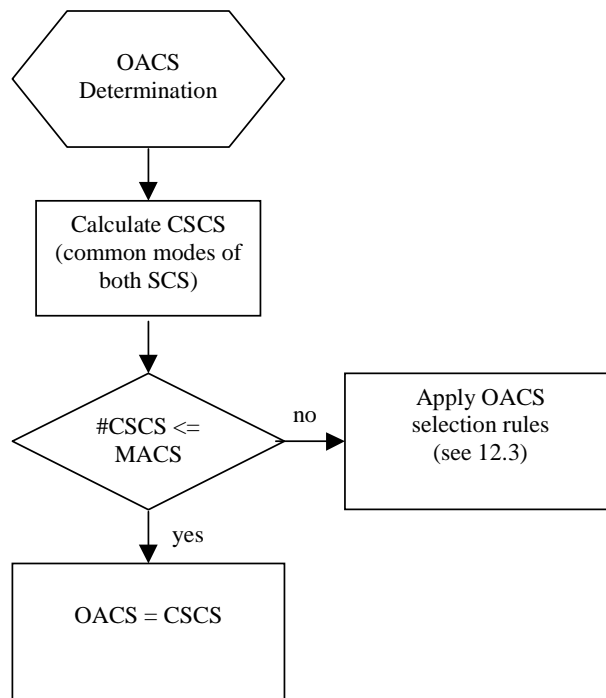


Figure 12.2.3-1: OACS Determination when both sides support ACS Change

## 12.3 OACS Selection Rules

If both radio legs support ACS change and if the number of modes contained in the CSCS is greater than the common MACS, the OACS is determined by the following rules. These rules are skipped as soon as an OACS containing CMACS modes is found.

The reference C-Code also implements the OACS rules (see Annex E). In case of inconsistencies between this clause and the C-code, the C-code takes precedence.

### 12.3.1 Case 1: No Half Rate Channel is involved

#### Case MACS == 1

1. Select mode according to preference list {6,7, 7,4, 5,9, 5,15, 4,75, 7,95, 10,2, 12,2}.

#### Case MACS == 2

1. If mode 10,2 is supported, do not include mode 12,2.
2. Select highest mode.
3. If mode 12,2 or mode 10,2 is selected, select mode according to preference list {6,7, 7,4, 5,9, 5,15, 4,75, 7,95, 10,2, 12,2}.
4. Select lowest mode.

#### Case MACS > 2

1. If mode 10,2 is supported, do not include mode 12,2.
2. If mode 4,75 is supported, do not include mode 5,15.
3. If mode 5,15 is supported, do not include mode 5,9.
4. If mode 5,9 is supported and mode 4,75 is not supported, do not include mode 6,7.
5. If mode (12,2 or 10,2) and 7,4 is supported, do not include mode 7,95.

6. If mode 7,95 is supported, do not include 7,4.
7. Select lowest mode.
8. Select highest mode.
9. Select mode 6,7.
10. Select mode 5,9.

## 12.3.2 Case 2: A Half Rate Channel is involved

### Case MACS == 1

1. Select mode according to preference list {5,9, 5,15, 4,75, 6,7, 7,4, 7,95}.

### Case MACS == 2

1. Select highest mode.
2. Select lowest mode.

### Case MACS > 2

The same rules apply as in clause 12.3.1 for the case MACS>2.

## 12.4 Rules for Contiguous Subset Selection

The rules for contiguous subset selection are necessary if one or both radio legs don't support ACS change. If TFO should be established in these cases, the resulting OACS must fulfil the contiguity rule considering the fixed ACS.

If the CSCS doesn't fulfil the contiguity rule, a contiguous subset with a maximum number of modes shall be selected as the new CSCS. This subset must contain the lowest mode of the fixed ACS, otherwise there is no OACS.

If the common MACS is lower than the number of modes in the CSCS, the highest modes shall be removed from the CSCS until the number of modes in the CSCS is equal to the common MACS. This new codec set defines the OACS.

## 12.5 Acceptability Rule for the OACS

An optimised ACS (OACS) is acceptable for AMR-NB TFO if

1. the Highest-Mode-Rule is fulfilled and
2. the Lowest-Mode-Rule is fulfilled.

High Mode Rule (don't give up tandem with high quality modes)

The highest mode in the OACS is not lower than one mode below the minimum of the highest modes of both ACS.

Low Mode Rule (tandem AMR with robust low modes performs better)

Either the lowest mode of the OACS is not higher than a specific maximum mode or both ACS don't contain lower modes than the lowest mode in the OACS. The specific maximum mode is 5,9 for TFO connections involving a half rate channel and 7,4 otherwise.

## 12.6 FR – HR – Matching

A common ACS (CACCS) is acceptable for immediate TFO establishment without consideration of the OACS if all of the following conditions are fulfilled:

- the one radio leg uses FR\_AMR or UMTS\_AMR\_2 or HR\_AMR, the other uses HR\_AMR [for AMR-NB TFO];
- the CACS is contiguous;
- the CACS fulfils the acceptability rule.



## 12.7 Contiguity Rule

The Contiguity Rule states that the codec modes of the CACS must be contiguous modes in the local ACS (LACS) and the distant ACS (DACs). Additionally, the CACS must contain the lowest mode of both ACS. The Contiguity Rule is used to enable TFO establishment on a CACS different from the ACS. In a GSM system this is necessary because link adaptation is only possible using maximum rate control with adjacent modes of the ACS.

Example A: LACS: 12,2 10,2 7,95 5,9  
 DACS: 10,2 7,95 5,9  
 CACS: 10,2 7,95 5,9      Contiguity Rule is fulfilled

Example B: LACS: 12,2 10,2      4,75  
 DACS: 10,2 7,4      4,75  
 CACS: 10,2      4,75      Contiguity Rule is not fulfilled for the DACS

## 12.8 Examples of OACS Computation

### 12.8.1 TFO between a full rate channel and a half rate channel

	SCS	ACS	CACS	OACS	CSCS	ACS	SCS
12,2	x						
10,2	x	x					
7,95	x						
7,4	x			x	x	x	x
6,7	x	x	x	x	x	x	x
5,9	x	x	x	x	x	x	x
5,15	x				x		x
4,75	x	x	x	x	x	x	x

This is an example for FR – HR – Matching.

Immediate TFO is possible using the CACS.

Afterwards, a codec mode optimisation is performed without interrupting the ongoing TFO connection.

### 12.8.2 TFO between two full rate channels with different ACS

	SCS	ACS	CACS	OACS	CSCS	ACS	SCS
12,2	x				x	x	x
10,2	x	x		x	x		x
7,95	x				x		x
7,4	x				x		x
6,7	x	x	x	x	x	x	x
5,9	x	x	x	x	x	x	x
5,15	x				x		x
4,75	x	x	x	x	x	x	x

The CACS is a contiguous subset if the OACS.

Immediate TFO and subsequent codec mode optimisation without interrupting TFO is performed.

### 12.8.3 Full Rate Channel with restricted capabilities

	SCS	ACS	CACS	OACS	CSCS	ACS	SCS
12,2	x						
10,2	x	x					
7,95	x						
7,4	x			x	x	x	x
6,7	x	x	x	x	x	x	x
5,9	x	x					
5,15	x						
4,75	x	x	x	x	x	x	x

Immediate TFO is not possible because the CACS is not contiguous.

TFO on the OACS is acceptable since a tandem connection would not provide a better speech quality.

The OACS is acceptable since both the High Mode Rule and the Low Mode Rule are fulfilled.

### 12.8.4 Scenario: Full Rate Channel with MACS == 2

	SCS	ACS	CACS	OACS	CSCS	ACS	SCS
12,2							
10,2							
7,95							
7,4	x	x		x	x		x
6,7						x	x
5,9	x			x	x		x
5,15	x	x					
4,75						x	x

The OACS is acceptable for a TFO connection. A tandem connection would not provide better speech quality. Both High Mode Rule and Low Mode Rule are fulfilled. For good radio channels a tandem between 7,4 and 6,7 is worse than a 7,4 TFO connection. For poor radio channels a 5,9 TFO connection is considered to be robust enough.

### 12.8.5 Scenario: AMR codec type with only one supported mode

	SCS	ACS	CACS	OACS	CSCS	ACS	SCS
12,2	x			x	x	x	x
10,2	x	x					
7,95	x						
7,4	x						
6,7	x	x					
5,9	x	x					
5,15	x						
4,75	x	x					

One side offers an FR\_AMR codec type with only the 12,2 mode in the supported codec set.

The OACS is not acceptable, TFO should not be established. A tandem connection would provide better overall speech quality. If the only supported mode is lower or equal to the 7,4 mode, TFO shall be established on this single mode. The 7,4 mode is considered to be robust enough in the case of poor radio channels. On the other hand, a tandem connection between 7,4 and 12,2 would be worse than a 7,4 TFO connection for good radio channels.

# Annex A (normative): In-band Signalling Protocol: Generic Structure

## A.0 Scope of Annex A and Annex B

Inband Signalling Messages (IS Messages) can be used to construct a specific IS Protocol for the communication between telecommunication entities for various purposes. The original purpose was to establish Tandem Free Operation of Mobile-to-Mobile calls in GSM networks. The IS Messages provide communication channels inside the speech signal paths between the speech transcoders.

In addition IS Messages allow the control of equipment within the speech signal paths between these telecommunication entities (e.g. speech transcoders). These equipments are termed "In Path Equipments" (IPEs).

Annex A defines the generic structure of these IS Messages and rules for the IS\_Sender.

Annex B defines the generic rules with respect to these IS Messages for the IPEs.

Annex A is mandatory for TFO-capable Transcoder Equipment and informative for IPEs.

Annex B is informative for TFO-capable Transcoder Equipment.

Annex B shall be followed by IPEs, which want to be compatible to IS Messages.

## A.1 Generic Structure of Inband Signalling Messages

All IS Messages follow a set of design rules, or a generic structure, which allow to identify and bypass them by IPEs without detailed knowledge of the IS Protocol served. The principle of the IS Protocol shall in that sense be future proof: it can be enhanced and extended to other applications without modifying the IPEs.

The IS Messages replace some of the LSBs of the PCM samples of the Speech, Audio or Modem signal.

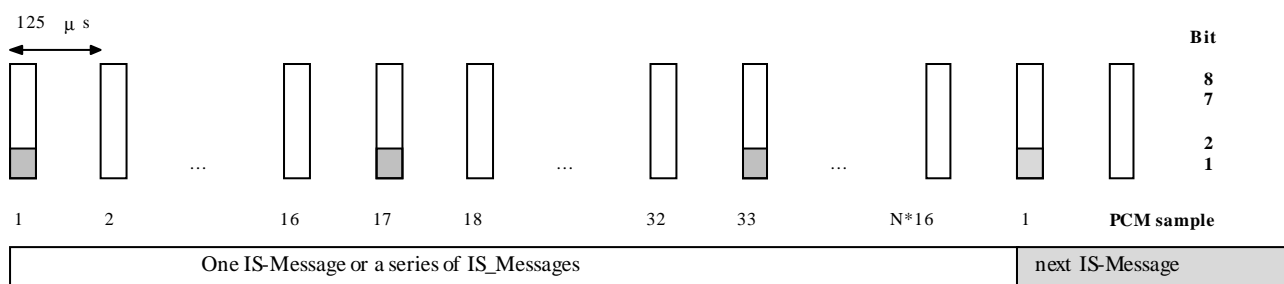
By construction the introduced signal distortion is practically inaudible in case of Speech signals.

Modem signals will in most cases not be affected with respect to their data transmission performance.

### A.1.1 Frequency and Order of Bit Transmission

IS Messages are transferred within the Least Significant Bit (LSB) of PCM samples on 64 kbit/s links, by replacing the LSB of every 16<sup>th</sup> consecutive PCM sample with one bit of the IS Message (16\_PCM\_Sample\_Grid).

This is equivalent to an average bit rate of 10 bit per 20 ms or 500 bits per second. See Figure A1.1-1:



**Figure A.1.1-1: Inband Signalling Structure**

A vertical bar denotes an 8-bit PCM sample, the shadowed box in bit 1 (LSB) represents an inserted bit of the IS-Message.

By definition each IS Message "occupies" an integer multiple of 16 PCM samples. Especially the 15 PCM samples after the last inserted bit of an IS Message "belong" still to that IS Message.

All IS Messages, whichever type, have by construction "0"-Bits at every 10<sup>th</sup> position, starting with position 1, 11, 21 and so on. This "0"-Bits occur therefor regularly every 20 ms and may be used for synchronization purposes.

Each IS Message consists of an IS\_Header followed by an IS\_Command\_Block. Most IS Messages have a number of further IS\_Extension\_Blocks. Figure A1.1-2 shows an example with two IS\_Extension\_Blocks.

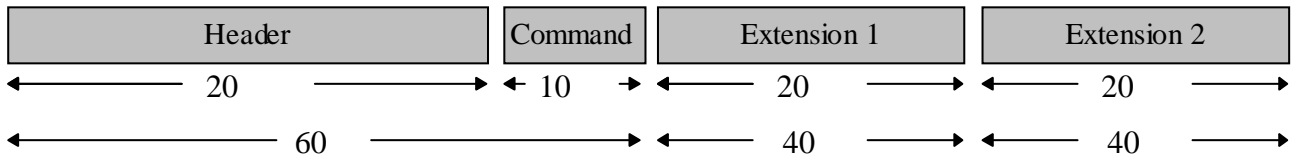


Figure A.1.1-2: Example for IS Message with two IS\_Extension\_Blocks

The MSB of each constituent field is transmitted first. The IS\_Header is transmitted first, followed by the IS\_Command\_Block and - if applicable - any further IS\_Extension\_Block(s).

By construction all IS Messages do have lengths of integer multiples of 10 bits, thus occupying integer multiples of 160 PCM samples, thus lasting integer multiples of 20 ms. The shortest IS Message has a length of 60 ms.

### A.1.2 IS\_Header

The IS\_Header consists of a 20-Bit long sequence, as defined in Figure A1.2-1:

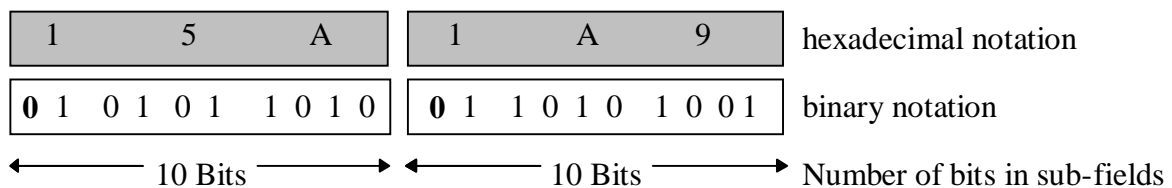


Figure A.1.2-1: Structure of the 20 bit IS\_Header

### A.1.3 IS\_Command\_Block

The IS\_Command identifies the IS Message and/or serves for the control of IPEs. The names of the IS\_Commands and their codes in hexadecimal notation in the IS\_Command\_Block are given in the Table A.1.3-1

Table A.1.3-1: Defined IS\_Commands

Index	Command	Code	Meaning / Action
		hexadecimal Nibble 1-3	
0	Reserved	0x000	no extension
1	<b>REQ</b>	0x05D	Denotes an IS_REQ Message, with extension
2	<b>ACK</b>	0x0BA	Denotes an IS_ACK Message, with extension
3	<b>IPE</b>	0x0E7	Denotes an IS_IPE Message, with extension, i.e. an IS_TRANS or the IS_NORMAL Message
4	<b>FILL</b>	0x129	Denotes the IS_FILL Message, no extension
5	<b>DUP</b>	0x174	Denotes the IS_DUP Message, no extension
6	<b>SYL</b>	0x193	Denotes the IS_SYL Message, no extension
7	reserved	0x1CE	no extension

All other values are reserved for future use.

Each IS\_Command is protected by the binary, systematic (9,3) block code with generator polynomial  $g(x) = x^6 + x^4 + x^3 + x^2 + 1$ . The minimum Hamming distance of this code is  $d_{min} = 4$ , which allows the correction of up to one bit error within each code word of length 9 bits.

The first bit (MSB) of the IS\_Command\_Block is defined to be "0", for synchronisation purposes, see Figure A1.3-1.

Table A-1 gives the hexadecimal notation of the complete IS\_Command\_Block.

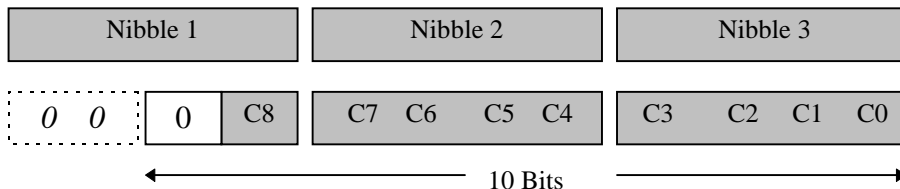


Figure A.1.3-1: General Construction of an IS\_Command\_Block

### A.1.4 IS\_Extension\_Block(s)

Most IS Messages have one or more IS\_Extension\_Block(s). Each IS\_Extension\_Block is 20 bits long and shall consist of two "0"-Synchronization\_Bits at position 1 (MSB) and 11, a 16-bit Information\_Field (split into two fields of 9 and 7 bits, respectively) and a 2-bit Extension\_Field (EX), see Figure A1.4-1:

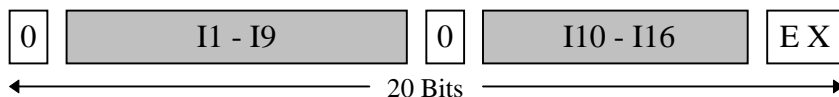


Figure A1.4-1: General Construction of an IS\_Extension\_Block

The Extension\_Field indicates if an other IS\_Extension\_Block is following (EX := "1.1" ) or not (EX := "0.0").

All other codes are reserved. This may be used to detect transmission errors within the Extension\_Field.

## A.2 Detailed Specification of IS Messages

### A.2.1 IS\_REQ Message

With the IS\_REQ Message an IS\_Sender can test, if there is an IS Partner and indicates that it is willing to negotiate.

IS\_REQ is used to initiate the IS Protocol or to indicate changes in the configuration, etc.

IS\_REQ has at least one IS\_Extension\_Block, containing the IS\_System\_Identification. (see clause A.5).

Other IS\_Extension\_Blocks may follow, see Figure A2.1-1.

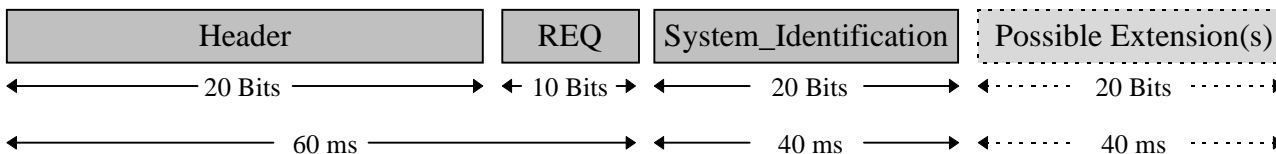


Figure A.2.1-1: General Construction of an IS\_REQ Message

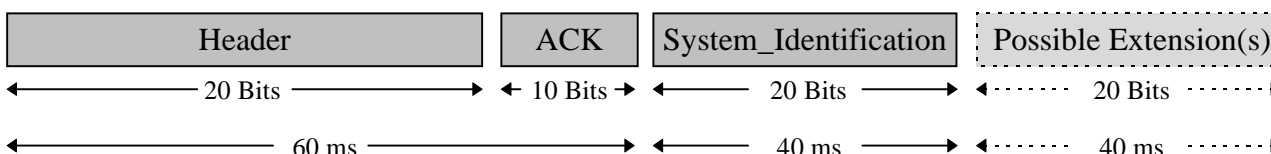
In general an IS\_REQ Message shall be as short as possible. Special care must be taken in the design of the IS\_Extension\_Blocks to avoid audible effects, since sometimes an IS\_REQ Message may be transmitted for quite some time (several seconds).

## A.2.2 IS\_ACK Message

With the IS\_ACK Message an IS Partner typically answers an IS\_REQ Message or an IS\_ACK Message. It can also be used to submit further information to the other IS Partner. IS\_REQ and IS\_ACK are the main message types between IS Partners.

The IS\_ACK has at least an IS\_Extension\_Block containing the IS\_System\_Identification (see clause A.5).

Other IS\_Extension\_Blocks may follow, see Figure A.2.2-1.



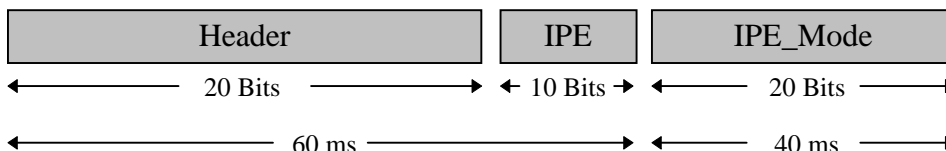
**Figure A.2.2-1: General Construction of an IS\_ACK Message**

No specific design constraints with respect to audibility exist, since IS\_ACK is typically not sent very often.

## A.2.3 IS\_IPE, IS\_TRANS and IS\_NORMAL Messages

The IPE command denotes IS\_IPE Messages. An IPE shall always look for this type of message and follow the instruction. An IS\_Sender shall use this IS\_IPE Message to command all IPEs into a specific mode of "Bit Transparency".

This Message has one IS\_Extension\_Block, indicating the requested IPE\_Mode. See Figure A.2.3-1.



**Figure A.2.3-1: General Construction of an IS\_IPE Message**

No specific design constraints with respect to audibility exist, since IS\_IPE is typically not sent very often.

Table A-2 defines 16 out of 32 possible IPE\_Commands. The other codes are reserved for future extensions.

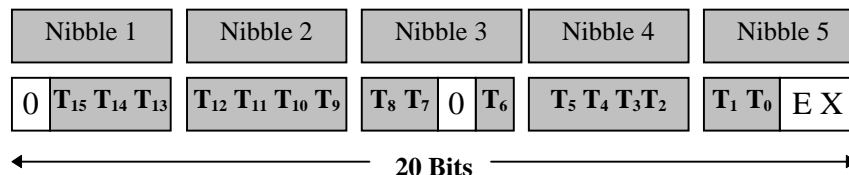
**Table A.2.3-1: Defined IPE\_Modes**

Index	IPE_Mode	Code	MEANING / ACTION
		hexadecimal Nibble 1 - 5	
0	Normal	0x00000	Normal Operation
1	Trans_1_u	0x044DC	pass 1 LSB; 7 upper Bits are used
2	Trans_2_u	0x089B8	pass 2 LSBs; 6 upper Bits are used
3	Trans_3_u	0x0CD64	pass 3 LSBs; 5 upper Bits are used
4	Trans_4_u	0x11570	pass 4 LSBs; 4 upper Bits are used
5	Trans_5_u	0x151AC	pass 5 LSBs; 3 upper Bits are used
6	Trans_6_u	0x19CC8	pass 6 LSBs; 2 upper Bits are used
7	Trans_7_u	0x1D814	pass 7 LSBs; 1 upper Bit is used
8	Transparent	0x22CE0	Full Transparent Mode for all eight bits
9	Trans_1	0x2683C	pass 1 LSB; 7 upper Bits are free and unused
10	Trans_2	0x2A558	pass 2 LSBs; 6 upper Bits are free and unused
11	Trans_3	0x2E184	pass 3 LSBs; 5 upper Bits are free and unused
12	Trans_4	0x33990	pass 4 LSBs; 4 upper Bits are free and unused
13	Trans_5	0x37D4C	pass 5 LSBs; 3 upper Bits are free and unused
14	Trans_6	0x3B028	pass 6 LSBs; 2 upper Bits are free and unused
15	Trans_7	0x3F4F4	pass 7 LSBs; 1 upper Bit is free and unused
16	reserved	0x41D1C	reserved
17..31	reserved	Reserved	reserved

The IPE\_Mode is protected by the binary, systematic (16,5) block code with generator polynomial  $g(x) = x^{11} + x^7 + x^5 + x^4 + x^2 + x + 1$ . The minimum Hamming distance of this code is  $d_{min}=7$ , which allows the correction of up to 3 bit errors within each code word of length 16 bits.

Bits 1 (MSB) and 11 are the synchronisation bits and set to "0", see Figure A-9. The EX field is set to "0.0" in all currently defined IPE\_Modes, i.e. no further IS\_Extension\_Block is following.

Table A2.3-2 defines the coding in hexadecimal notation for the complete IPE\_Mode\_Extension\_Block, with EX := 00.



**Figure A.2.3-2: IPE\_Mode\_Extension\_Block for the IS\_IPE Message**

An IS\_IPE Message containing the NORMAL command is termed **IS\_NORMAL Message**.

An IS\_IPE Message containing a TRANS\_x command is termed **IS\_TRANS\_x Message**.

An IS\_IPE Message containing a TRANS\_x\_u command is termed **IS\_TRANS\_x\_u Message**.

The latter two are sometimes also termed **IS\_TRANS Message**, if the details are not important.

The behaviour of IPEs, when receiving such commands, is described in Annex B.

The first IS Message in a series is often "swallowed" by IPEs (see Annex B). An IS\_IPE Message must therefore never be the first message of a series of IS Messages, i.e. it shall be sent as an isolated IS Message or after a (sufficiently long) uninterrupted IS Protocol.

## A.2.4 IS\_FILL Message

The IS\_FILL Message has no IS\_Extension\_Block and no specific meaning. An IS\_Sender can use the IS\_FILL Message to fill a temporary gap in the protocol flow. This may be important to keep all IPEs in synchronization and open for further IS Messages, see Figure A-10. An IS\_FILL Message shall also be used by the IS\_Sender to resynchronize all IPEs in case of a phase shift of the Keep\_Open\_Indication.

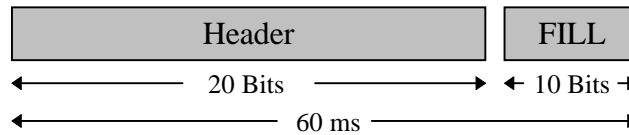


Figure A.2.4-1: Construction of the IS\_FILL Message

IS\_FILL is designed in a way that multiple repetitions cause minimal audible effects.

## A.2.5 IS\_DUP Message

The IS\_DUP Message may be used between IS Partners to indicate a half duplex mode. It may be especially important in Handover situations. The IS\_DUP Message has no IS\_Extension\_Block, see Figure A.2.5-1.

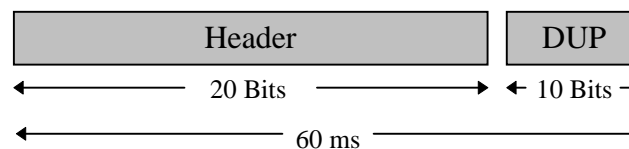


Figure A.2.5-1: Construction of the IS\_DUP Message

## A.2.6 IS\_SYL Message

The IS\_SYL Message may be used between IS Partners to indicate the loss of synchronisation. It may be especially important in Handover situations. The IS\_SYL Message has no IS\_Extension\_Block, see Figure A.2.6-1.

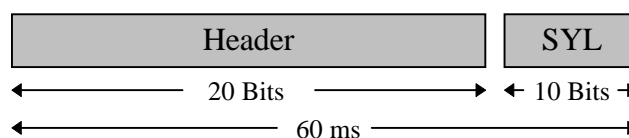


Figure A.2.6-1: Construction of the IS\_SYL Message

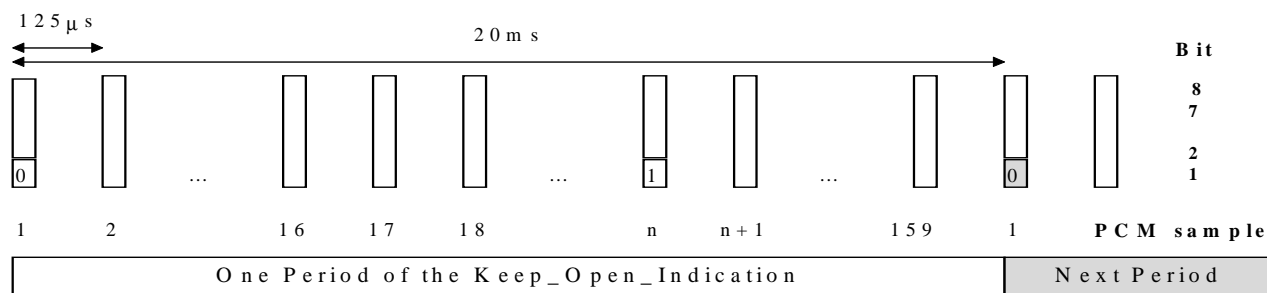
## A.3 Keep\_Open\_Indication

In Transparent\_Mode, i.e. after properly receiving an IS\_TRANS Message, all IPEs shall monitor the bypassing bit stream for the Keep\_Open\_Indication. If this Keep\_Open\_Indication is not seen for some time, then the IPEs shall fall automatically back into normal operation, i.e. the mode of operation before the IS\_TRANS Message.

This automatic fall back shall have the same effect as the IS\_NORMAL Message would have.

By definition the Keep\_Open\_Indication is a continuous bit stream of one "0"-Bit in the LSB of every 160<sup>th</sup> PCM sample, i.e. every 20 ms. At least one "1"-Bit must be present within the LSBs of the other 159 PCM samples, see Figure A.3-1.





**Figure A.3-1: Keep\_Open\_Indication**

The "0"-Bit stream of the Keep\_Open\_Indication shall always be present as long as the IPEs need to be in Transparent\_Mode.

The Keep\_Open\_Indication shall be in phase with the preceding IS Messages., i.e. the first bit of the Keep\_Open\_Indication shall be in the position of the first bit of the (hypothetical) next IS Message. In fact, the IS Messages themselves contain this Keep\_Open\_Indication by definition.

In case of a known phase shift of the Keep\_Open\_Indication, the IS\_Sender has to send at least one IS Message, which defines the new phase position of the Keep\_Open\_Indication. If no other IS Message is to be sent, then the IS\_FILL Message shall be used. If an IS Message longer than 160 ms is scheduled for transmission, then an IS\_FILL Message should be inserted before, to guarantee fast resynchronization of the IPEs.

## A.4 Rules for Sending of IS Messages

IS Messages replace some bits of the PCM samples and therefore cause a minimal signal distortion. Therefore IS Messages shall be used with care and not longer than necessary. The IS Protocol is kept to a minimum to avoid unnecessary complexity. One basic assumption is that only one IS Protocol is active at a time between two IS Partners.

Only specific telecommunication entities shall be allowed to initiate IS Protocols. They are called **IS\_Active** or active IS Partners. In principle these shall only be terminal devices or their "representatives" within the network. Examples are ISDN-Terminals, Speech-Servers and Transcoders (as representatives of the MSs).

Other telecommunication entities shall only react on IS Protocols. They are called **IS\_Passive**. Most IPEs are of this type. They bypass the IS Messages, they obey the IS\_IPE Messages, but they never initiate IS Messages.

Other telecommunication entities are IS\_Passive by default. But if they receive IS Protocols that they can understand, then they may become IS\_Active and start to initiate IS Protocols. They thus become active IS Partners and shall take care that only one IS Protocol is active on both of their sides. They are called **IS\_Responsive**. TCMEs are examples of such entities.

Active IS Partners shall send:

- either continuous sequences of IS Messages without interruption of the 16\_PCM\_Sample\_Grid; or
- isolated IS Messages with same message lengths; or
- isolated IS Messages with sufficient distance between them, if shorter IS Messages follow longer IS Messages.

The latter case is important, because shorter isolated IS Messages travel faster through IPEs than longer ones, see annex B.

As said above, after initialisation of an IS Message sequence, no interruption of the 16\_PCM\_Sample\_Grid shall occur within the sequence. Adjustments of the phase position of the Keep\_Open\_Indication shall be done only after the IS\_TRANS Message by inserting the necessary number  $n$  (with  $0 < n < 160$ ) of "1" Bits (termed "T\_Bits") into the LSBs of the PCM samples that have to be skipped. The first PCM sample for this insertion of T\_Bits is the one where the next regular IS Message or next regular Keep\_Open\_Indication would begin. At the new phase position the next IS Message or the IS\_FILL Message shall be sent, to allow IPEs to resynchronize fast, see Figure A.4-1.

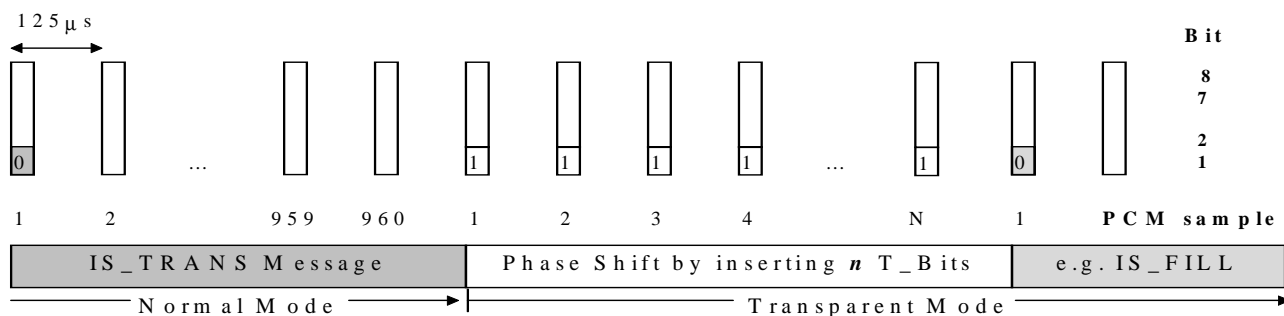


Figure A.4-1: Phase Shift of the 16\_PCM\_Sample\_Grid by inserting T\_Bits

Similarly, the adjustment of the phase between two Keep\_Open\_Indications shall be done by inserting the necessary number of T\_Bits and by sending an IS Message - preferably, but not necessarily - the IS\_FILL.

Finally a "negative" phase adjustment between two Keep\_Open\_Indications shall be allowed by shortening the cycle by a maximum of 2 PCM samples and sending an IS Message (see above) at the new phase position.

### A.5 System Identification and IS\_System\_Identification\_Block

The IS\_System\_Identification\_Block is a mandatory IS\_Extension\_Block for the IS\_ACK and IS\_REQ messages with the 16-bit Information\_Field containing the IS\_System\_Identification. It identifies the system within which the message is generated. Table A.5-1 shows the defined IS\_System\_Identification codes and the SysID as used in TFO16k Frames (see also Figures A.5-1 and A.5-2).

Table A.5-1: Defined SysID and IS\_System\_Identification Codes

System	SysID (S1..S8) (in binary)	IS_System_Identification (in hex)	
		if EX == "0.0"	if EX == "1.1"
<b>GSM</b>	0000.0000	0x53948	0x5394B
TIA/EIA-136 (TDMA)	0000.0001	0x53414	0x53417
TIA/EIA-95 (CDMA)	0000.0010	0x528AC	0x528AF
reserved	0000.0011	0x525F0	0x525F3
<b>UMTS</b>	0000.0100	0x51C80	0x51C83
reserved	0000.0101	0x511DC	0x511DF
reserved	0000.0110	0x50D64	0x50D67
reserved	0000.0111	0x50038	0x5003B

All other codes are reserved. Additional IS\_System\_Identification Codes for other systems shall be defined in a way that the audibility is minimal and the hamming distances to the already defined codes is maximal.

The SysID is protected by the binary, *systematic* (16,8) block code with generator polynomial  $g(x) = x^8 + x^7 + x^6 + x^4 + x^2 + x + 1$ . The minimum Hamming distance of this code is  $d_{min}=5$ , which allows the correction of up to 2 bit errors within each code word of length 16 bits. The first, upper eight bits represent the systematic part, the following lower eight bits the redundant part of the code words.

The resulting 16 bits are placed into the IS\_System\_Extension\_Block and then the whole 20 bit word is additionally EXORed with the fixed code word 0x53948 to minimize audible effects. The final result gives the IS\_System\_Identification and is shown in Figure A.5-1 for GSM and A.5-2 for UMTS.

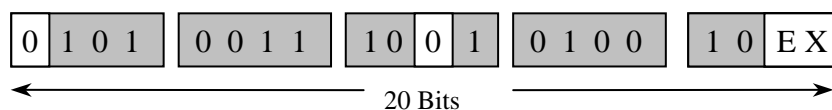
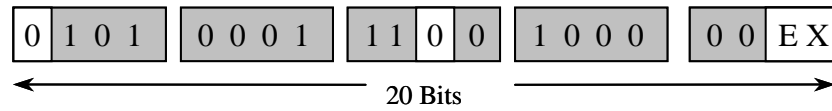


Figure A.5-1: IS\_System\_Identification for GSM



**Figure A.5-2: IS\_System\_Identification for UMTS**

Please note that the *systematic part* is also used within the TFO16k Frames (S1...S8) of GSM, TIA/EIA-136, TIA/EIA-95 and UMTS systems for System Identification, see main part of the document.

---

## Annex B (informative): In Path Equipment: Generic Rules and Guidelines

Scope: See Annex A

---

### B.1 Types of In Path Equipment

The term "In Path Equipment" (IPE) is used for any telecommunication equipment within the (64 kbit/s) transmission path for the speech signal between two entities, which want to communicate via IS Messages, i.e. the IS Partners.

In modern telecommunication networks most of these IPEs are digitally transparent for the complete 64 kbit/s data stream all the time after call establishment until call release. These IPEs are optimal and need no consideration here.

Some IPEs are most of the time digitally transparent, but disturb the link every now and then. Examples are:

- switches, which interrupt the link during Handover;
- switches, which insert a kind of conference bridge for a short while during Handover;
- links, which do octet deletions or insertions (octet slips);
- DTMF generators, which insert DTMF tones sometimes for a short while.

Other IPEs are digitally transparent in one direction, but not in the other. Examples are:

- DTMF generators, which insert the DTMF tones only in one direction;
- Network Echo Cancellers (NECs), which let the signal pass unaltered towards the PSTN, but cancel the echo.

Other IPEs are semi-transparent, i.e. let most or some of the bits pass, but not all. Examples are:

- A/ $\mu$ \_Law converters;
- $\mu$ /A\_Law converters; and
- especially the tandem connection of A/ $\mu$ \_Law and  $\mu$ /A\_Law converters, or vice versa.
- links, which insert inband signalling by bit stealing (T1 links).

Other IPEs are not transparent at all to the digital bit stream, although the speech signal pass more or less unaltered.

EXAMPLE 1: level shifters, which adjust the signal levels, e.g. between national networks.

EXAMPLE 2: DCMEs (Digital Circuit Multiplication Equipment), which compress the bit stream by encoding/decoding the speech signal for cost efficient transmission.

Many of these IPEs - for some time - will be not compliant with the IS Message principle described in Annex A. The IS Messages will not pass these non-compliant IPEs or not in both directions, or not always. Care must be taken to identify situations where IPEs are part-time-transparent or semi-transparent, when applying IS Messages. Other IPEs - at some point in time in the future - will be compliant to the IS Message principle. The rules they have to fulfil are described below.

## B.2 IS\_Compliant IPEs

### B.2.1 Typical IPEs are IS\_Passive

In general, an IPE shall *never* actively initiate the exchange of IS Messages. The active initiation is only done by terminals or their "representatives". This avoids uncontrolled and unnecessary fluctuation of IS Messages within the network.

Most IPEs shall never actively respond to IS Messages by sending other IS Messages. Such equipment are called **IS\_Passive**.

They need not and do not understand the IS Protocol, but let it just pass unaltered and obey the relevant IS\_IPE Messages.

Some IPEs may, however, respond on received IS Messages, modify these and/or respond with own IS Messages, if they understand the IS Protocol and can take or bring advantage to the overall system performance or system quality. These IPEs are called **IS\_Responsive**. Examples are GSM-specific Digital Circuit Multiplication Equipments (TCMEs), which reduce transmission costs without degrading the speech quality. These IPEs may be able to step into the IS Protocol, interpret and respond to it and modify the speech signal in a system compliant way. Thus they become **IS\_Active** Partners themselves.

### B.2.2 IS Message\_Transparency

When commanded into a Transparent Mode, the IPEs are fully transparent at least for the LSBs in all PCM samples. Therefore the following rules are needed only and only do apply for the IPEs, when in Normal\_Mode:

- IPEs shall let the IS Messages bypass, or re-insert them, from their input to their respective output.
- They shall not alter them, nor do any kind of error correction. Exceptions are the IS\_Responsive IPEs.

#### B.2.2.1 First IS Message

During its Normal\_Mode an IS\_Compliant IPE shall always monitor the incoming PCM data stream for the occurrence of the IS\_Header sequence. If the IS\_Header is detected after a period without IS Messages, the IPE shall store the following IS\_Command and IS\_Extension\_Block(s). During reception of this first IS Message, the normal operation of the IPE is maintained with the consequence that the first IS Message may not appear at the output of the IPE.

#### B.2.2.2 IS Messages within a Sequence

All further IS Messages which follow directly after the first detected IS Message in the same phase position shall be passed unaltered to the output of the IPE with exactly that delay the IPE would later introduce when commanded into Transparent\_Mode by one of the IS\_TRANS commands, see Figure B.2.2.2-1.

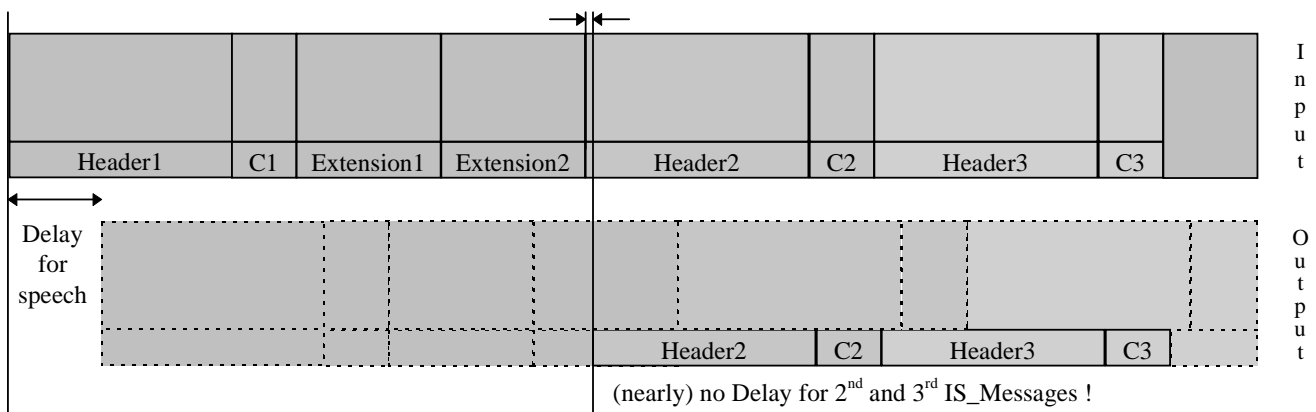


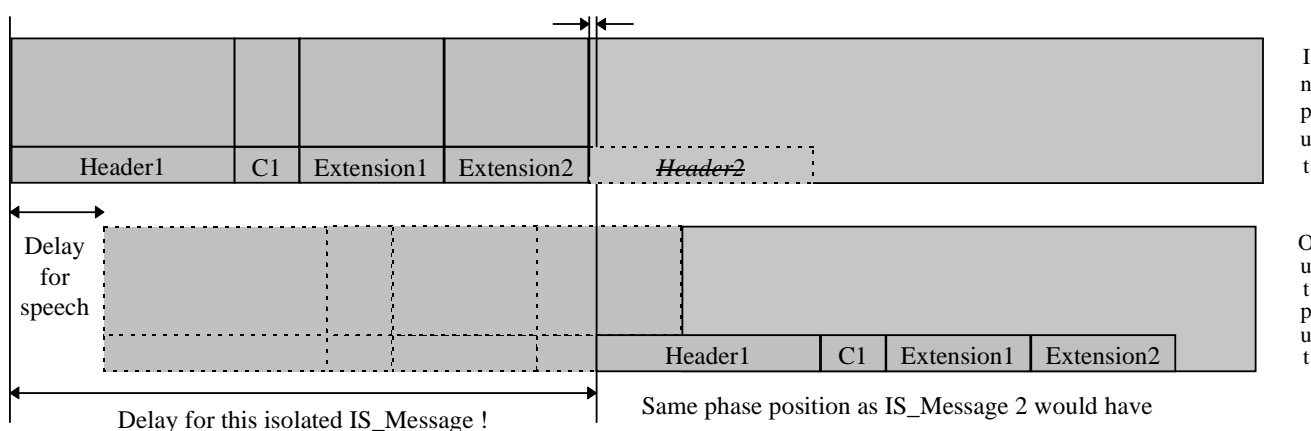
Figure B.2.2.2-1: Transparency and Delay for first and following IS Messages

The upper row symbolizes the speech signal at the input of the IPE, with the PCM samples drawn vertically and the IS Messages inserted into the LSBs. The lower row symbolizes the speech signal at the output of the IPE. The vertical lines denote the boundaries of the IS Message elements.

Figure B-1 shows an example where the first IS Message is detected, but not passed through. The distortion caused by the first IS Message is still "somehow" there (indicated by the empty dashed boxes in the LSB), but the message is destroyed. The second and third IS Messages are passed through unaltered. Note, however, that the delay of the speech signal is (in this example) substantially higher than the delay of the IS Messages. They travel faster than the speech signal through this IPE.

### B.2.2.3 Isolated IS Message

In cases where the first detected IS Message is not immediately followed by further IS Messages, the IPE shall insert this first IS Message (which the IPE has stored) into its output PCM bit stream, with exactly the delay and phase position a second IS Message would have, see Figure B.2.2.3-1, which shows an example where an isolated IS Message is travelling through an IPE.



**Figure B.2.2.3-1: Transparency and Delay for an isolated IS Message**

Note that the delay of an isolated IS Message is depending on its own length! Longer IS Messages will have more delay, shorter less. It could, in principle, happen that a second, shorter isolated IS Message would "bypass" the first longer IS Message - with the consequence that the first one would be destroyed. This is especially important when there are several IPEs in the path, since the delay effects accumulate. Therefore it is not allowed to send shorter isolated IS Messages too close after longer IS Messages. IS Messages with same length have no restriction.

In summary, the first IS Message in a series of IS Messages is "swallowed" by an IPE, while all the following IS Messages pass unaltered and with minimal delay. If an IS Message occurs isolated, then it is not swallowed, but delayed by exactly its own length. The latter mechanism ensures that isolated IS Messages can pass through an unlimited number of IPEs.

### B.2.2.4 Check if IS Message is following

The checking, whether an other IS Message is following or not is done "on the fly", i.e. bit by bit. This is possible due to the fact that all messages begin with exactly the same IS\_Header. The decision, whether an IS Message is an isolated message or the first message in a series, can be done latest after the last bit of the (next) IS\_Header, see Figure B2.2.3-1.

Consequently, after detection of the first IS Message, the IS\_Header is in any case inserted at the output in the correct position, regardless, whether a second message follows or not.

## B.3 IPE State Representation

Concerning the IS Protocol, an IPE can be described with five major States in two main Modes, where the States describe the IPE with respect to the IS Protocol and the Modes describe the IPE with respect to the operation on PCM data. Figure B.3-1 shows a graphical representation of the State diagram of an IPE.

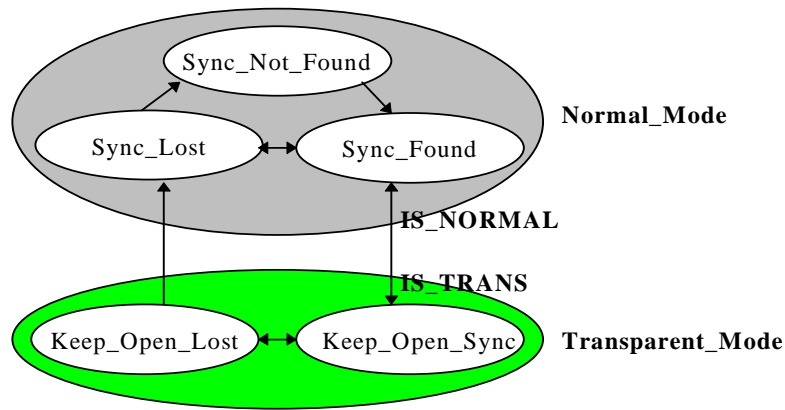


Figure B.3-1: Principle of a State Diagram of an IPE

### Some Definitions:

An IS Message shall be recognized as "*error-free*", if no error can be detected, neither within the IS\_Header, nor in the IS\_Command nor in any IS\_Extension\_Block.

An IS Message shall be recognized as "*single-error*", if no more than one bit position differs in the IS\_Header or the IS\_Command\_Block or the IPE\_Mode\_Block or one EX-field or one Sync bit.

An IS Message shall be recognized as "*correctable*", if the phase position is as in preceding IS Messages and:

- no more than 2 bit position differs in the IS\_Header; and
- no more than 1 error is detected within the IS\_Command\_Block; and
- no more than 3 errors are detected within the IPE\_Mode\_Block; and
- no more than 0 error is detected within the EX-field(s); and
- no more than 1 error is detected within the Sync-Bit(s); and
- the total number of detected errors is not higher than 3.

IS Messages, which are error-free, single-error or correctable are also called "*valid*" IS Messages.

An IS Message shall be recognized as "*present*", if the phase position is as in preceding IS Messages and:

- no more than 4 bit position differs in the IS\_Header and
- no more than 2 errors are detected within the IS\_Command\_Block; and
- no more than 3 errors are detected within the IPE\_Mode\_Block; and
- no more than 1 error is detected within the EX-field(s); and
- no more than 2 errors are detected within the Sync-Bit(s); and
- the total number of detected errors is not higher than 4.

Sequences, which differ in more than "*present*" are not recognized as IS Messages at all, i.e. "*not\_present*".

Note that the insertion of T\_Bits may change the phase position of an IS Message. The IS Message shall in that case be classified after the removal of the T\_Bits.

An octet slip may also change the phase position of an IS Message. If an error-free or a single-error IS Message can be found after considering a hypothetical octet slip ( $\pm 1$  sample), then it may be regarded as error-free or single-error and the new phase position shall be regarded as valid, if no valid or present IS Message can be found at the old phase position.

### B.3.1 IPE in Sync\_Not\_Found

After start-up or after a long interruption of the IS Protocol an IPE is in Normal\_Mode, performing its normal operation. IS Messages have not been found and consequently no bypassing of IS Messages is performed.

The algorithm for initial synchronization shall be able to detect each single IS Message, especially the first or an isolated one. An IPE shall always, during Normal\_Mode and during Transparent\_Mode, search for the IS\_Header and consequently for complete IS Messages. When found, it can be assumed that with high probability the following IS Messages and the Keep\_Open\_Indication will stay within the found "grid" or "phase" of every 16<sup>th</sup> PCM sample, the 16\_PCM\_Sample\_Grid.

An IPE transits from Sync\_Not\_Found into Sync\_Found, if and only if an error\_free IS Message is detected. Then the IPE lets the following IS Messages bypass, as described above.

If the first IS Message is an error\_free IS\_TRANS Message, then the IPE transits directly into the Transparent\_Mode.

### B.3.2 IPE in Sync\_Found

The IPE continues its normal operation, but opens an IS\_Door every 16<sup>th</sup> LSB for the bypassing IS Messages.

An IPE shall regard sync as continued, i.e. stay in Sync\_Found, if after each IS Message another valid IS Message follows within the same phase position, i.e. within the 16\_PCM\_Sample\_Grid.

For any deviations from a valid IS Message, the IPE transits to Sync\_Lost.

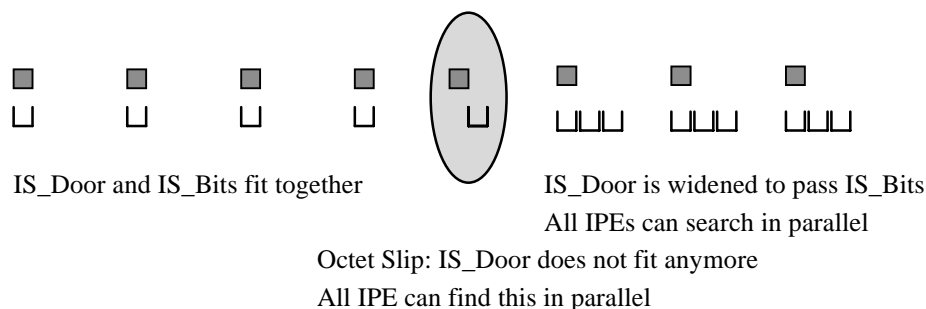
If an error\_free or correctable IS\_TRANS is received in Sync\_Found, then the IPE transits into the Transparent\_Mode.

### B.3.3 IPE in Sync\_Lost

In Sync\_Lost, an IPE shall search for IS Messages on all positions as for initial synchronisation. In parallel, an IPE shall bypass not\_valid, but present IS Messages at the found phase position for a maximum of one second. An IPE shall close the IS\_Door after that, if no valid IS Message is following, i.e. transit into Sync\_Not\_Found.

A single valid IS Message brings the IPE back into Sync\_Found.

As soon as the IPE detects in Sync\_Found or in Sync\_Lost a single or more deviations from an error\_free IS Message, then the IPE may optionally open the IS\_Door also at positions ±1 around the present (0) phase position for a maximum of one second ] to allow other IPEs in the path for parallel re-synchronization, see Figure B.3.3-1. The IPE may try to find a continuation of the disturbed IS Message at these 3 positions. If the IPE can detect an error-free or a single-error IS Message in this way, then it shall accept the new phase position, if no IS Message can be found at the old phase position anymore.



**Figure B.3.3-1: Handling of octet slip for fast and parallel re-synchronization of all IPEs (optional)**



### B.3.4 IPE in Keep\_Open\_Sync

The IPE enters this State by receiving a valid IS\_TRANS Message. This is the main State of the Transparent\_Mode.

It depends on the specific IPE, if this Transparent\_Mode is active only for the commanded direction (that is the default assumption) or in both directions (because for a specific IPE it might be useless or impossible to maintain Normal\_Mode in one direction and Transparent\_Mode in the other one).

The IPE shall bypass the commanded LSBs and handle the upper bits accordingly (IPE specific).

The IPE shall search in parallel for IS\_IPE Messages (IS\_TRANS, IS\_NORMAL) and

transit - if necessary - to Normal\_Mode or an other Transparent\_Modes (other number of transparent LSBs).

The IPE shall monitor the bypassing bit stream for the Keep\_Open\_Indication and accept the Keep\_Open\_Indication only at the phase position defined by the preceding IS Message.

If the Keep\_Open\_Indication is not seen anymore then the IPE transits into Keep\_Open\_Lost.

### B.3.5 IPE in Keep\_Open\_Lost

The IPE shall continue its operation in Transparent\_Mode and Keep\_Open\_Lost for a maximum of one second before it shall return to Normal\_Mode. During that time the IPE shall try to resynchronize either by finding an IS Message or by finding the Keep\_Open\_Indication at positions  $\pm 1$  and 0 around the present phase position (handle of Octet Slip).

The IPE may take advantage of the fact that T\_Bits are inserted or deleted by the IS\_Sender in case of an intentional phase adjustment.

An IS Message at any arbitrary phase position followed by a valid Keep\_Open\_Indication is accepted as re-defining the Keep\_Open phase position, if and only if the Keep\_Open\_Indication is no longer present at the old phase position.

A Keep\_Open\_Indication at a phase position  $\pm 1$  PCM sample interval around the old phase position is accepted as re-defining the Keep\_Open phase position, if and only if the Keep\_Open\_Indication is no longer present at the old phase position.

The Keep\_Open\_Indication is "valid", as long as at least 40 "0"-Bits are seen at the correct positions within a sliding window of length of one second. At least one "1"-Bit must be seen in between each pair of the expected "0"-Bits.

---

## B.4 IPE Error Handling

The first IS\_Message shall only be accepted, if there is no detectable error.

For all following IS\_Messages it shall apply:

Errors in IS Messages shall be passed unaltered through the IPEs. This shall hold for all IS Messages.

Only error-free or correctable IS\_IPE Message shall be applied by the IPE to its own operation. Other IS\_IPE Messages shall be ignored, but bypassed.

---

## B.5 IPE Transmission Delay

The transmission delay introduced by an IPE for the speech, audio or modem signal is in general different in Normal\_Mode and Transparent\_Mode. Some IPEs may have several different Normal\_Modes with possibly different signal delays. IS Messages are transmitted within the regular 16\_PCM\_Sample\_Grid. It is important that this regularity is not disturbed. Therefore care must be taken at the transition between these modes.

The transmission delay of a specific IPE is in general lower for IS Messages than for speech, audio or modem signals.

## B.5.1 IPE Transmission Delay in Normal\_Mode

The delay for IS Messages in Normal\_Mode shall be identical to the delay in that Transparent\_Mode, that follows after the first IS\_TRANS Message. If different Transparent\_Modes with different delays could follow, then the shortest delay of all possible Transparent Modes shall be selected for IS Messages in Normal\_Mode.

If an IPE in Normal\_Mode has to change its transmission delay, then this shall not affect the delay of the IS Messages.

## B.5.2 IPE Transmission Delay in Transparent\_Mode

In the majority of all cases the IPE will keep the transmission delay for the IS Messages in Normal\_Mode also in Transparent\_Mode for the transmission of the commanded transparent LSBs. IPEs which do not understand the IS Protocol shall never modify the transparent bits, so they are also not allowed to change delay.

Some IPEs, which understand a specific IS Protocol, may have even different Transparent\_Modes and also here the transmission delays may differ. TCMEs are an examples of such equipment.

If an IPE has to change its transmission delay at the transition from Normal\_Mode to Transparent\_Mode, then the IPE shall readjust the phase of the Keep\_Open\_Indication after transition into the Transparent\_Mode with higher delay by inserting the relevant number of T\_Bits after the first IS\_TRANS Message and before the next IS Message. If no other IS Message is following, then the IS\_FILL shall be inserted, obeying all other relevant rules of the specific IS Protocol (e.g. EMBED bit C5 in TFO Frames).

If an IPE has to change from one Transparent\_Mode to an other one with a different transmission delay, then the IPE shall readjust the phase of the Keep\_Open\_Indication after transition into the new Transparent\_Mode by inserting the relevant number of T\_Bits. If no other IS Message is following, then the IS\_FILL shall be inserted at the new phase position to mark the new grid position of the 16\_PCM\_Sample\_Grid and to allow other IPEs to resynchronize, obeying all other relevant rules of the specific IS Protocol (e.g. EMBED bit C5 in TFO Frames).

---

## B.6 Compliance to IS Messages

An IS\_Compliant IPE shall be capable of interpreting and obeying the IS\_IPE Messages.

It depends on the intelligence and task of an IPE, how many and which of the other IS Messages it needs to understand.

The IPEs shall synchronise to all IS Messages, especially to find or refind the Keep\_Open\_Indication. All IPEs shall resynchronize, if they see an IS Message in a new phase position, and if the synchronization can not be found in the old phase position anymore.

### B.6.1 Compliance to IS\_REQ and IS\_ACK Messages

Most IPEs need not and do not understand these messages. They just synchronise to them and let them pass unaltered.

Only IS\_Responsive IPEs may take advantage. This is system specific and IPE specific.

### B.6.2 Compliance to IS\_NORMAL Message

The IPE shall act in response to the receipt of an IS\_NORMAL Message such that:

- The IPE shall synchronise to it. The message shall appear unchanged at the output of the IPE.
- The IPE shall resume its Normal\_Mode of operation for all data received subsequent to the IS\_NORMAL Message, until a different command is received.

It depends on the type and operation of the specific IPE, whether the Normal\_Mode is resumed in both directions, or only in the direction in which the IS\_NORMAL Message flows. It must be assumed that in general only this one direction is affected.

### B.6.3 Compliance to IS\_TRANS\_x Messages

The IPE shall act in response to the receipt of an IS\_TRANS\_x Message (x in the range 1 to 8) such that:

- The IPE shall synchronise to it. The IS\_TRANS\_x Message shall appear unchanged at the output of the IPE.
- The IPE shall be transparent in all x LSBs of all PCM samples received subsequent to the IS\_TRANS Message.
- The transparency shall persist as long as the Keep\_Open\_Indication persists, or until a different command is received.
- The (8-x) upper bits of the PCM samples are not of interest and may be modified arbitrarily by the IPE.

It depends on the type and operation of the specific IPE, whether the Transparent\_Mode is resumed in both directions, or only in the direction in which the IS\_TRANS Message flows. It must be assumed that in general only this one direction is affected.

### B.6.4 Compliance to IS\_TRANS\_x\_u Messages

The IPE shall act in response to the receipt of an IS\_TRANS\_x\_u Message (x in the range 1 to 7) such that:

- The IPE shall synchronise to it. The messages shall appear unchanged at the output of the IPE.
- The IPE shall be transparent in all x LSBs of all PCM samples received subsequent to the IS\_TRANS Message.
- The transparency shall persist as long as the Keep\_Open\_Indication persists, or until a different command is received.

The (8-x) upper bits of the PCM samples are important and in general shall not be modified by the IPE, but shall be bypassed transparently in exactly the same manner and delay as the x LSBs. It is important that this transparency for the upper bits is provided by IPEs that do not understand the specific IS Protocol (e.g. do not understand the IS\_System\_Identification or the protocol of the transmitted parameters).

Only IPEs which do exactly understand the specific IS Protocol shall take advantage of the opportunities given with the IS\_TRANS\_x\_u Messages. An example is the TCME, which transmits internally only the coded speech parameters and re-generates the upper x bits at its output (termed here as "first solution"). The resulting delay in the upper 8-x bits shall be identical to the delay in the x LSBs.

If this transparency of the upper (8-x) bits or their re-generation can not be established, then the upper bits shall contain a constant pattern, giving the least output energy (PCM\_Silence). This "second solution" may cause temporary interruptions of the speech signal in some transition cases (e.g. hand over in some tandem free GSM mobile-to-mobile calls). Therefore the first solution is the preferred one.

IPEs, which implement the second solution shall switch to the full transparent 64 kbit/s channel as soon as they lose synchronisation with the protocol of the transmitted parameters (e.g. the "TFO Frames" in GSM Systems). The full transparency shall be executed for both directions. The near side shall be fully transparent in less than 60 ms and the other side the one way delay of that IPE later.

It depends on the type and operation of the specific IPE, whether the Transparent\_Mode is resumed in both directions, or only in the direction in which the IS\_TRANS Message flows. It must be assumed that in general only this one direction is affected.

### B.6.5 Compliance to IS\_FILL Message

The IS\_FILL Message has no specific meaning, but may serve for two purposes.

First of all, it can be used to close the gap in an IS Protocol to keep all IPEs synchronized. Otherwise - in case of an interruption - the *n* IPEs in the path would swallow the next *n* IS Messages again.

Second, an IS\_FILL Message can be used to resynchronize all IPEs to a new grid position, if necessary.

## B.6.6 Compliance to IS\_DUP Messages

The IS\_DUP Message is sent by an IS Partner to the distant IS Partner to inform about a specific Half\_Duplex reception.

Most IPEs need not and do not understand this message. They just synchronize to it and let it pass unaltered.

Only IS\_Responsive IPEs may take advantage. This is system specific and IPE specific.

## B.6.7 Compliance to IS\_SYL Messages

The IS\_SYL Message is sent by an IS Partner to the distant IS Partner to inform about a specific Sync\_Lost Situation.

Most IPEs need not and do not understand this message. They just synchronize to it and let it pass unaltered.

Only IS\_Responsive IPEs may take advantage. This is system specific and IPE specific.

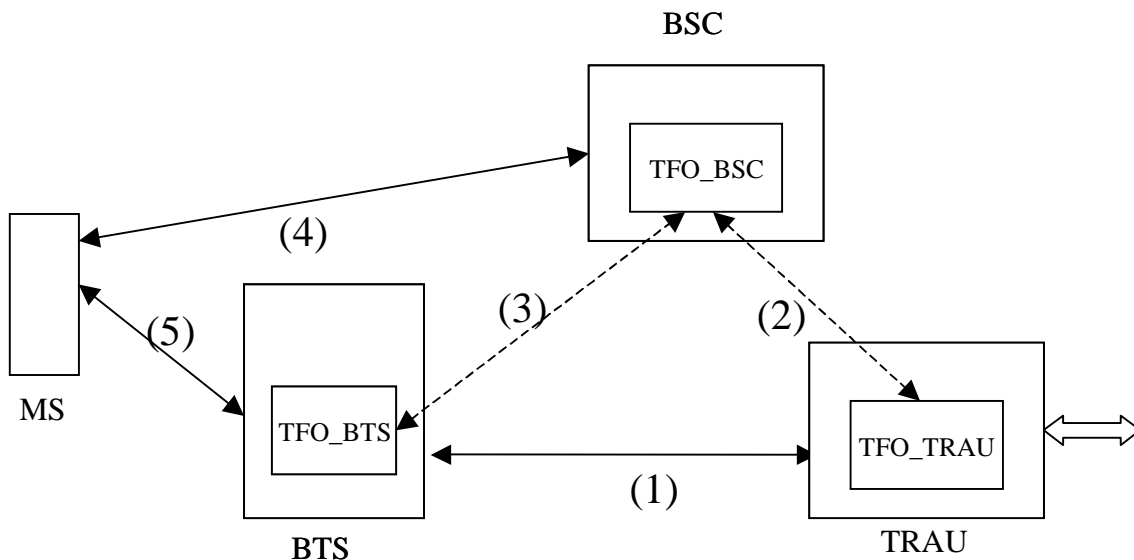
## Annex C (normative): Tandem Free Operation in GSM, including AMR-WB

### 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.

TFO in GSM involves the following processes:

- **TFO\_TRAU**: Mandatory for all Speech Codec Types
- **TFO\_BTS**: Not existent for GSM\_FR, GSM\_HR and GSM\_EFR. Some parts are mandatory, some are optional for the AMR and AMR-WB Speech Codec Type
- **TFO\_BSC**: Optional for all Speech Codec Types

## C.2.1 TFO\_TRAU

Tandem Free Operation is essentially managed by the TRAU. In the simplest implementation version the TRAU can establish and maintain TFO fully on its own (within certain limits) as described below.

For all Codec Types the TRAU is responsible for the inband TFO Protocol, i.e. the TFO negotiation, TFO setup and the fast fall back to normal operation, if necessary. The TRAU has to monitor the ongoing call permanently for fast reaction, if required.

In all cases the TRAU has to perform the TFO Decision algorithm (see clauses 11 and 12). This TFO decision algorithm takes all known local and distant configuration parameters into account and identifies whether TFO is possible and what are the optimal call configuration parameters (Optimal Codec Type and Codec Configuration) in the given situation. The TRAU has the responsibility to inform the BSC (and the BTS) about any change in the distant call configuration. It is optional for the BSC and the BTS to evaluate this information.

The TRAU may provide to the BSC and the BTS the optimal call configuration parameters resulting from the TFO Decision algorithm. It is optional for the BSC and/or BTS to evaluate these parameters. See also Annex D (TFO in UMTS) where the TC runs the TFO Decision algorithm and may provide the optimal configuration parameters to the serving MSC.

In case of the AMR and AMR-WB Codec Types the TRAU is responsible for the TFO relevant Rate Control. It shall limit the maximally allowed Rate (Codec Mode) in a way that it is always within the Common Active Codec Set of both sides. During TFO Konnect the TRAU is responsible to steer the uplink rate down to the TFO Setup Mode and release it as soon as TFO is in Operation.

If informed by the BSC with Pre-Handover Notification (optional), the TRAU is responsible for a safe handover in TFO by steering the uplink and downlink rates down into the Handover Mode, to fit best after handover.

## C.2.2 TFO\_BSC

The BSC has the overall responsibility, especially for all resources, on the radio channel and the BSS. For all Codec Types the BSC is responsible for Call Setup and for the support of BTS and TRAU with the necessary configuration parameters (Codec Type, Codec Configuration, alternative Codec List, radio channel capacity, Abis channel capacity, etc). The BSC is responsible to enable or disable TFO.

The BSC is responsible for Handover and should take care that the call configuration is not modified during handover unless absolutely necessary, because every local change has direct influence on the distant side.

The BSC is responsible that TFO is properly terminated before handover, if the call configuration after handover is not longer TFO compatible. This feature is optional. The BSC may delegate this responsibility to the TRAU, but this can only perform optimal, if supported by Pre-Handover Notification (optional).

The BSC is responsible for the change of the Codec Type, e.g. for Mismatch Resolution and Optimisation for TFO, if this is required or better for Tandem Free Operation. This feature is optional. This modification needs to be performed by BSS-MS Layer 3 signalling (Intra-cell Handover).

For the AMR and AMR-WB Codec Types the BSC is responsible for the change of the AMR configuration, if this is required or better for Tandem Free Operation. This feature is optional; it is signalled by the Optimisation Mode. If the BSC signals that it is willing to change, then it shall perform the change when necessary. The change may be performed by BSS-MS Layer 3 signalling (Intra-cell Handover or Mode Modify) or by BTS-MS inband signalling (RATSCCH). The BSC may delegate the responsibility for changes of the AMR Configuration temporarily or fully to the BTS (optional). If this option is selected, then the BSC shall guarantee that the MS gets the correct and consistent configuration after local handover. This may be achieved by the BSC in two ways: either by withdrawing this responsibility from the BTS before every local handover in order to guarantee that the BTS terminates a potentially ongoing configuration modification properly; or by providing the full set of Configuration parameters for the time after handover to the MS and new BTS.

### C.2.3 TFO\_BTS

The BTS is not specifically involved in TFO processes for the Non\_AMR Codec Types ( GSM\_FR, GSM\_HR, GSM\_EFR).

For the GSM AMR and AMR-WB Codec Types (FR\_AMR, HR\_AMR, FR\_AMR-WB) the BTS is responsible for the following functions. Some are optional.

The BTS receives the Codec Type and Codec Configuration from the BSC. The BTS shall send them in Config Frames uplink to the TRAU.

NOTE: The term "Config Frame" is used whenever configuration data are exchanged between BTS and TRAU, although in some Codec Modes these configuration data can be embedded into speech frames. But this is not relevant for the procedures and the understanding.

The BTS is responsible for the Rate Control concerning its local uplink and downlink radio interface.

The BTS shall take the Rate Control commands (CMR) from the TRAU into account, regardless whether TFO is ongoing or not. By this the TRAU can steer the Codec Mode (Rate) into the TFO Setup Mode (before TFO) and into the Handover Mode (in TFO, if informed properly by the BSC), and the TRAU can keep the Rates within the Common Active Codec Set.

The BTS shall suspend Time Alignment, when TFO is announced or established by the TRAU. Instead the BTS shall buffer the downlink TRAU frames for the proper transmission on the air interface. The BTS may perform phase alignment on the downlink radio interface by RATSCCH to optimise the downlink speech delay. This feature is optional.

The BTS shall perform bad frame handling and SID and No\_Data frame handling in downlink.

The BTS has the (optional) ability to perform a traffic synchronised modification of the AMR Configuration (Active Codec Set) by the RATSCCH protocol without interrupting the speech communication. This is important in TFO situations where the distant TFO Partner modifies its AMR Configuration relatively often. This RATSCCH protocol can be triggered by the BSC. If delegated by the BSC to the BTS the RATSCCH protocol can be triggered by the BTS itself, or by the TRAU. The latter two options reduce the signalling and computational load of the BSC.

## 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:

TFO Decision by →	TRAU (always necessary)	BTS (optional)	BSC (optional)
↓ Execution of change by			
TRAU (only Rate Control)	Version 0	-	-
BTS (only Configuration change by RATSCCH)	Version 5	Version 3	Version 2
BSC (Codec Type change by Layer 3 and Configuration change by Layer 3)	Version 6 (used in UMTS)	Version 4	Version 1

**Version 0, TRAU decided, no change:** The TRAU gets the distant Codec Type and Codec Configuration and runs the TFO Decision algorithm. No change of Codec Configuration or Codec Type is allowed. The TRAU may only limit the maximally allowed Codec Mode via Rate Control.

**Versions 1 and 2, BSC decided:** The BSC gets the distant Codec Type and Codec Configuration from the TRAU and runs the TFO Decision algorithm (in addition to the TRAU). If necessary the BSC modifies the Codec Type (including the Codec Configuration) by Intra Cell Handover (Version 1 only). If only the Codec Configuration has to be changed, the BSC can do this either by Intra Cell Handover or by Mode Modify (Version 1) or by RATSCCH (Version 2).

NOTE 1: These versions provide the slowest Codec Configuration modification on interface (5), due to the signalling on interface (3) and potential latency time within the (loaded) BSC. They generate some signalling load on interfaces (3) and (4) and some computational load within the BSC. The AMR internal Rate Control and Configuration problems are clearly visible for the BSC. The BSC has full control. Intra Cell handover for Codec Configuration modification requires radio capacity and some interruption of the speech path. Mode Modify for this purpose does not guarantee a synchronised update in MS and BTS. In both cases it is recommended to terminate TFO before, if ongoing.

The TFO Decision algorithm must be implemented and updated identically in TRAU and BSC to get consistent results.

**Versions 3 and 4, BTS decided:** If delegated so by the BSC the BTS has to run the TFO Decision algorithm (in addition to the TRAU) and has to perform Configuration Optimisation and Modification by the RATSCCH protocol (Version 3). In this case the BTS has to inform the BSC after each successful modification on the radio interface. The BSC can suspend this BTS process at any time. It may be necessary to suspend it by the BSC especially before handover and delegate it after handover again. In cases when the Codec Type must be modified, the BTS must send the Optimal Codec Type and Codec Configuration to the BSC for the modification and shall not perform any modification itself (Version 4).

NOTE 2: Version 3 provides the fastest Codec Configuration modification on interface (5) with minimal signalling on interfaces (3) and (4) and minimal computational load within the BSC. It hides AMR internal Rate Control and Configuration problems for the BSC. The BSC has not to run the TFO Decision algorithm, but the BTS. Version 4 is similar to version 1 in timing.

**Versions 5, TRAU decided, BTS executed:** The TRAU has to run the TFO decision algorithm anyway. It sends the Optimal Codec Type and Codec Configuration down to the BTS. This eliminates the need to run the TFO Decision algorithm in the BTS and/or BSC again. In cases when the Codec Type must be modified, the BTS must send the Optimal Codec Type and Codec Configuration to the BSC for the modification and shall not perform any modification itself (see Version 6).

If delegated by the BSC the BTS has to perform Codec Configuration modification (if the Codec Type does not change) by the RATSCCH protocol. In this case the BTS has to inform the BSC after each successful modification. The BSC can suspend this BTS process at any time. It must be suspended by the BSC especially before handover and delegated after handover again.



NOTE 3: This version provides the fastest Codec Configuration modification on interface (5) with minimal signalling on interfaces (3) and (4) and minimal computational load within the BTS and BSC. It hides AMR internal Rate Control and Configuration problems for the BSC. The BTS and the BSC do not have to run the TFO Decision algorithm. This version is preferred in networks with different configurations in neighbouring cells and/or the TFO partners, where the configuration changes often during handovers, especially at the distant side.

**Version 6, TRAU decided, BSC executed:**

The TRAU has to run the TFO decision algorithm anyway. It sends the Optimal Codec Type and Codec Configuration down via the BTS to the BSC, or via a proprietary TRAU-BSC interface directly to the BSC. This eliminates the need to run the TFO Decision algorithm in the BTS and BSC again. The further procedures are as in version 1, BSC executed.

NOTE 4: The TFO Decision algorithm must only be implemented and updated in one unit, the TRAU. This guarantees consistency. The BTS and BSC functions for TFO remain relatively simple. This version is preferred in networks with identical or compatible configurations in neighbouring cells and similar TFO partners. It performs best if the configuration do not have to be changed during handovers on both sides. In the optimal case (full AMR set in all cells) the Codec Configuration need not to be modified at all and the TFO\_BSC and TFO\_BTS processes disappear.

This version is used for TFO in UMTS (see Annex D).

These different processes as well as the inter-processes dialogues are described in the following clauses in detail.

---

## C.3 TFO\_TRAU

The following clauses describe the actions within the TRAU to establish and maintain Tandem Free Operation in terms of a State Machine, respectively TFO Processes, handling synchronisation and protocol. The description of the TFO Protocol does not reflect implementation details for the I/O Processes (Rx\_TRAU, Tx\_TRAU, Tx\_TFO, and Rx\_TFO), but they may need to be considered for the exact understanding of the behaviour. Only the TFO\_Protocol Process is detailed, which is responsible for the handling of the TFO Protocol.

The TFO\_TRAU can be regarded as consisting of five processes, which are strongly coupled to each other, which run in parallel, but phase shifted. The TFO\_Protocol Process communicates with the TFO I/O processes and, optionally, with its corresponding process within the BSS (TFO\_BSC and/or TFO\_BTS) to resolve Codec Mismatch, see Figure C.3-1.

Under normal circumstances (exceptions occur during time alignments or octet slips) all TFO I/O Processes are triggered every 160 samples or every speech frame of 20 ms. All events and actions are quantized in time into these smallest intervals.

It can be assumed that the processing times for the TFO Processes are very short and negligible. However, it must be ensured that no timing ambiguity occurs between the Processes.

This means the processing and exchange of information between them do not overlap in time. Care must be taken especially when time alignment occurs, which may be independent in uplink and downlink.

During these time alignments the TFO Frames or TFO Messages may shift in time and consequently the triggering point for the related TFO Processes changes, too.

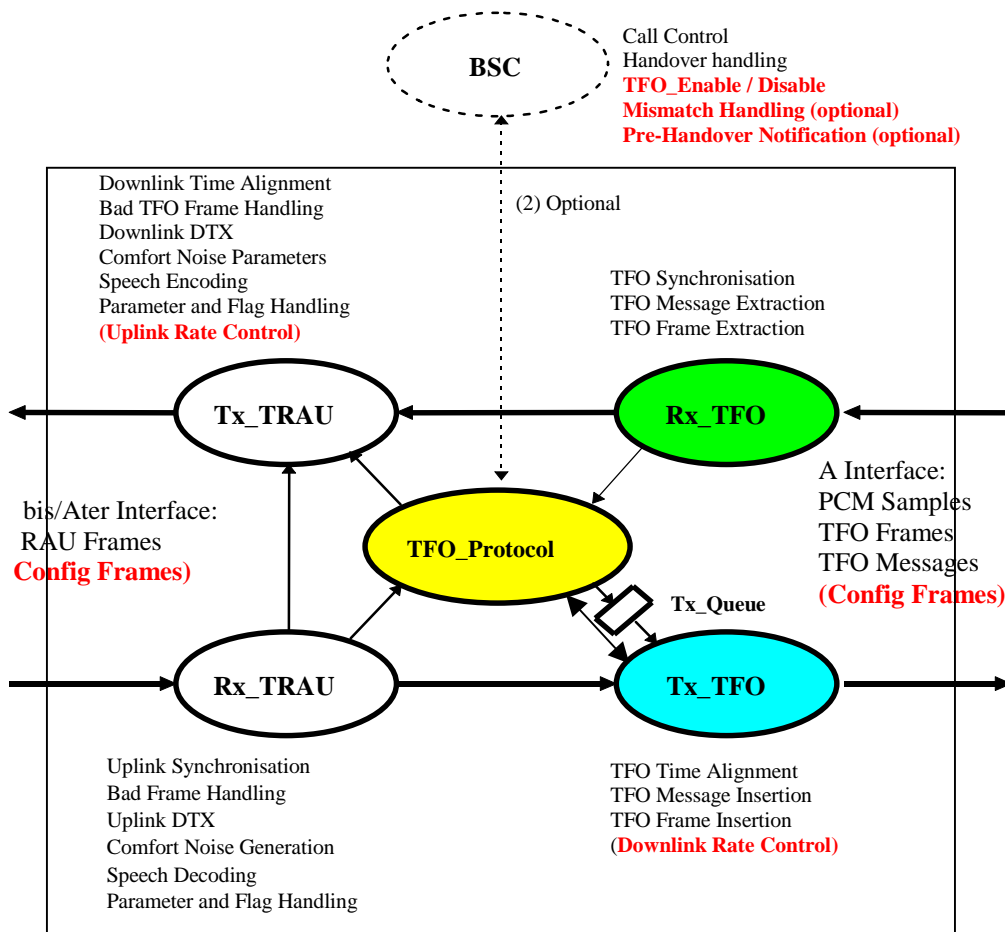


Figure C.3-1: The five TFO\_TRAU Processes

### C.3.1 Rx\_TRAU Process

The Rx\_TRAU Process receives Uplink TRAU Frames from the Abis/Ater Interface and synchronises to them, i.e. checks correct synchronisation and contents. It performs all actions of a conventional Uplink TRAU (see 3GPP TS 48.060 [3] and 3GPP TS 48.061 [4]). It extracts the data bits and calls, if appropriate, the Bad Frame Handler, the Uplink DTX functions and Comfort Noise Generator and finally the Speech Decoder. In case of AMR-WB a successive downsampling is performed before G.711 encoding.

The resulting speech samples are handled to the Tx\_TFO Process for output to the A interface. In addition Rx\_TRAU passes the Uplink TRAU Frames directly and unaltered to Tx\_TFO.

It further extracts the control bits and commands from the Uplink TRAU Frames and sends corresponding Rx\_TRAU Messages to the Tx\_TRAU Process (see 3GPP TS 48.060 [3] and 3GPP TS 48.061 [4]) and the TFO\_Protocol Process (see clause C.3.5).

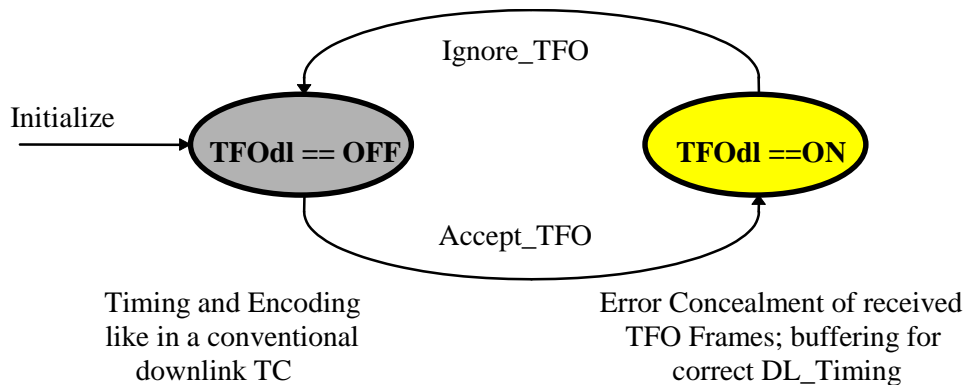
In case of the AMR new Configuration parameters may be received via Config frames. They are always directly passed to Tx\_TFO, although they are only sent in TFOul == ON (see Tx\_TFO) to the distant TFO partner. The Configuration parameters are also sent to TFO\_Protocol and Tx\_TRAU.

### 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 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.

**In case of AMR or AMR-WB** Tx\_TRAU furthermore modifies the CMI/CMR phase alignment when requested by TFO\_BTS via the Rx\_TRAU. The Tx\_TRAU sends on command by TFO\_Protocol the Distant or Optimal TFO configuration parameters by a Config Frame downlink to the BTS. This Tx\_TRAU indicates in addition by TFO\_Soon that TFO will be established soon, or by TFO\_Off that a mismatch has been detected by the TRAU and TFO has been terminated.

**During TFOdl == ON**, in case of the GSM\_FR, GSM\_EFR and GSM\_HR Codec\_Types, the Tx\_TRAU performs Bad Frame Handling and Comfort Noise Parameter Handling on parameter level on the received TFO Frames, if necessary. The resulting speech parameters and control bits are buffered until they are passed as Downlink TRAU Frames in correct phase position to the BTS.

**During TFOdl == ON**, in case of the AMR or AMR-WB Codec\_Types, no Bad Frame Handling or Comfort Noise Parameter Handling are performed in the Tx\_TRAU. The speech parameters and control bits extracted from the TFO Frames are passed as Downlink TRAU Frames with least possible delay down to the BTS.

**In case of AMR or AMR-WB** the Tx\_TRAU sends on command by TFO\_Protocol the distant TFO configuration parameters and/or the Optimal Codec Type and Optimal Configuration via a Config Frames downlink to the BTS. Tx\_TRAU indicates in addition by TFO\_On that TFO is established.

In case of AMR or AMR-WB the transition from TFOdl == OFF to TFOdl == ON and vice versa causes in general a phase shift of the downlink TRAU frames. Tx\_TRAU shall in these cases always complete the transmission of the ongoing TRAU frame and shall then insert the necessary number (0 to 159) of "1" bits (TRAU\_8k) or "11" pairs (TRAU\_16k) or "1111" quadruples (TRAU\_32k) on the Abis/Ater interface before the next TRAU frame is sent.

### C.3.2.1 Downlink Speech Transmission and DTX handling if TFO is ON

There are four possible cases regarding DTX in a Mobile-to-Mobile communication, as reflected in table C.3.2.1-1.

**Table C.3.2.1-1: DTX configurations in Mobile-To-Mobile communications**

Case	Local TRAU: Downlink	Distant TRAU: Uplink
0	No-DTX	No-DTX
1	No-DTX	DTX
2	DTX	DTX
3	DTX	No-DTX

### C.3.2.1.1 GSM\_FR, GSM\_EFR and GSM\_HR cases

If neither Distant Uplink nor Local Downlink DTX are active (case 0 in Table C.3.2.1-1), the Tx\_TFO Process receives TFO Frames from the A Interface with SID == "0". It synchronises to them, i.e. checks correct synchronization and content. It extracts the data bits and calls, if appropriate (e.g. if BFI == "1" or if the TFO Frame is not-valid, see clause C.6.2), a Bad Frame Handler to derive suitable parameters for Downlink TRAU Frames. This Bad Frame Handler on parameter level is subject to manufacturer dependent future improvements and is not part of this recommendation.

If Distant Uplink DTX is active, but not Local Downlink DTX (case 1 in Table C.3.2.1-1), then the Tx\_TFO Process receives TFO Frames containing speech parameters (SID == "0": handling as in case 0, see above), but also TFO Frames containing SID parameters (SID == "1" or "2") and TFO Frames marked with BFI == "1" during speech inactivity. Tx\_TFO then calls a Comfort Noise Generator to derive suitable speech parameters for Downlink TRAU Frames. The SP flag shall always be set to SP = "1". The Downlink TRAU Frames shall not contain the SID codeword, but parameters that allow a direct decoding. Also this Comfort Noise Generator on parameter level is subject to manufacturer dependent future improvements and is not part of this recommendation.

If Distant Uplink DTX and Local Downlink DTX are active (case 2 in Table C.3.2.1-1), then the Tx\_TFO Process receives TFO Frames containing either Speech parameters (SID == "0", handling see clause C.7.1) or SID parameters (SID == "1" or "2") or TFO Frames marked with BFI == "1" during speech inactivity due to transmission errors.

If a TFO Frame marked as a valid SID frame (SID == "2", BFI == "0") is received, it shall be stored in Tx\_TRAU and its parameters shall be sent directly as Downlink TRAU SID Frame with correct timing. The DL\_TRAU SID Frames produced from the valid stored frame are output repeatedly to the Abis/Ater interface whilst invalid SID frames (SID == "1") or frames marked as bad (BFI == "1") are received. These Downlink TRAU SID Frames shall be marked with the SP flag = "0" and shall all contain the SID codeword.

The stored SID Frame shall be considered as being valid for SID frame generation purposes until the receipt of the second instance of TAF == "1" (in a TFO Frame) following its initial storage. On expiry of the stored SID frame a suitable Bad Frame Handler for SID Frames shall be invoked to mute the Comfort Noise. Also, this Bad Frame Handler for SID Frames on parameter level is subject to manufacturer dependent future improvements and is not part of this recommendation.

If distant Uplink DTX is not active, but local downlink DTX is on (case 3 in Table C.3.2.1-1), i.e. only TFO Frames containing speech parameters are received, then one of the following alternative methods shall be used. The implementation of any of these alternatives is manufacturer dependent.

Alternative 1: The speech Frames are passed as DL\_TRAU Frames to the BTS. This is virtually identical to case 0 in Table C.3.2.1-1, with no speech pauses detected, and handled like described above.

Alternative 2: A voice activity detector makes the decision as to whether the frame contains speech or not based on the PCM samples received from the A interface. During periods decided as "Active Speech" the speech Frames are passed as DL TRAU Frames to the BTS as described above. During periods of "Speech Pause" Comfort Noise Parameters are calculated. These operations in alternative 2 are manufacturer dependent and not detailed here.

Alternative 3: The received Speech Frames may be decoded and the resulting PCM samples used for normal downlink VAD and DTX functions.

### C.3.2.1.2 AMR and AMR-WB cases

The Tx\_TRAU receives TFO Frames from the Rx\_TFO and converts them in DL TRAU frames. No Error concealment and Comfort Noise Generation is performed by the Tx\_TRAU. This is instead handled within the BTS and the Mobile Station. Since some of the control bits may change from TFO to TRAU frames it might be necessary to re-compute the relevant CRCs.

## C.3.2.2 Synchronisation and Bit Errors in Received TFO Frames

### C.3.2.2.1 GSM\_FR, GSM\_EFR and GSM\_HR cases

If Rx\_TFO detects an error in the received TFO Frame synchronization or control bits or if a CRC error is detected, and the error is detected *prior* to beginning the output of the same frame (as a Downlink TRAU Frame), then Tx\_TRAU shall either substitute parameters from the last good TFO Frame, or shall encode the received PCM samples for the duration of this frame.

If Rx\_TFO detects an error in the received TFO Frame synchronization or control bits or if a CRC error is detected, and the error is detected *after* beginning of the output of the same frame (as a Downlink TRAU Frame), then Tx\_TRAU shall deliberately corrupt the remaining, still unsent synchronization bits by setting them all to "0" and deliberately shall corrupt the remaining CRC bits. This will result in the BTS discarding this TRAU Frame, and transmitting a Layer 2 Fill frame or CRC-Inverted frame to the Mobile station (see 3GPP TS 48.060 and 3GPP TS 48.061). The effect of the frame error will subsequently be masked by the Mobile station's handling of bad frames.

### C.3.2.2.2 AMR and AMR-WB cases

#### C.3.2.2.2.1 No format conversion

When TFO and TRAU frames have the same format i.e. TFO\_16k and TRAU\_16k for FR\_AMR and FR\_AMR-WB or TFO\_32k and TRAU\_32k for FR\_AMR-WB or AMR\_TFO\_8+8k and AMR\_TRAU\_8+8k for HR\_AMR, then the received TFO frame shall be relayed as a DL TRAU frame toward the BTS. The Tx\_TRAU shall not perform any Error Correction.

#### C.3.2.2.2.2 With format conversion

If the BTS does not support the optional TRAU\_8+8k Frame Format, then TFO and TRAU frames may have different formats, e.g. AMR\_TFO\_8+8k and TRAU\_16k. Then the received TFO frame format is converted into a DL TRAU frame format toward the BTS. The Tx\_TRAU shall not perform any Error Correction, but rather relay the received parameters unaltered through. It might be necessary to re-compute the relevant CRCs.

If a CRC error is detected in the TFO Frame, the corresponding CRC, if any, shall be inverted in the DL TRAU frame. If there is no corresponding CRC, the remaining synchronization bits shall be inverted.

If a synchronization error is detected, the remaining synchronization bits shall be inverted in the DL TRAU frame as well.

## C.3.2.3 Maximum Rate Control

In case of the non\_AMR Codec Types (GSM\_FR, GSM\_HR, GSM\_EFR) no rate control is applied.

In case of AMR and AMR-WB Rate Control shall be performed for both directions. This Rate Control shall be independent of the TFO States in TRAU and BTS. In case the TFO\_Protocol alters the Max\_Rate parameter this shall be taken into account to the earliest possible point in time for all following frames in both directions. During the TFO negotiation the Max\_Rate can be set to the TFO Setup Mode. While in Tandem Free it can be set to Handover Mode before a handover occurs.

**TFO Setup Mode:** AMR mode to be used when switching to Tandem Free Operation. During the TFO negotiation the CACS to be used in TFO is determined (see clause 12). The corresponding TSM is derived in a similar way as the ICM (see [9]). Prior to switching to TFO the AMR modes are steered to the TSM.

**Handover Mode:** It is determined before the handover based on the new CACS after handover according to the rules for the new default ICM available in [9].

NOTE 1: It is recommended that the operator uses the default rule of ICM definition rather than setting it to an arbitrary value. Otherwise the Handover Mode won't be identical to the ICM of the new cell.

**Maximum Rate Control for the downlink direction:** Tx\_TRAU shall switch the AMR codec mode for the downlink direction (encoding) according to the UL CMC (Rate Control) received from the Rx\_TRAU and the local "Max\_Rate" parameter by taking the minimum of both.

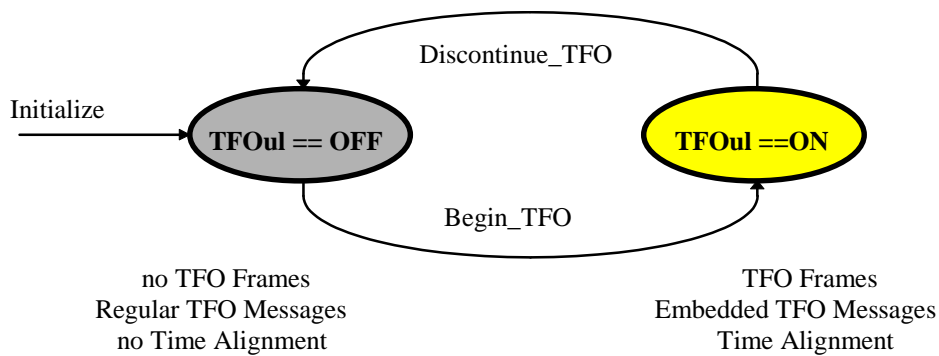
**Maximum Rate Control for the uplink direction:** Tx\_TRAU shall take the minimum of the local "Max\_Rate" parameter and the received Rate Control parameter (CMR) from Rx\_TFO and shall send this result downlink to the BTS within the CMR field. If no CMR is received from Rx\_TFO, because TFO is not ongoing, then this CMR shall be assumed to be at maximum (7).

### C.3.3 Tx\_TFO Process

The Tx\_TFO Process gets directly the unaltered Uplink TRAU Frames (containing the speech parameters and the control bits) and the decoded speech PCM samples from Rx\_TRAU. It further gets internal messages (commands) from TFO\_Protocol via the Tx\_Queue or directly (Max\_Rate parameter).

Tx\_TFO has two major States: TFO == OFF (default at beginning) and TFO == ON, see Figure C.3.3-1.

Toggleing between these two States is commanded by TFO\_Protocol with Begin\_TFO (BT) and Discontinue\_TFO (DT).



**Figure C3.3-1: States of the Tx\_TFO Process**

During TFOtx == OFF, decoded speech PCM samples and regular TFO Messages (if any) are sent onto the A interface. Time Alignment takes place only once, just before the beginning of the first regular TFO Message.

During TFOtx == ON, TFO Frames and embedded TFO Messages (if any) are sent. Time Alignment takes place just before the first TFO Frame and then whenever required in between two TFO Frames.

The Tx\_TFO Process builds the relevant TFO Frames and sends them in the correct phase relation onto the A-Interface. Time alignment of TFO Messages and TFO Frames are handled autonomously and independent of the TFO\_Protocol Process. Rx\_TRAU informs Tx\_TFO about any changes in the phase position of the Uplink TRAU Frame and Tx\_TFO inserts automatically the correct number of T\_Bits before the next TFO Frame, and embeds autonomously the next TFO\_Message or a TFO\_FILL Message, if necessary.

The TFO\_Protocol Process can send internal messages into the **Tx\_Queue** (First In, First Out). Tx\_TFO shall take the message from the Tx\_Queue one by one, shall process them autonomously and shall send the resulting TFO Messages in correct order and phase position, as regular or as embedded TFO Messages. Tx\_TFO shall generate a Runout Message to TFO\_Protocol, if the last TFO\_Message is sent without guarantee of a direct continuation by another TFO Message, i.e. if the (possible) IPEs may have run out of synchronisation (see Appendix A). TFO\_Protocol may delete messages from Tx\_Queue, as long as they are in there:  
Command "Clear Tx\_Queue", at time  $T_c$ .

Basically, messages or commands that are already in processing by Tx\_TFO at  $T_c$  can not be stopped, deleted or interrupted. The TFO Protocol is designed to work properly with that default handling, although not with fastest processing.

But, Tx\_TFO shall investigate at  $T_c$ , how far the transmission of the current TFO Message has proceeded and shall "Modify on the Fly" this last TFO\_Message before  $T_c$  into the first one after  $T_c$ , see Figure C3.3-2.

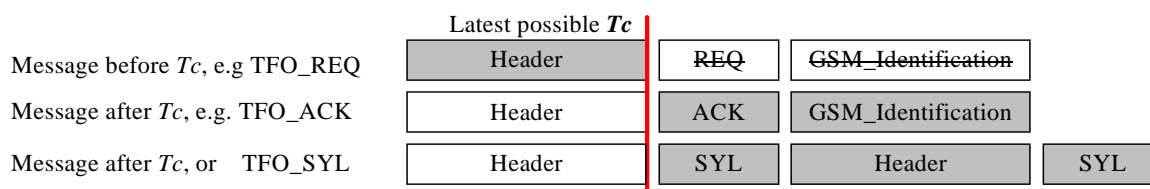


Figure C.3.3-2: Examples of Modification on the Fly within the Header Transmission

### C.3.3.1 Maximum Rate Control

In case of the non\_AMR Codec Types (GSM\_FR, GSM\_HR, GSM\_EFR) no rate control is applied.

**Maximum Rate Control for the downlink direction in TFO:** Tx\_TFO shall take the minimum of the local "Max\_Rate" parameter and the received Rate Control parameter (CMR) from the BTS via Rx\_TRAU and sends this minimum uplink to the distant TFO partner as CMR. This Rate Control is independent of the TFO State, but has only effect if TFO Frames are sent. In case the TFO\_Protocol alters the Max\_Rate parameter this shall be taken into account to the earliest possible point in time for all following frames.

## C.3.4 Rx\_TFO Process

The Rx\_TFO Process receives TFO Messages and TFO Frames from the A-Interface and synchronises to them, i.e. checks correct synchronisation and contents. It bypasses all PCM samples and TFO Frames directly to Tx\_TRAU for further processing. The Rx\_TFO Process further extracts all the control bits and TFO Messages and sends corresponding Rx\_TFO Messages to the TFO\_Protocol Process.

If Embedded messages are detected in the TFO frames, the altered synchronization bits may be reconstructed with '1' bits before passing them to Tx\_TRAU.

When the Rx\_TFO received distant TFO parameters, either by TFO Messages or TFO Frames (Config\_Prot Frames), it relays them to the TFO\_Protocol.

When the Rx\_TFO receives distant TFO parameters within Config\_Prot Frames, it passes them directly through to Tx\_TRAU and further on to the BTS.

### C.3.4.1 Search for and Monitoring of TFO Synchronization

The monitoring of TFO Frame or TFO Message synchronisation shall be a continuous process. Typically, TFO Messages and TFO Frames follow each other with a well-defined phase relation. Insertion of T\_Bits or octet slips may, however, disturb that regular phase relation every now and then and shall be taken into account. In all error cases, the receiver shall investigate, if sync has been lost due to octet slip, phase adjustment or other events and shall try to resynchronize as fast as possible.

Typically, TFO Frame synchronisation is similar or identical to TRAU Frame synchronisation, see 3GPP TS 48.060 [3] and 48.061.

During Tandem Free Operation, however, it is sometimes necessary to exchange TFO Messages by embedding them into the TFO Frame flow. This is explicitly indicated by a control bit (C5) for the 16 kbit/s TFO frame, and implicitly by the TFO frame itself for the GSM HR Codec Type. Some of the TFO Frame synchronization bits are then replaced by bits of the TFO Message. TFO Messages follow specific design rules, which can be used to check if synchronisation is still valid. For the 8 kbit/s AMR TFO frames the presence of an embedded TFO Message is not specifically indicated. The potential presence of an embedded TFO Message shall be checked every time a corrupted synchronization pattern is received.

The first TFO Message or the first TFO Frame (whatever comes first) shall be completely error free to be acceptable by Rx\_TFO. After that all "valid" (see clause 8.4.2) TFO Messages shall be reported to TFO\_Protocol with a respective message. If a TFO Message has been received before and synchronisation is not found again for more than 60 ms, i.e. no "present" or "valid" TFO Message can be found during that time, then Rx\_TFO shall generate a MSL (Message\_Sync\_Lost) Message to TFO\_Protocol. A "not-valid", but "present" TFO Message shall not be reported, but also no MSL shall be reported, i.e. synchronisation is regarded as not lost, but the TFO Message is ignored.

Similarly, the first five "valid" TFO Frames shall be reported to TFO\_Protocol with frame number n (n == 1,2, ..5). Further "valid" TFO Frames do not need to be reported.

Similar, if for the first time after the PCM\_Idle period, PCM\_Non\_Idle samples are received, then a PCM\_Non\_Idle Message shall be sent to TFO\_Protocol. Further PCM\_Non\_Idle samples need not be reported.

If TFO Frame Synchronization is lost, or if too many errors are detected in TFO Frames (no present TFO Frames), then the Rx\_TFO shall generate a FSL (Frame\_Sync\_Lost) Message to TFO\_Protocol with frame number n (n == 1,2,3), the number of lost TFO Frames since the last valid TFO Frame. No more than three FSL Messages need to be reported in a series.

NOTE: The MSL and FSL Messages shall not be mixed up with the TFO\_SYL Message, by which the distant TFO Partner reports lost synchronisation.

TFO Messages with Extension\_Blocks that can not be understood by the receiving TRAU, but which follow the design rules for IS\_Extension\_Blocks, shall be ignored. This guarantees future expandability.

### C.3.4.2 Errors in TFO Messages and TFO Frames

Some Definitions, which may serve as a guideline:

A TFO Message is called "error-free", if no error can be detected within the whole message.

A TFO Message is called "single-error", if no more than one bit position differs either in the IS\_Header or the IS\_Command\_Block or the GSM\_Ident\_Block or the IPE\_Mode\_Block or the Sync bits and no errors are detectable within the CRC fields or the EX-fields.

A TFO Message may be regarded as "correctable", if the phase position is the same as the preceding TFO Messages and

- no more than 2 bit positions differ in the IS\_Header; and
- no more than 1 error is detected within the IS\_Command\_Block; and
- no more than 3 errors are detected within the IPE\_Mode\_Block; and
- no more than 3 errors are detected within the GSM\_Ident\_Block; and
- no more than 1 error is detected within the Sync-Bit(s); and
- no more than 0 error is detected within the EX-field(s); and
- no more than 0 error is detected within the CRC-fields; and
- the total number of detected errors is not higher than 3.

TFO Message, which are error-free, single-error or correctable are also called "valid" TFO Messages. All other TFO Messages are called "not-valid".

A TFO Message may be regarded as "present", if the phase position is the same as the preceding TFO Messages and

- no more than 4 bit positions differ in the IS\_Header; and
- no more than 2 errors are detected within the IS\_Command\_Block; and
- no more than 3 errors are detected within the IPE\_Mode\_Block; and
- no more than 3 errors are detected within the GSM\_Ident\_Block; and
- no more than 2 errors are detected within the Sync-Bit(s); and
- no more than 1 error is detected within the EX-field(s); and
- no more than 1 error is detected within the CRC-fields; and
- the total number of detected errors is not higher than 5.



Sequences, which are not "*valid*" or "*present*" shall not be recognized as TFO Messages at all ("*not-present*").

Note that the insertion of T\_Bits may change the phase position of the TFO Frames and of bits of an embedded TFO Message. The TFO Message shall in that case be classified after the removal of the T\_Bits.

An octet slip may also change the phase position of bits within a regular or embedded TFO Message.

If an error-free or a single-error TFO Message can be found after considering a hypothetical octet slip ( $\pm 1$  sample), then it may be regarded as error-free or single-error and the new phase position shall be regarded as valid, if no valid or present TFO Message can be found at the old phase position.

A TFO Frame is called "*error-free*", if no error can be detected within the whole frame.

A TFO Frame is called "*single-error*", if no more than one bit position differs either in the synchronisation bits or the T\_Bits and if no other errors can be detected. TFO Frames, which are error-free, or single-error are also called "*valid*" TFO Frames. All other TFO Frames are called "*not-valid*".

A TFO Frame may be regarded as "*present*", if

- no more than 4 bit positions differ in the synchronisation bits
- no more than 2 errors are detected within the T\_Bits;
- no more than 1 error is detected within the control bits;
- no more than 1 error is detected within the CRC block; and
- the total number of detected errors is not higher than 5.

Bit sequences, which are not "*valid*" or "*present*" shall not be recognized as TFO Frames at all ("*not-present*").

Note that the insertion or deletion of T\_Bits may change the phase position of the TFO Frames. The TFO Frame shall in that case be classified after considering the T\_Bits.

An octet slip may also change the phase position of bits within a TFO Frame. Typically a TFO Frame can not be corrected after an octet slip, but the next TFO Frame shall be found again.

The speech data bits of a valid TFO Frame shall be regarded as "*bad*", if the BFI flag is set to  $BFI == "1"$ . In that case Bad Frame Handling shall be performed for the GSM\_FR, GSM\_HR and GSM\_EFR speech Codec Types. For AMR and AMR-WB, all frames are passed unchanged to the Tx-TRAU. Similar definitions hold for other valid TFO Frames, equivalent to Uplink TRAU Frames, e.g. Invalid SID... (see 3GPP TS 48.060 and 48.061).

### C.3.5 TFO\_Protocol Process

The TFO\_Protocol Process is typically invoked whenever a message is received, either from Rx\_TRAU, Rx\_TFO, Tx\_TFO or the local BSC.

Two key events are due to modifications of the local configuration,

- a modification of the used speech Codec Type (New\_Local\_Codec);
- or its Configuration Parameters (e.g. the ACS in case of AMR or AMR-WB) (New\_Local\_Config); and
- a modification of the list of the alternative speech Codec Types (New\_Local\_Codec\_List);
- TFO Enable or TFO Disable;
- Handover Soon.

The New\_Local\_Codec is extracted from the uplink TRAU Frames and reported by Rx\_TRAU.

The other parameters are received from the BSC, via the BTS in Config Frames (AMR and AMR-WB cases only) or in an manufacturer dependent way.

### C.3.5.1 Messages from Rx\_TRAU or local BSS

Rx == New_Speech_Call ();	Rx_TRAU is activated by BTS (several TRAU Frames).
Rx == New_Local_Codec ();	In Call Modification to other Codec Type (several TRAU Frames).
Rx == New_Local_Config ();	In call modification (e.g. new ACS, in Config Frame)
Rx == Data_Call;	Received from Rx_TRAU: In Call Modification to Data_Call.
Rx == Local_Codec_List;	Manufacturer dependent
Rx == TRAU_Idle;	Manufacturer dependent, either from Rx_TRAU or BSC.
Rx == TFO_Enable;	Received from Rx_TRAU for AMR or AMR-WB: Enable the TFO process Optionally received from the BSC for GSM_FR, GSM_HR and GSM_EFR.
Rx == TFO_Disable;	Received from Rx_TRAU for AMR or AMR-WB: Disable the TFO process Optionally received from the BSC for GSM_FR, GSM_HR and GSM_EFR.
Rx == TFO_Soon;	The sent TFO_Soon is acknowledged by the BTS, especially important and handled as RC_Ack in WAIT_RC State.
Rx == Handover_Soon ();	Optional Pre-Handover warning (e.g. in Config_Frame)

### C.3.5.2 Messages to Tx\_TRAU

Tx_TRAU := Accept_TFO;	If TFO Frames are correctly received, they shall be used. Rate Control in Tx_TRAU shall take the distant side into account.
Tx_TRAU := Ignore_TFO;	TFO Frames shall be ignored in general. Rate Control in Tx_TRAU shall ignore the distant side..
Tx_TRAU := Set_Max_Rate ();	The Rate Control shall be limited to the give maximum rate, e.g. TFO Setup Mode, Handover Mode, Maximum mode of the Common ACS. The new Max_Rate value shall be taken into account in the next possible frames.
Tx_TRAU := Config_Frame ();	A Dis_Req frame with all available distant TFO parameters is sent to the BTS (The BTS acknowledges this by UL_Ack).
Tx_TRAU := TFO_Soon;	TFO_Soon is sent to the BTS (The BTS stops Time alignment and acknowledges with TFO_Soon => RC_ACK).
Tx_TRAU := TFO_On;	TFO_On is sent to the BTS (The BTS may perform round trip delay measurements; the BSC should not alter the configuration during handover).
Tx_TRAU := TFO_Off;	TFO_Off is sent to the BTS after no more TFO Frames are received and the normal Tx_TRAU operation has been resumed. The BTS shall resume normal operation, too.

### C.3.5.3 Optional Messages to the local BSC

Tx\_BSC := TFO (Distant\_Used\_Codec, Distant\_Codec\_List, Distant\_Configuration, Optimal Codec Type and Configuration, ...).

For the AMR, AMR-WB, GSM\_FR, GSM\_HR and GSM\_EFR Codec Types these parameters may be transmitted on a proprietary interface to the BSC to allow the BSC to perform the optional Codec Type and Codec Configuration Mismatch resolution and Optimisation.

In case of AMR and AMR-WB these configuration parameters are transferred in Config\_Prot Frames or on a proprietary interface to the BSC to allow the BSC to perform the optional Codec Type and Codec Configuration Mismatch resolution and Optimisation.

### C.3.5.4 Messages to Tx\_TFO

The symbol () indicates that these Messages contain parameters, see Clause 8.

Tx := TFO_REQ ();	main TFO_REQ Message.
Tx := TFO_ACK ();	main TFO_ACK Message, response only to TFO_REQ.
Tx := TFO_REQ_L ();	used in Mismatch, Operation and Periodic_Retry to inform about alternative Codecs.
Tx := TFO_ACK_L ();	response only to TFO_REQ_L.
Tx := TFO_TRANS ();	commands IPEs to go transparent.
Tx := TFO_NORMAL;	resets IPEs into their normal operation.
Tx := TFO_FILL;	mainly to pre-synchronise IPEs.
Tx := TFO_DUP;	"I receive TFO Frames in Establishment".
Tx := TFO_SYL;	"I lost TFO Frame synchronisation".
Tx := Begin_TFO;	Insert TFO Frames from now on.
Tx := Discontinue_TFO;	Discontinue inserting TFO Frames.
Tx_TFO := Set_Max_Rate ();	The Rate Control shall be limited to the given maximum rate, e.g. Handover Mode, Maximum mode of the Common ACS. The new Max_Rate value shall be taken into account in the next possible frames.
Clear Tx_Queue;	Clears all remaining commands from Tx_Queue.
Rx == Runout;	Reports that the continuous stream of outgoing TFO Messages may be interrupted (from Tx_TFO).
Tx_TFO := Con_Req();	Send a Con_Req config frame.
Tx_TFO := Con_Ack();	Send a Con_Ack config frame.

### C.3.5.5 Messages from Rx\_TFO

The symbol () indicates that these Messages contain parameters, see Clause 8.

Rx == TFO_REQ ();	
Rx == TFO_ACK ();	
Rx == TFO_REQ_L ();	
Rx == TFO_ACK_L ();	
Rx == TFO_TRANS ();	may serve as alternative TFO_ACK in some cases!
Rx == TFO_NORMAL;	
Rx == TFO_FILL;	
Rx == TFO_DUP;	
Rx == TFO_SYL;	
Rx == TFO_Frame ();	TFO_Frame (Distant_Used_Codec; Number_of_Received_Frames).
Rx == Distant_Config();	
Rx == Frame_Sync_Lost ();	Frame_Sync_Lost (Number_of_Lost_Frames).

Rx == Mess\_Sync\_Lost;      Message\_Sync\_Lost.

Rx == PCM\_Non\_Idle;      at the beginning of a period with several samples/frame different from PCM\_Idle.

The message "TFO\_Frame ()" needs to be sent only at the first five occurrences, either after a not valid TFO Frame, or if the Distant\_Used\_Codec changed.

The message "Frame\_Sync\_Lost ()" needs to be sent only at the first five occurrences of errors in TFO Frames or loss of synchronisation, after a correctly received TFO Frame.

The message "Mess\_Sync\_Lost" is sent, when after a valid TFO Message no following TFO Message is found.

---

## C.4 TFO\_BTS

The following clauses apply only when an AMR or AMR-WB Codec Type is the Used\_Codec\_Type and when TFO is enabled.

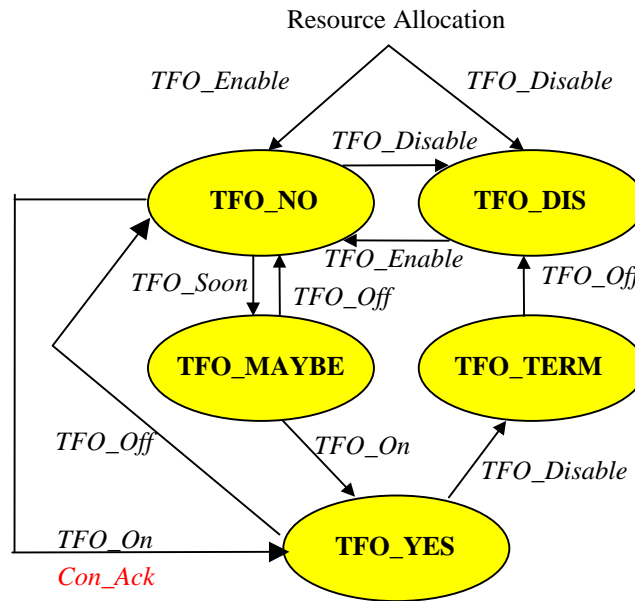
### C.4.1 TFO\_States and Transitions

The BTS needs to know the status of the TFO connection for best operation of the AMR Link adaptation and Optimal Handover procedure.

The TFO\_BTS state machine is made of five states:

- **TFO\_DIS:**                      No Tandem Free Operation is allowed or ongoing;
- **TFO\_NO:**                      Tandem Free Operation is enabled, but is neither ongoing nor under establishment;
- **TFO\_MAYBE:**                  Tandem Free Operation is under establishment, but is still not ongoing;
- **TFO\_YES:**                      Tandem Free Operation is ongoing.
- **TFO\_TERM:**                    Tandem Free Operation is still ongoing, but will terminate soon.

The following TFO\_State diagram (Figure C.4.1-1) shows the five States and the most important transitions.



**Figure C.4.1-1: Main TFO\_State Diagram within the BTS**

At resource allocation the BTS enters either **TFO\_DIS** or **TFO\_NO**, depending on the Configuration Message from the BSC (see clause C.5.2.1). The transition from one state to another one is triggered by the reception of a message, either from the BSC or the TRAU. According to the TFO\_State the BTS shall initiate different actions.

In **TFO\_DIS** and **TFO\_NO** the BTS may perform Time and Phase Alignment. In all other States (**TFO\_MAYBE**, **TFO\_YES**, **TFO\_TERM** which are often gathered under the expression "TFO ongoing", the BTS should not send Time or Phase Alignment Messages to the TRAU, since the TRAU shall not obey them. In State **TFO\_YES** the BTS may perform Phase Alignment on the air interface, see 3GPP TS 45.009 [9].

*TFO\_Enable* and *TFO\_Disable* are not messages per se, but are included in Configuration Message from the BSC (see clause C.5.1) by setting or resetting the TFO\_Enable bit. In any case the local configuration parameters shall be sent to the TRAU immediately.

*TFO\_Soon*, *TFO\_On* and *TFO\_Off* are sent from the TRAU, either with or without configuration parameters and rate Control commands from the distant side.

*TFO\_Enable* at resource allocation brings the BTS into **TFO\_NO**. *TFO\_Enable* is relayed to the TRAU by the BTS (TFOE bit in TRAU frames). The TRAU shall then start TFO\_Negotiation with a potential TFO\_Partner.

*TFO\_Enable* in State **TFO\_DISABLED** or **TFO\_TERMINATING** starts the same procedure and brings the BTS also into State **TFO\_NO**. In any other State the *TFO\_Enable* has no effect on the ongoing procedures.

*TFO\_Disable* at resource allocation brings the BTS into **TFO\_DISABLED**. The TRAU shall not initiate nor respond to any TFO\_Negotiation. It shall terminate TFO operation or Negotiation.

*TFO\_Disable* in **TFO\_YES** brings the BTS into State **TFO\_TERMINATING**. *TFO\_Disable* in any other State brings the BTS immediately into **TFO\_DISABLED**.

If TFO is enabled the TRAU will get the knowledge about the distant side by the first received TFO\_REQ or TFO\_ACK Message or by Con\_Req or Con\_Ack Messages. As soon as the TRAU gets knowledge that a TFO\_Partner exists, it informs the BTS in downlink about the Distant configuration, see clauses C.6.1 and C.6.2). If TFO is possible, the TRAU sends a *TFO\_Soon* Message to the BTS. If TFO is not possible, the BSS may then perform Mismatch Handling. Alternatively the TRAU sends only the Optimal Codec Type and Optimal Codec Configuration to the BTS and/or further to the BSC.

*TFO\_Soon* in State **TFO\_NO** brings the BTS into State **TFO\_MAYBE**. The BTS has to discontinue Time and Phase Alignment with the TRAU and instead has to buffer the received TRAU frames for downlink transmission.

*TFO\_On* reports that finally TFO is ongoing, i.e. TFO Frames are exchanged in both directions. The BTS enters State **TFO\_YES** and enables the AMR Adaptation, now considering both radio legs for the selection of the optimal Codec\_Mode. In TFO handover situations a Con\_Ack instead of a TFO\_On will bring the BTS into State **TFO\_YES**.

*TFO\_Off* brings the BTS immediately into the State **TFO\_NO**. The BSC should be informed.

## C.4.2 Handling of downlink DTX in TFO

If TFO is ongoing and the BTS receives downlink TRAU frames classified with "SID\_First" or "SID\_Update", it shall use one of the following options:

- Option 1) The BTS performs normal DTX operation in downlink if DTX DL is enabled.
- Option 2) The BTS shall send the SID\_First, SID\_Update frames as in normal DTX, but shall send SID\_Filler frames between SID frames when DTX DL is disabled.

See 3GPP TS 26.093 for the definition of the SID\_Filler frames.

Note : In all cases ONSET frames may be ignored, see 3GPP TS 45.009 [9], but may be used to ensure proper synchronisation.

## C.4.3 Handling of Errors in Configuration Parameters

The BTS shall check the consistency of the configuration data sent by the TRAU. If inconsistent they shall be ignored, i.e. no report is made to the BSC, no change of the MS-BTS ACS is attempted, no acknowledgement is sent back to the TRAU. The missing Acknowledgement will trigger a repetition of the configuration data.

## C.4.4 Procedures for Round Trip Delay Measurements

In case of AMR and AMR-WB, the link adaptation may need information on the round trip delay between the local BTS and the local TRAU or - when TFO is ongoing - with the distant BTS. Therefore, the BTS shall count the number of elapsed TRAU frames between the sending of a "Con\_Req" (see clause C.6.2) message and the receipt of the corresponding acknowledgement. This number, multiplied by 20 ms, gives an estimate of the round trip delay between the BTS and its partner. The type of acknowledgement (DL\_Ack or Con\_Ack) indicates the type of partner, i.e. whether the local TRAU or the distant BTS has answered.

This procedure may be repeated whenever the status of the connection changes. The round trip delay measurement is triggered by the transition into State TFO\_YES. But there are other cases, where a new delay measurement is required, although the State TFO\_YES has not changed. This is e.g. the case when a distant handover occurred. The BTS where the handover takes place shall send the "Handover\_Complete" Notification within the Time Alignment field of a Con\_Req frame to the other BTS. This then shall repeat the Delay Measurement on its side.

The Handover\_Complete Notification shall be re-sent in every Con\_Req frame until a Con\_Ack was received.

The BTS may report the round trip delay measurement result to the BSC by sending a round Trip Delay Report (see 3GPP TS 48.058). Any substantial change (more than 60 ms difference) in the round trip delay may be reported, too.

---

## C.5 TFO\_BSC

The role of the BSC in TFO depends on the speech Codec Type in use, and on the degree of flexibility desired.

For the GSM\_FR, GSM\_EFR and GSM\_HR speech Codec Types the BSC may perform the Resolution of Codec Type Mismatch and Codec Type Optimization (see clause C.5.1).

For the AMR and AMR-WB Speech Codec Types the role of the BSC can be much more important (see clause C.5.2).

### C.5.1 Resolution of Codec Type Mismatch and Codec Type Optimization

The BSC is in charge of solving the Codec Type Mismatches. The BSC receives from the TRAU or the BTS in case of AMR and AMR-WB the distant speech service configuration (e.g. the distant used codec and the distant codec list), or alternatively, the Optimal Codec Type and the Optimal Codec Configuration.

The BSC transmits to the TRAU the Local configuration, via the BTS for AMR and AMR-WB. It may have to refresh this information if the configuration changes along the time.

The BSC may implement the TFO decision algorithm provided in the Clause 11 and 12, or alternatively get the results from the BTS or TRAU. This TFO decision algorithm ensures that both BSCs obtain the same result. The BSC can then initiate an intra-cell Handover if a different Codec Type is required to ensure Tandem Free Operation.

## C.5.2 Role of the BSC for AMR and AMR-WB TFO

AMR introduces a degree of complexity, due to its multi-rate nature, to its link adaptation and to the different options it allows. It is required that the AMR configurations of the two terminals and two BSS be aligned.

The ACS can vary and depends on the BTS generation, BTS manufacturer or on Operators' preferences. The ACS can be tailored to cope with the environment of a given cell, e.g. a dense urban area or a flat rural area.

The MS may either support FR\_AMR only or FR\_AMR and HR\_AMR and FR\_AMR-WB. The BSS can support from one mode to all fourteen AMR modes (8 in FR\_AMR and 6 in HR\_AMR) and between one and seven modes for AMR-WB. The ACS in GSM may include between 1 and 4 modes for AMR and AMR-WB.

In addition to resolving the Codec Type Mismatch as explained in clause C.5.1, the BSC can also be involved in the following TFO related tasks:

1. Determination and Establishment of the Optimal ACS.
2. Keep as far as possible the same ACS during Handovers.

### C.5.2.1 Configuration of the AMR and AMR-WB speech service.

The MS is configured by the BSC at Call set-up and during handovers through Layer 3 signalling (see GSM 04.18 [14]). The BTS is configured through the CHANnel ACTIVation message (see 3GPP TS 48.058). The TRAU circuit pools are managed by the MSC on request of the BSC (see 3GPP TS 48.008 [10]).

The AMR configuration of the MS and BTS can be changed during the call by:

- Intra-Cell Handover (see 3GPP TS 44.018 and 3GPP TS 48.058 [12]),
- Mode-Modify (see 3GPP TS 44.018 and 3GPP TS 48.058 [12]),
- RATSCCH (see 3GPP TS 45.009 [9] and 3GPP TS 48.058 [12]).

These procedures are initiated by the BSC. The RATSCCH can in addition be delegated to the BTS by the BSC at the Channel Activation. This can modify the way TRAU handles TFO setup. (see clause C.5.2.2)

The RATSCCH is the most efficient technique from a speech quality point of view since it can be faster and can minimize the number of lost frames.

The Intra-Cell Handover is a synchronized handover and creates less speech frame losses than the typical Handovers.

The Mode Modify offers the advantage of keeping the same radio resource but can introduce long speech blanks.

### C.5.2.2 Determination and Establishment of the Common ACS

The resolution of the AMR Codec Configuration Mismatch is based on similar principles as the Codec Type Mismatch. The corresponding TFO Decision algorithm is defined in Clause 12. When applied, it leads to a common optimal ACS at both ends of the TFO connection.

The resolution of Codec Configuration Mismatch depends on the Optimisation Mode, see table C.5.2.2-1.

**Table C.5.2.2-1: Coding of the Optimisation Mode (OM)**

OM Code	Optimisation Mode	Comment
0	No Change	Change of the ACS is not supported
1	Change	Change of the ACS is supported

The reporting of the Configuration parameters from the TRAU to the local BTS depends on the "Optimal or Distant Configuration (OD)" parameter, see table C.5.2.2-2.

**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.

The configuration is changed using one of the methods listed in the clause C.5.2.1.

### 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.

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).

---

## 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 TRAU's 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 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 TRAU's or BTSs in case of 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 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 in 3GPP 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 bits see 3GPP TS 45.009 [9]):

The ACS 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.

#### **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.. The coding is according to 3GPP TS 26.103.

#### **Maximum Number of Modes in the ACS** (MACS: 3 bits)

The MACS 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.

#### **AMR TFO Version Number** (ATVN: 1 bit)

The current AMR TFO Version Number is 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.

#### **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 <sup>(5)</sup>	2	D65..D66	D65..D66	# (= 0.0)		D1..D2	D1..D2	D1..D2
OD	1	D67	D67	#		D3	D3	D3
OM <sup>(3)</sup>	1	D68	D68	#		D4	D4	D4
ACS <sup>(3)</sup> (Optimal ACS) <sup>(5)</sup>	8	D69..D76	D69..D76	#		D5..D12	D5..D12	D5..D12
SCS <sup>(3)</sup>	8	D77..D84	D77..D84	#		D13..D20	D13..D20	D13..D20
ATVN <sup>(3)</sup> , short <sup>(6)</sup>	1	D85	D85	#		D21	D21	# (= 0)
Sys_ID, short <sup>(6)</sup>	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)	# <sup>(1)</sup>
ACT <sup>(3)</sup> (Optimal ACT) <sup>(5)</sup>	4	D96..D99	D96..D99	#		D234..D237	D234..D237	D234..D237
MACS <sup>(3)</sup>	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)	# <sup>(2)</sup>
SCS_2 <sup>(4)</sup>	8	D17..D24	# (= 1..1) <sup>(7)</sup>	#		D203..D210	D203..D210	# (= 1..1) <sup>(7)</sup>
OM_2 <sup>(4)</sup>	1	D25	# (= 0)	#		D211	D211	# (= 0)
MACS_2 <sup>(4)</sup>	3	D26..D28	# (= 1.0.0)	#		D212..D214	D212..D214	# (= 1.0.0)
ATVN_2 <sup>(4)(6)</sup>	1	D29	# (= 0)	#		D215	D215	# (= 0)
SCS_3 <sup>(4)</sup>	8	D30..D37	# (= 1..1) <sup>(7)</sup>	#		D216..D223	D216..D223	# (= 1..1) <sup>(7)</sup>
OM_3 <sup>(4)</sup>	1	D38	# (= 0)	#		D224	D224	# (= 0)
MACS_3 <sup>(4)</sup>	3	D39..D41	# (= 1.0.0)	#		D225..D227	D225..D227	# (= 1.0.0)
ATVN_3 <sup>(4)(6)</sup>	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<sup>(7)</sup>. 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) <sup>(8)</sup>	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 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 mechanism 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 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 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).

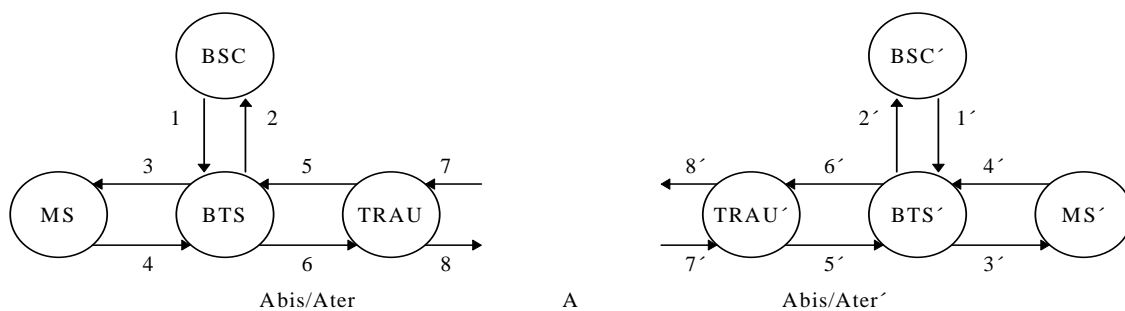
The BTS should report the Round trip delay it has estimated (Round Trip Delay REPORT). It should report it every time a significant change (e.g. 60 ms) is detected in the round trip delay (see clause 8.2.4).

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 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 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 4<sup>th</sup> 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 TRAU in 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. TCMs) 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.



---

## C.9 Location of the TFO Decision Algorithm

The TFO Decision Algorithm as described in clause 11 and 12 shall always be located within the TRAU. Optionally it may in addition be located in the BTS (for Codec Configuration Optimisation) and the BSC (for Codec Type and Codec Configuration Optimisation).

### C.9.1 Immediate TFO Set-up

The TFO Decision Algorithm shall always be within the TRAU. This is important and sufficient for Immediate TFO\_Setup. It might be available also within the BTS, but that is not essential.

The TRAU shall inform the BTS with TFO\_Soon, that Immediate TFO is possible (TFO\_BTS into TFO\_MAYBE).

The TRAU shall inform the BTS with CMR  $\leq$  RC<sub>i</sub> about the allowed Rate Control.

The TRAU may send a Dis\_Req to the BTS with the available distant configuration parameters, or, alternatively, with the Optimal Configuration Parameters.

Important is that the BTS shall acknowledge the TFO\_Soon with TFO\_Soon.

The TRAU shall wait in State WAIT\_RC until the BTS has acknowledged. Then it shall start to send TFO\_TRANS and TFO Frames.

When informed with TFO\_Soon that Immediate TFO Setup is ongoing, the BTS shall not change the ACS on the air interface, but wait at least until in State TFO\_YES.

The BTS shall restrict the rate adaptation within the limits given by the TRAU within the downlink CMR.

The TRAU shall release the rate control when in state "Operation" to the rates within the common ACS.

### C.9.2 Codec Configuration Optimisation

The TFO Decision Algorithm shall always be within the TRAU. The TRAU shall inform the BTS either about the distant Codec Configuration or, alternatively the optimal Codec Configuration (defined by the OD parameter).

In the first case the BTS shall also run the TFO Decision Algorithm (again) to determine the optimal Configuration.

In the second case the TFO decision Algorithm is not needed within the BTS.

If authorised so by the BSC the BTS shall perform Codec Configuration Modification by RATSCCH in State TFO\_NO (for Mismatch Resolution) or in State TFO\_YES (for Optimisation). The BTS shall inform the BSC hereafter.

If not authorised by the BSC, or if the Codec Type has to be modified in addition, the BTS shall not perform any modifications, but only inform the BSC.

### C.9.3 Codec Type Optimisation

The TFO Decision Algorithm shall always be within the TRAU. The TRAU shall inform the BTS either about the distant Codec Configuration or, alternatively the optimal Codec Configuration (defined by the OD parameter).

In the first case the BTS shall also run the TFO Decision Algorithm (again) to determine the optimal Configuration.

In the second case the TFO decision Algorithm is not needed within the BTS.

If the Codec Type has to be modified, the BTS shall not perform any modifications, but only inform the BSC, either by sending the distant Configuration or, alternatively the optimal Configuration.

In the first case the BSC has to run the TFO Decision Algorithm (again), in the second case the TFO Decision Algorithm is not needed within the BSC.

The BSC shall perform a necessary Codec Type Modification or Codec Configuration Modification, when it had set the Configuration parameters accordingly (Codec\_List contains more than the Active Codec Type, the Optimisation\_Mode is set to "Change").

---

## Annex D (normative): Tandem Free Operation in 3G including AMR-WB

### D.1 Scope

This Annex D describes the mandatory and optional actions within the Transcoder (TC) and the MSC Server in **3G** for Tandem Free Operation in 3G-3G calls and in 3G-2G calls.

Note: The actions within the MSC Server are harmonised with the Out-of-Band Transcoder Control (OoBTC) for Transcoder Free Operation (TrFO).

---

### D.2 Overview

Tandem Free Operation in 3G-3G calls and 3G-GSM calls implies that the different entities of the Core Network and Radio Access Networks collaborate. Figures D.2-1a and D.2-1b provide an overview of the nodes involved in Tandem Free Operation and the interfaces between these nodes.

The interfaces as shown in figures D.2-1a and D.2-1b are:

- **MSC-MSC Interfaces:** The ISUP protocol is not influenced by TFO. Optionally the OoBTC protocol (not shown) should take the Optimal Codec Type and Configuration and the Distant Codec List into account. If this feature is not desired then the Optimisation Mode shall be set to "No Change". This feature is mandatory when the Optimisation Mode has been set to "Change".
- **RANAP:** This Interface between MSC and RNC is not influenced by TFO.
- **Iu Interface:** This interface between MGW and RNC is not influenced by TFO.
- **H.248:** This interface between MGW and MSC Server has to provide the configuration data to the TC. In the minimal version this shall contain the Local Codec Type and the Configuration, with the Optimisation Mode set to "No Change". The Local Codec List is optional. If the Optimisation Mode has been set to "Change" then the TC shall send (after the TFO Negotiation has taken place) the Optimal Codec Type and Optimal Configuration, as well as the Distant Codec List back to the MSC Server.
- **Nb Interface:** This interface carries (in case of TFO) the PCM samples and embedded in these the TFO Messages and TFO Frames.

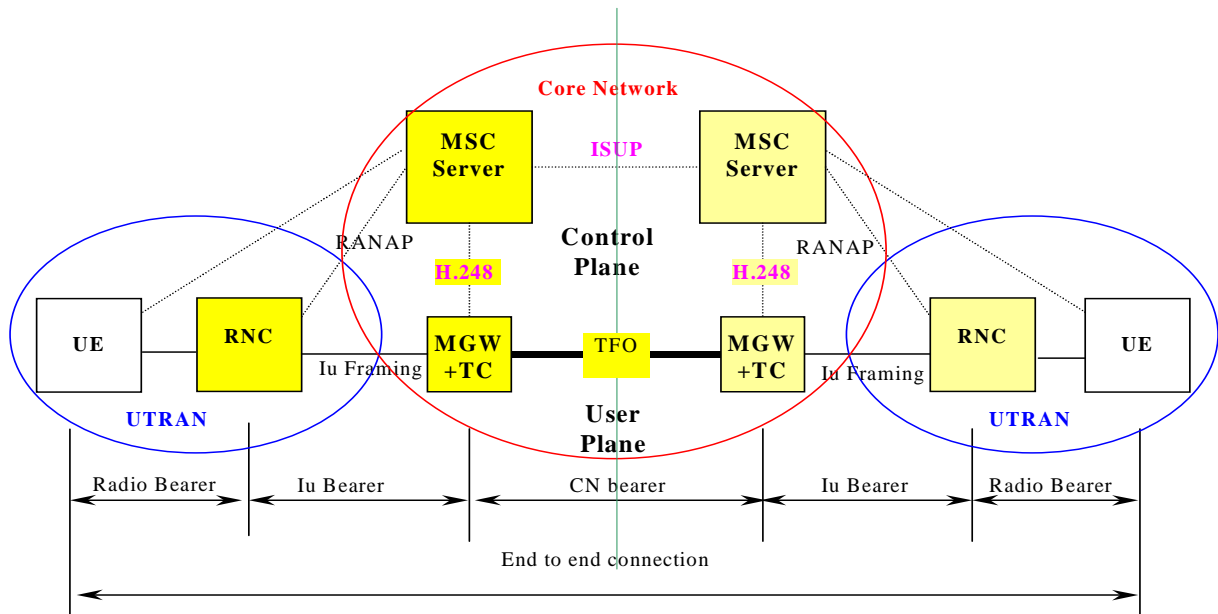


Figure D.2-1a: Nodes and Interfaces for TFO in UMTS-UMTS Calls

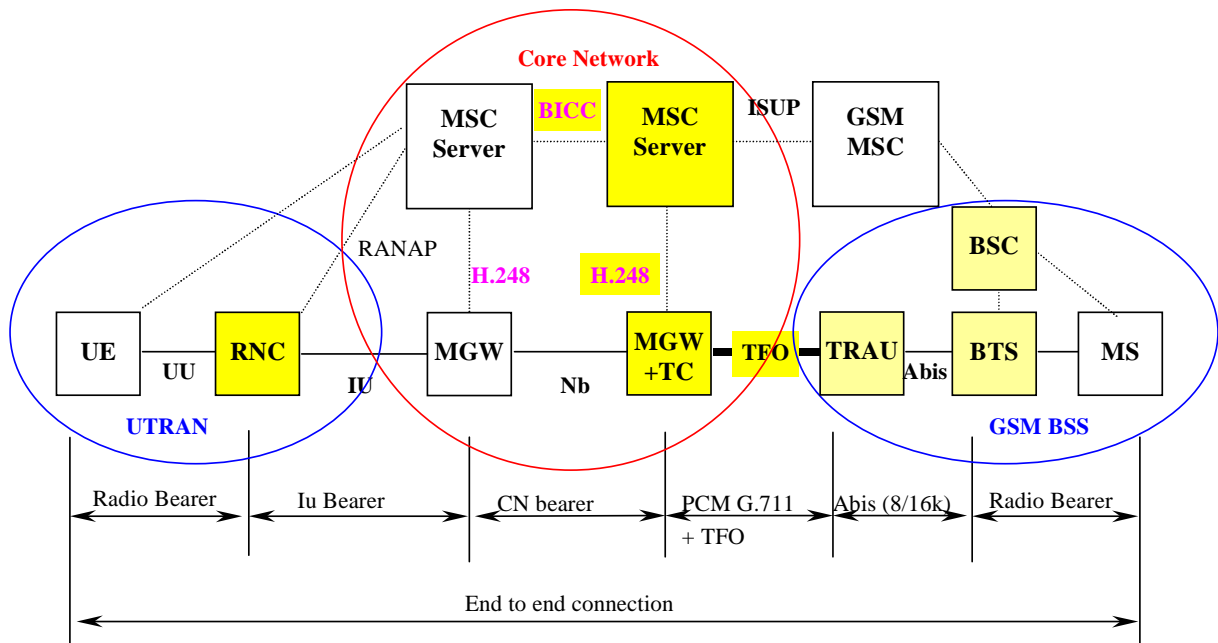


Figure D.2-1b: Nodes and Interfaces for TFO in UMTS-GSM Calls

TFO in UMTS involves the following two processes:

**TFO\_TC:** within the MGW, for all Speech Codec Types;  
handles the TFO\_Protocol and its four sub-processes,  
including Codec Optimisation and Mismatch Resolution and Rate Control

**TFO\_MSC:** within the MSC\_Server; for all Speech Codec Types;  
handles Initialisation and optionally Codec Modification

The RNC handles the Rate Control, required for the AMR and AMR-WB Speech Codec Types, but this procedure is not impacted by TFO.

These different processes and the inter-processes dialogues are described in the following clauses.

## D.2.1 TFO\_TC

Tandem Free Operation is essentially managed by the TC. In the simplest implementation version (Optimisation Mode set to "No Change") the TC can establish and maintain TFO fully on its own (within certain limits) as described below.

For all Codec Types the TC is responsible for the inband TFO Protocol, i.e. the TFO negotiation, TFO setup and the fast fall back to normal operation, if necessary. The TC has to monitor the ongoing call permanently for fast reaction, if required.

In all cases the TC has to perform the TFO Decision algorithm (see clauses 11 and 12). This TFO decision algorithm takes all known local and distant configuration parameters into account and identifies whether TFO is possible and what are the optimal call configuration parameters (Optimal Codec Type and Codec Configuration) in the given situation. If the Optimisation Mode is set to "Change" then the TC has the responsibility to inform the MSC Server about any change in the distant call configuration, especially the distant alternative Codec List. It is then mandatory for the MSC Server to evaluate this information.

If the Optimisation Mode has been set by the MSC Server to "Change", then the TC shall provide to the MSC Server the optimal call configuration parameters resulting from the TFO Decision algorithm. It is then mandatory for the MSC Server to evaluate these parameters and to perform the necessary Codec Modification.

In case of the AMR Codec Types the TC is responsible for the TFO relevant Rate Control. It shall limit the maximally allowed Rate (Codec Mode) in a way that it is always within the Common Active Codec Set of both sides. During TFO Konnect the TC is responsible to steer the uplink rate down to the TFO Setup Mode and release it as soon as TFO is in Operation.

## D.2.2 TFO\_MSC

The Call Control Layer has the overall responsibility, especially for all resources, on the Radio Access Network (RAN) and the Core Network (CN). For all Codec Types it is responsible for Call Setup, Handover and Supplementary Services. The Call Control Layer should take care that the call configuration is not modified during handover unless absolutely necessary, because in TFO (TrFO) every local change has direct influence on the distant side. The Call Control Layer is responsible that TFO is properly terminated before handover, if the call configuration after handover is not longer TFO compatible.

The MSC Server is responsible for the interaction between Call Control Layer and the inband TFO signalling. It shall support of the TC with the necessary configuration parameters (Codec Type, Codec Configuration, Optimisation Mode, optional the alternative Codec List, etc). The MSC Server is responsible to enable or disable TFO.

The MSC Server is responsible for the change of the Codec Type and/or Codec Configuration, e.g. for Mismatch Resolution and Optimisation for TFO, if this is required or better for Tandem Free Operation and requested by the TC. This feature is optional for the MSC Server unless the Optimisation Mode is set to "Change".

---

## D.3 TFO\_TC

The following clauses describe the actions within the TC to establish and maintain Tandem Free Operation in terms of a State Machine, respectively TFO Processes, handling synchronization and protocol. The description of the TFO

Protocol does not reflect implementation details for the I/O Processes (Rx\_IU, Tx\_IU, Tx\_TFO, and Rx\_TFO), but they may need to be considered for the exact understanding of the behavior. Only the TFO\_Protocol Process is detailed, which is responsible for the handling of the TFO Protocol.

The TFO\_TC can be regarded as consisting of five processes, which are strongly coupled to each other, which run in parallel, but phase shifted. The TFO\_Protocol Process communicates with the TFO I/O processes and, optionally, with its corresponding process within the MSC Server (TFO\_MSC) to resolve Codec Mismatch, see Figure D.3.1-1.

Under normal circumstances (exceptions occur during time alignments or octet slips) all TFO I/O Processes are triggered every 160 samples or every speech frame of 20 ms. All events and actions are quantized in time into these smallest intervals.

It can be assumed that the processing times for the TFO Processes are very short and negligible. However, it must be ensured that no timing ambiguity occurs between the Processes. This means the processing and exchange of information between them do not overlap in time. Care must be taken especially when time alignment occurs, which may be independent in uplink and downlink. During these time alignments the TFO Frames or TFO Messages may shift in time and consequently the triggering point for the related TFO Processes changes, too.

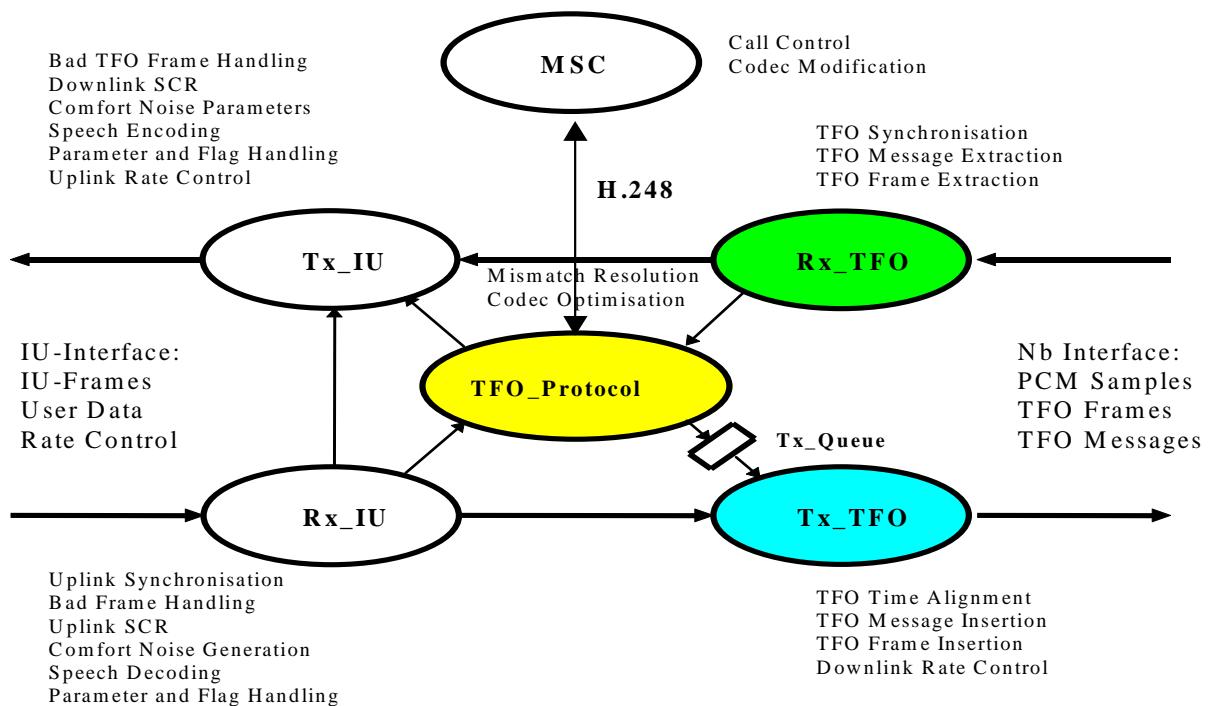


Figure D.3-1: The five TFO\_TC Processes and the TFO\_MSC Process

### D.3.1 Rx\_IU Process

The Rx\_IU Process receives Uplink IU Frames from the IU Interface and checks correct synchronisation and contents. It performs all actions of a conventional Uplink TC. It extracts the data bits and calls, if appropriate, the Bad Frame Handler, the Uplink DTX functions and Comfort Noise Generator and finally the Speech Decoder. In case of AMR-WB a successive downsampling is performed before G.711 encoding.

The resulting speech samples are handled to the Tx\_TFO Process for output to the Nb interface. In addition Rx\_IU passes the Uplink IU Speech Parameters directly and unaltered to Tx\_TFO.

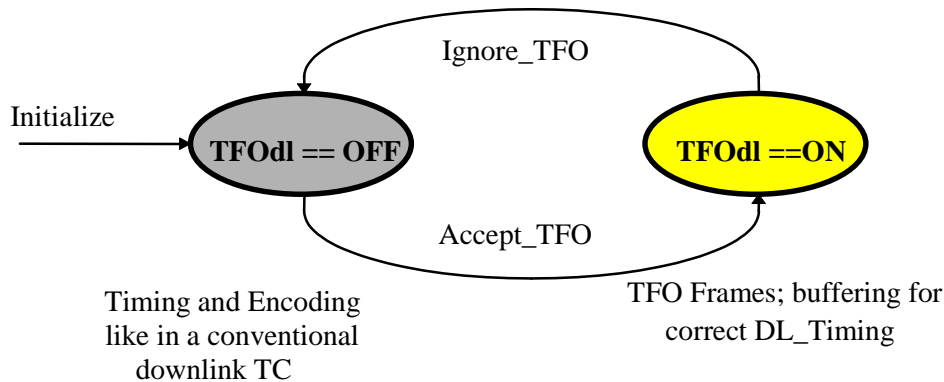
It further extracts the Rate Control information (if any) from the Uplink IU Frames and sends corresponding Rx\_IU Messages to the Tx\_IU Process, the Tx\_TFO Process and the TFO\_Protocol Process.

## D.3.2 Tx\_IU Process

The Tx\_IU Process builds autonomously the relevant downlink IU Frames and sends them in the correct phase relation onto the IU-Interface as commanded by the (optional) time alignment from the RNC.

Tx\_IU has two major States: TFOdl == OFF (start-up default state) and TFOdl == ON (see Figure D.3.2-1).

TFO\_Protocol controls the transitions between these states using the Accept\_TFO (AT) and Ignore\_TFO (IT) commands.



**Figure D.3.2-1: States of the Tx\_IU Process**

**During TFOdl == OFF** Tx\_IU performs all actions of a conventional downlink TC: On command from Rx\_IU it performs necessary downlink time alignments (optional). It samples one frame of speech samples in the correct phase position and calls the Speech Encoder. 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 IU interface. In case of AMR, Tx\_IU furthermore steers the AMR Codec Mode according to the UL Rate Control Command received from the Rx\_IU.

**During TFOdl == ON** no Bad Frame Handling or Comfort Noise Parameter Handling are performed. The speech parameters extracted from the TFO Frames are passed as Downlink IU Frames with least possible delay down to the RNC. The Tx\_IU shall not perform any Error Correction, but rather relay the received parameters unaltered through. If a synchronisation error or a CRC error is detected in the TFO Frame, the payload CRC of the IU frame shall be inverted bit by bit.

Tx\_IU performs Maximum Rate Control for the uplink direction by taking the minimum of the local "Max\_Rate" parameter and the received Rate Control parameter from Rx\_TFO and sends this downlink to the RNC, whenever a change in this result occurs. This Rate Control is independent of the TFO state. The exact handling of the Rate Control Commands on the IU interface is described in 3GPP TS 25.415. In case the TFO\_Protocol alters the Max\_Rate parameter a Rate Control Command has to be sent.

## D.3.3 Tx\_TFO Process

The Tx\_TFO Process gets directly and with minimal delay the unaltered Uplink speech parameters and control bits and with some delay the decoded speech PCM samples from Rx\_IU. It further gets internal messages (commands) from TFO\_Protocol via the Tx\_Queue, or directly without delay.

Tx\_TFO has two major States: TFOul == OFF (default at beginning) and TFOul == ON, see Figure D3.3-1. Toggling between these two States is commanded by TFO\_Protocol with Begin\_TFO (BT) and Discontinue\_TFO (DT).

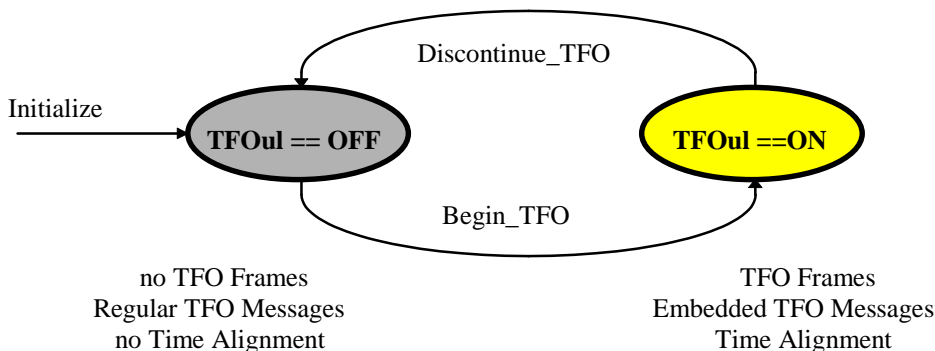


Figure D.3.3-1: States of the Tx\_TFO Process

During TFOul == OFF, decoded speech PCM samples and regular TFO Messages (if any) are sent onto the Nb interface. Time Alignment takes place only once, just before the beginning of the first regular TFO Message.

During TFOul == ON, TFO Frames and embedded TFO Messages (if any) are sent. Time Alignment takes place just before the first TFO Frame and then whenever required in between two TFO Frames.

The Tx\_TFO Process builds the relevant TFO Frames and sends them in the correct phase relation onto the Nb-Interface. Time alignment of TFO Messages and TFO Frames are handled autonomously and independent of the TFO\_Protocol Process. Rx\_IU informs Tx\_TFO about any changes in the phase position of the Uplink IU Frames and Tx\_TFO inserts automatically the correct number of T\_Bits before the next TFO Frame, and embeds autonomously the next TFO\_Message or a TFO\_FILL Message, if necessary.

The TFO\_Protocol Process can send internal messages into the Tx\_Queue (First In, First Out). Tx\_TFO shall take the message from the Tx\_Queue one by one, shall process them autonomously and shall send the resulting TFO Messages in correct order and phase position, as regular or as embedded TFO Messages. Tx\_TFO shall generate a Runout Message to TFO\_Protocol, if the last TFO\_Message is sent without guarantee of a direct continuation by another TFO Message, i.e. if the (possible) IPEs may have run out of synchronisation (see Appendix A). TFO\_Protocol may delete messages from Tx\_Queue, as long as they are in there: Command "Clear Tx\_Queue", at time Tc.

Basically, messages or commands that are already in processing by Tx\_TFO at Tc can not be stopped, deleted or interrupted. The TFO Protocol is designed to work properly with that default handling, although not with fastest processing.

But, Tx\_TFO shall investigate at Tc, how far the transmission of the current TFO Message has proceeded and shall "Modify on the Fly" this last TFO\_Message before Tc into the first one after Tc, see Figure D.3.3-2.

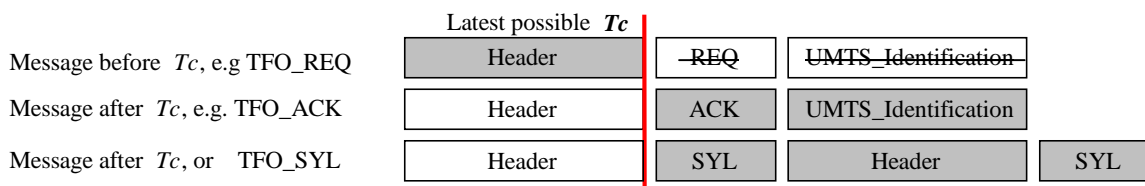


Figure D.3.3-2: Examples of Modification on the Fly within the Header Transmission

Tx\_TFO performs Maximum Rate Control for the downlink direction by taking the minimum of the local "Max\_Rate" parameter and the received Rate Control parameter from Rx\_IU and sends this minimum uplink to the distant TFO partner.

### D.3.4 Rx\_TFO Process

The Rx\_TFO Process receives TFO Messages and TFO Frames from the Nb-Interface and synchronises to them, i.e. checks correct synchronisation and contents. It bypasses all PCM samples and Speech parameters directly to Tx\_IU for further processing. The Rx\_TFO Process further extracts all the control bits and TFO Messages and sends corresponding Rx\_TFO Messages to the TFO\_Protocol Process.

When the Rx\_TFO received distant TFO parameters, either by TFO Messages or TFO Frames (Config\_Prot Frames), it relays them to the TFO\_Protocol Process.

### D.3.4.1 Search for and Monitoring of TFO Synchronization

See Annex C, clause C.3.4.1 for the detailed description.

## D.3.5 TFO\_Protocol Process

The TFO\_Protocol Process consists of a set of different states. The initial state shall be Not\_Active. The TFO\_Protocol Process is typically invoked whenever a message is received, either from Rx\_IU, Rx\_TFO, Tx\_TFO or the MSC Server.

Some key events are due to modifications of the local configuration:

- a modification of the used speech Codec Type (New\_Local\_Codec); or
- its Configuration Parameters (e.g. the ACS in case of AMR) (New\_Local\_Configuration); and
- a modification of the list of the alternative speech Codec Types (New\_Local\_Codec\_List);
- TFO Enable or TFO Disable.

These parameters are received from the MSC Server, e.g. via the vertical interface using H.248.

### D.3.5.1 Messages from the MSC Server to TFO\_Protocol

Rx == New_Speech_Call (Local_Used_Codec);	Initialises the TC.
Rx == New_Local_Codec (New_Local_Used_Codec);	In Call Modification to another Codec Type or Configuration.
Rx == Data_Call;	In Call Modification to Data_Call (not relevant)
Rx == New_Local_Codec_List (Codec_List);	Information on available resources
Rx == TC_Idle;	The TC is set into Idle mode (equivalent to TRAU_Idle see clauses 10.4 and C.3.5.1)
Rx == TFO_Enable;	Enable the TFO_Protocol process
Rx == TFO_Disable;	Disable the TFO_Protocol process

### D.3.5.2 Messages from TFO\_Protocol to Tx\_IU

Tx_IU:= Accept_TFO;	If TFO Frames are correctly received, they shall be used.
Tx_IU:= Ignore_TFO;	TFO Frames, even if received correctly, shall be ignored.
Tx_IU:= Set_Max_Rate (Max_Rate);	The Rate Control is limited to an upper bound. This command is executed immediately. It triggers a Rate_Control_Req to be sent to the RNC. The RNC has to acknowledge this by Rate_Control_Ack.

### D.3.5.3 Messages from TFO\_Protocol to the MSC Server

Tx_MSC:= Optimal_Codec_Type ();	Triggers a Codec Modification by OobTC
Tx_MSC:= TFO_Status ();	Inform about TFO status and configuration

### D.3.5.4 Messages between TFO\_Protocol and Tx\_TFO

The symbol () indicates that these Messages contain parameters, see Clause 8.

Tx:= TFO\_REQ ();                    main TFO\_REQ Message.



Tx:= TFO_ACK ();	main TFO_ACK Message, response only to TFO_REQ.
Tx := TFO_REQ_L ();	used in Mismatch, Operation and Periodic_Retry to inform about alternative Codec Types and Configurations
Tx:= TFO_ACK_L ();	response only to TFO_REQ_L.
Tx:= Con_Req ();	used in TFO to inform the distant TFO Partner about the local Configuration; second method to TFO_REQ_L with same parameters, but 10 times faster;
Tx:= Con_Ack ();	used in TFO to respond to Con_Req;
Tx:= TFO_TRANS ();	command IPEs to go transparent.
Tx:= TFO_NORMAL;	reset IPEs into their normal operation.
Tx:= TFO_FILL;	mainly to pre-synchronise IPEs.
Tx:= TFO_DUP;	"I receive TFO Frames in Establishment".
Tx:= TFO_SYL;	"I lost TFO Frame synchronisation".
Tx:= Begin_TFO;	Insert TFO Frames from now on.
Tx:= Discontinue_TFO;	Discontinue inserting TFO Frames.
Clear Tx_Queue;	Clears all remaining commands from Tx_Queue. This command is executed immediately and does not go via the Tx_Queue (of course not).
Tx:= Set_Max_Rate (Max_Rate);	The Rate Control is limited to an upper bound.. This command is executed immediately and does not go via the Tx_Queue!
Rx == Runout;	Tx_TFO reports that the continuous stream of outgoing TFO Messages may be interrupted soon.

### D.3.5.5 Messages from Rx\_TFO to TFO\_Protocol

The symbol () indicates that these Messages contain parameters, see Clause 8.

Rx == TFO_REQ ();	
Rx == TFO_ACK ();	
Rx == TFO_REQ_L ();	
Rx == TFO_ACK_L ();	
Rx:= Con_Req ();	
Rx:= Con_Ack ();	
Rx == TFO_TRANS ();	serves as alternative, faster TFO_ACK in some cases!
Rx == TFO_NORMAL;	
Rx == TFO_FILL;	
Rx == TFO_DUP;	
Rx == TFO_SYL;	
Rx == TFO_Frame ();	TFO_Frame (Distant_Used_Codec; Number_of_Received_Frames).
Rx == Frame_Sync_Lost ();	Frame_Sync_Lost (Number_of_Lost_Frames).

Rx == Mess\_Sync\_Lost;      Message\_Sync\_Lost.

Rx == PCM\_Non\_Idle;      at the beginning of a period with several samples/frame  
that are different from the PCM\_Idle sample.

The message "TFO\_Frame ()" needs to be sent only at the first five occurrences, either after a not valid TFO Frame, or if the Distant\_Used\_Codec changed.

The message "Frame\_Sync\_Lost ()" needs to be sent only at the first five occurrences of errors in TFO Frames or loss of synchronisation, after a correctly received TFO Frame.

The message "Mess\_Sync\_Lost" is sent, when after a valid TFO Message no following TFO Message is found.

### D.3.5.6 Messages from Rx\_IU to TFO\_Protocol

Rx\_IU := Rate\_Control\_Ack (Max\_Rate);      The Rate\_Control\_Req is acknowledged.  
This is important for the TFO Protocol  
In addition the downlink rate may be limited to an upper bound.  
This is reported to Tx\_TFO and to Tx\_IU.

## D.4 TFO in the RNC

The RNC does not differentiate between "Normal Tandem Operation", "Transcoder Free Operation" or "Tandem Free Operation". Therefore no TFO\_RNC process is necessary.

The RNC is aware that Rate\_Control\_Req Commands may be sent from the CN that restrict the maximally allowed rate in uplink direction. It sends Rate\_Control\_Ack messages back for confirmation, including the Rate Control for downlink. For details see 3GPP TS 25.415. The Rate\_Control\_Ack is important for the TFO\_Protocol to go into the KONNECT state.

Note 0: Iu User Plane Frame Protocol (see [17]) Release '99 cannot be used for TFO, unless there's only one mode in the ACS, since it does not support up-link rate control.

## D.5 TFO\_MSC

The MSC Server in UMTS, which controls the Media Gate Way (MGW) and the Transcoder (TC) within the MGW, is responsible for the interaction between the "Out-of-Band-Transcoder Control" (OoBTC) and the "inband TFO" control. The communication between this Control Layer and the Transport Layer is performed e.g. via a "vertical" interface using the H.248 protocol .

The MSC Server provides the necessary configuration parameters to the TC at call setup:

- Used Codec Type                      (mandatory)
- Codec Configuration                (mandatory)
- Optimisation Mode                  (mandatory)
- Alternative Codec List              (optional)
- TFO\_Enable / TFO\_Disable.      (mandatory)

These parameters may be changed during the call ("Codec Modification").

It is up to the MSC Server, which Codec Types and Codec Configuration parameters it provides to the TC. But once it has provided them, the MSC Server commits to perform In\_Call\_Modifications, in case the TFO Protocol decides that another Codec Type or Configuration is preferred.

After call setup the TFO\_Protocol performs the inband negotiation and may determine a better, optimal Codec Type with optimal Configuration for TFO. This optimal Codec Type with Optimal Configuration parameters is reported to the MSC Server via the same vertical interface. The MSC Server has the duty to perform "Codec\_Modification", if it has offered these options, via the OoBTC.

In addition the TC provides the MSC Server with the distant Codec List, as received via the TFO interface. The TC has the duty to update the MSC Server with these parameters whenever a modification on the distant side becomes available.

When the MSC Server got notice that TFO is ongoing it shall try to avoid changes of the Codec Type or Configuration.

## D.5.1 Status of the Connection

The TC shall inform the MSC Server of its status with two messages:

- *TFO\_Off* TFO is not established.
- *TFO\_On* TFO is established and ongoing.

## D.5.2 Change of Codec Type or Configuration

If TFO is ongoing the MSC Server shall try to keep the Codec Type and Configuration as far as possible during the call. If this is not possible, e.g. due to handover or supplementary services, then the MSC Server shall disable TFO before changing to a new Codec Type or to a new Configuration that is not TFO compatible.

The new Codec Type may be selected taking the Codec List of the distant TFO partner into account. TFO may be enabled again by the MSC Server after the change has been performed.

---

## D.6 Determination of the Optimal Codec Type and Optimal Configuration Parameters

The determination of the Optimal Codec Type and Optimal Configuration Parameters for TFO is performed within the TC, based on the TFO Decision rules (see clauses 11 and 12). The Optimal Codec Type and Configuration is then reported to the MSC Server.

If a change of the Codec Type is not desired, then the MSC Server shall not provide more than one Codec Type within the Codec List.

If a change of the Codec Configuration is not desired, then the MSC Server shall not provide the Optimisation Mode with "Change".

But if Mismatch Resolution and Optimisation is allowed, then the MSC Server shall receive from the TC the optimal Codec Type and Optimal Configuration Parameters and the distant Codec List. The MSC Server shall accept the proposed Optimal Codec Type and its proposed optimal Configuration and perform Codec Modification. This ensures that both radio legs obtain the same result, as negotiated via the TFO protocol. If necessary TFO is disabled before and enabled after the modification.

## 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 for AMR-NB 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 program 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 '-'. They are indicating the 8 AMR-NB 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. 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.:

```

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.

---

## Annex F (informative): Operator's Guide

This clause presents guidelines, which should be followed by the operator to optimise the establishment of TFO with AMR, and avoid unnecessary intra cell hand-overs for configuration optimisation once TFO is established.

The guidelines can be classified into the following families :

- Avoidance of Codec Type Optimisation;
- Earliest possible TFO Establishment;
- Usage of AMR tandem in preference of TFO with GSM\_EFR, GSM\_FR, or GSM\_HR;
- Balance between Speech Quality and Network Capacity.

The guidelines are most helpful inside one PLMN. They can be applied to inter-PLMN constellations to extend their benefits. They may also be applied in parts of a PLMN, which would of course lower their benefits.

---

### F.1 Avoidance of Codec Type Optimisation

Depending on the call configuration on both sides of the connection a Codec Type Optimisation may follow after TFO has been established, because the full list of supported Codec Types is only available then. For this Codec Type Optimisation an intra-cell hand-over is required. In many call scenarios with a TFO Connection with HR\_AMR the resulting communication quality may be sufficient. The benefits of a Codec Type Optimisation by an intra cell hand-over may be not obvious, but the signalling load may be too costly.

#### **Guideline 1:**

If the operator wants to avoid any Codec Type Optimisation, then the supported Codec List shall contain only one, the Active Codec Type.

---

### F.2 Earliest possible TFO Establishment

Since speech quality is improved by TFO, it is important to establish TFO as soon as possible. This can be achieved by reducing / simplifying the TFO negotiation.

This leads to two categories of guidelines:

1. Immediate TFO establishment without Codec Mode Optimisation (TFO is established with the current ACS, or with a subset of this ACS).
2. Immediate TFO establishment with Codec Mode Optimisation (after TFO establishment the ACS may be changed by a) Intra Cell Hand-over, b) Mode Modify or c) RATSCCH).

#### F.2.1 Avoidance of Codec Mode Optimisation

##### **Guideline 2:**

If the operator wants to avoid Codec Mode Optimisation after TFO establishment with AMR, then he shall set the "Optimisation Mode" to "No\_Change".

##### **Guideline 3:**

The operator should configure AMR so that MACS = 4 and the ACS e.g. corresponds to the default sets (10,20, 6,70, 5,90, 4,75 for FR\_AMR, UMTS\_AMR\_2 and OHR\_AMR and 7,40, 6,70, 5,90, 4,75 for HR\_AMR). For UMTS\_AMR the ACS should include only mode 12,20 kbps. By this the chance for Inter-PLMN TFO is enhanced. Other ACSs for FR\_AMR, UMTS\_AMR\_2, OHR\_AMR and HR\_AMR are possible. They should include as many

as possible common Codec Modes in the lower, contiguous subsets. Also for UMTS\_AMR other ACSs are possible. In that case Inter-PLMN TFO is not as obvious and may need other inter-operator agreements.

NOTE: The default sets correspond to the ACSs determined by the TFO Decision algorithm, when all Codec Modes of the ACSs are included in the corresponding SCS.

#### **Guideline 4:**

The operator should configure AMR so that the ACSs are homogeneous within the whole PLMN (same ACS used in all BSS of a given PLMN for a given Codec Type: UMTS\_AMR\_2, FR\_AMR, OHR\_AMR, HR\_AMR). The ACSs of different Codec Types of the AMR Family should contain as many as possible Codec Modes within the common, lower, contiguous subset. Also UMTS\_AMR should be configured homogeneous within the whole PLMN, but since it is not compatible to the other AMR codec types, another configuration is possible (see Guideline 3).

#### **Guideline 5:**

If the network is heterogeneous, the operator should choose ACSs so that all resulting Common ACSs are acceptable (see clause 12), with as many as possible Codec Modes within the common, lower, contiguous subset.

#### **Guideline 6:**

If for AMR-WB an optimisation shall be restricted to certain configurations, this can be achieved by choosing a suitable configuration with OM=F(orbidden) in a combination with Maximum Rate Control.

Example: If configuration 4 and 5 shall not be possible, then configuration 2 shall be indicated. Then the effective ACS A and B (see Table 12.8-1) are possible, but not ACS C.

## F.2.2 Immediate TFO establishment with Codec Mode Optimisation

#### **Guideline 6:**

The operator should choose the ACSs in a way that all resulting immediate Common ACSs are acceptable and CACSs are subsets of Optimised ACSs (see clause 12).

Then TFO will most of the times establish immediately (with the obvious benefits in speech quality) and the Codec Mode Optimisation may be achieved with Mode Modify or RATSCCH, i.e. without the problematic Intra-Cell hand-over.

Remark: This guideline is not easy to fulfil since it is of course in general not possible to foresee all possible ACS constellations, especially not for inter-PLMN calls.

---

## F.3 Usage of AMR Tandem compared to TFO with GSM\_EFR, GSM\_FR, or GSM\_HR

#### **Guideline 7:**

If an AMR is the Active Codec Type and the operator prefers a tandeming connection with this AMR Codec on one side to a tandem free connection with GSM\_EFR, GSM\_FR or GSM\_HR, then he should not include GSM\_EFR, GSM\_FR or GSM\_HR within the supported Codec list.

Reason: The TFO algorithm will always try to establish TFO with the best available **common** Codec Type, which could be GSM\_EFR, GSM\_FR or GSM\_HR. But often a Tandem Connection including one AMR Codec Type may be preferable in terms of speech quality.

---

## F.4 Balance between Speech Quality and Network Capacity

The preference order for the Codec Type Optimisation and Codec Mismatch Resolution is based on speech quality and does not take into account the load in the network.

***Guideline 8:***

In capacity limited networks, the operator should only include Codec Types using half rate channels in the supported Codec List (GSM\_HR, HR\_AMR).



## Annex G (informative): Call flows for AMR TFO

Some example TFO protocol flows are shown for GSM (2G: left side) and UMTS (3G: right side) for a GSM-UMTS (2G-3G) TFO call. Other scenarios, like for GSM-GSM or UMTS-UMTS TFO calls can be derived by mirroring the relevant side. In cases where this is not directly obvious the other side is shown, too.

Configuration Frames (Con\_Req, Con\_Ack, etc) exist in GSM between BTS and TRAU (Abis Interface) as well as between TRAU and TRAU (the TFO Interface). They are used for delay measurements and for fast exchange of configuration parameters.

These Configuration Frames exist in UMTS between TC and TC or TC and TRAU (the TFO interface), but not on the Iu Interface. Instead the TFO Configuration is exchanged between SMSC and TC directly, e.g. via H.248 protocol. Optionally a proprietary interface between BSC and TRAU may also be used in GSM. In that case the Configuration Frames on the BTS-TRAU interface may be irrelevant.

The example in G.4 shows the version where the complete distant configuration is sent down to the BTS and further on to the BSC. In another version, G.5, only the Optimal Codec Type and Configuration is sent down to the BTS and BSC.

The protocol flows on the TFO interface (TRAU-TRAU, TRAU-TC, TC-TC) is in all cases identical.

### Notations:

The TFO\_Protocol States and the States of TFO\_BTS are marked in yellow. The messages are shown as they appear on the interfaces, i.e. after the TFO\_Protocol has already entered the new State.

The colours of the TFO Messages and Comments have no further meaning than highlighting the important parts and indicating what belongs together.

Some of the closed boxes contain comments or global descriptions of the ongoing procedures.

The TFO\_Messages require a relatively long transmission time, up to several hundreds of milliseconds. These transmission times are not reflected within these call flow charts. But please consider that some sequences that appear in chronological sequence within the charts are in practise occurring in parallel and are overlapping in time.

The left side in the flow charts is arbitrarily called "local" side and is shown as GSM, while the right side is called "distant" and is mainly shown as UMTS. But that is in most scenarios not relevant and the opposite is as true. The hand-over is per definition on the local side, just to simplify the discussion.

The mapping of the messages shown in the flow charts to the BTS-BSC messages is:

TFO\_Report (Distant Configuration) is "RemoteCodecConfigurationReport (Distant Parameters)"

TFO\_Report (Optimal Configuration) is "MultiRateCodecModeReq (Optimal Parameters)"

TFO\_Report (Delay) is "RoundTripDelayReport (delay)"

TFO\_Report (Status) is "TFO\_Report (Status)"

ChannelActivation () is "ChannelActivation (Configuration, Handover\_Indication)"

# G.1 Typical Initialisation for TRAU, TC and TFO Protocol

The following protocol flows show schematically the typical Initialisation of the TRAU, the TC and the TFO Protocol.

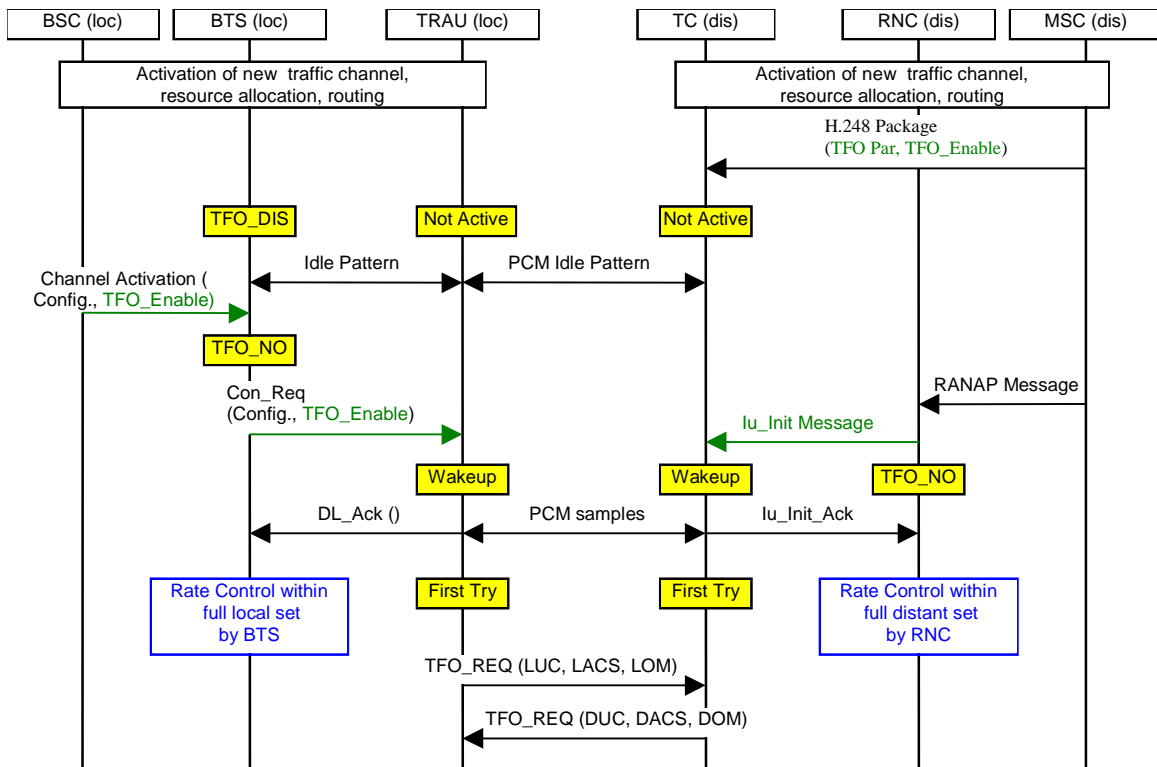


Figure G.1-1: Typical Initialisation of the TRAU, the TC and the TFO Protocol

## G.2 Re-Initialisation during the Call after TFO\_Disable

Sometimes the TFO Protocol is re-initialised during the ongoing call, e.g. after a TFO\_Disable.

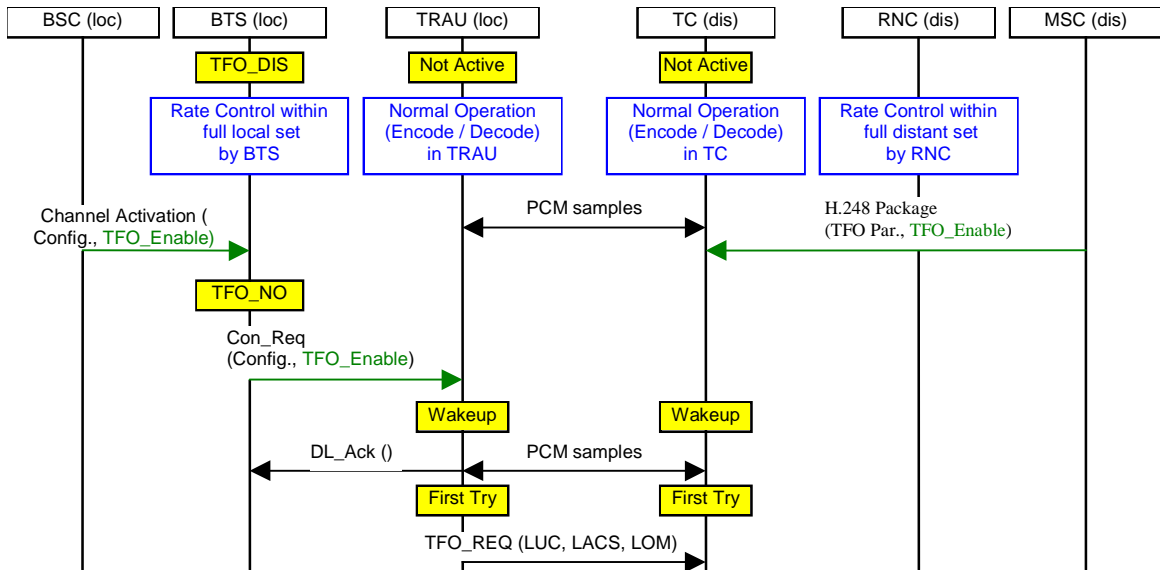


Figure G.2-1: Re-Initialisation during the Call after TFO\_Disable

## G.3 TFO\_Disable during Operation

### G.3.1 TFO\_Disable – passive partner: UMTS

The following protocol flow shows TFO\_Disable, where UMTS is the passive partner.

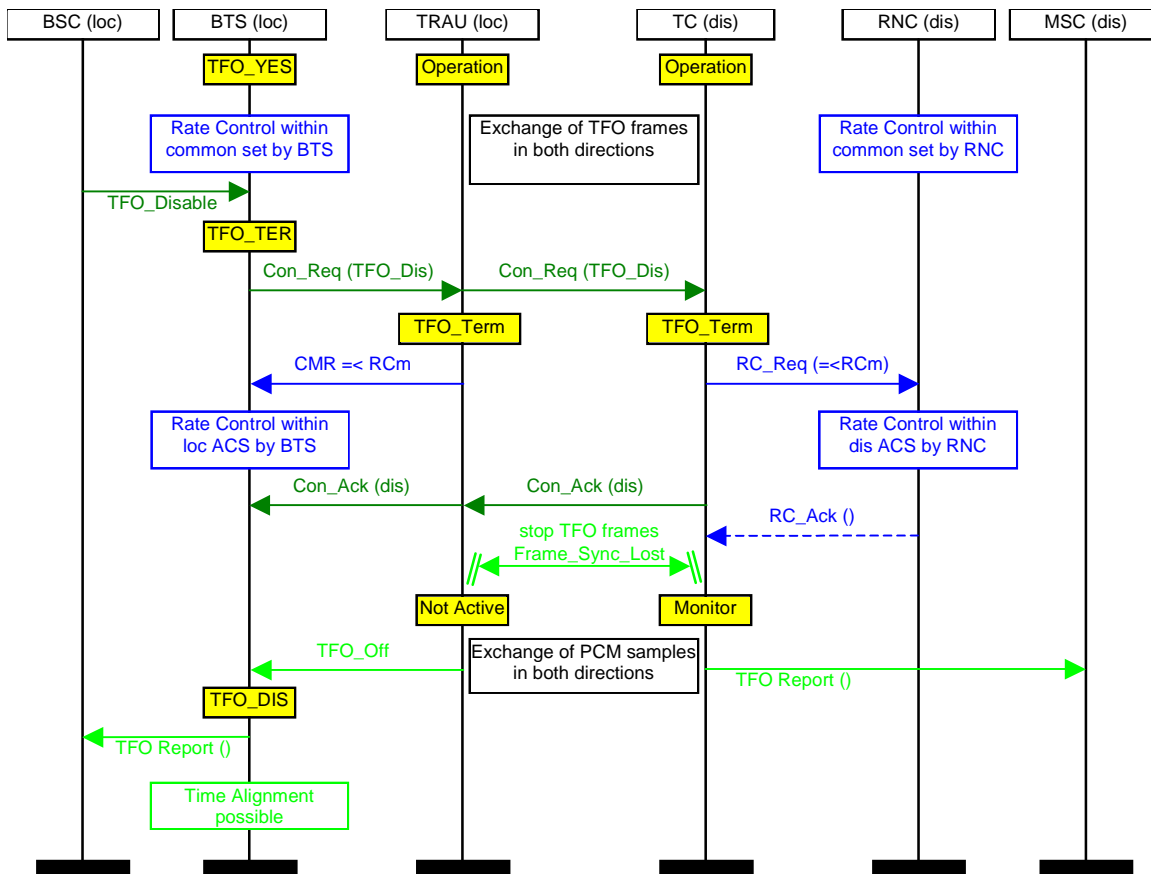


Figure G.3.1-1: TFO\_Disable during operation – passive partner: UMTS

### G.3.2 TFO\_Disable – passive partner: GSM

The following protocol flow shows TFO Disable, where GSM is the passive partner.

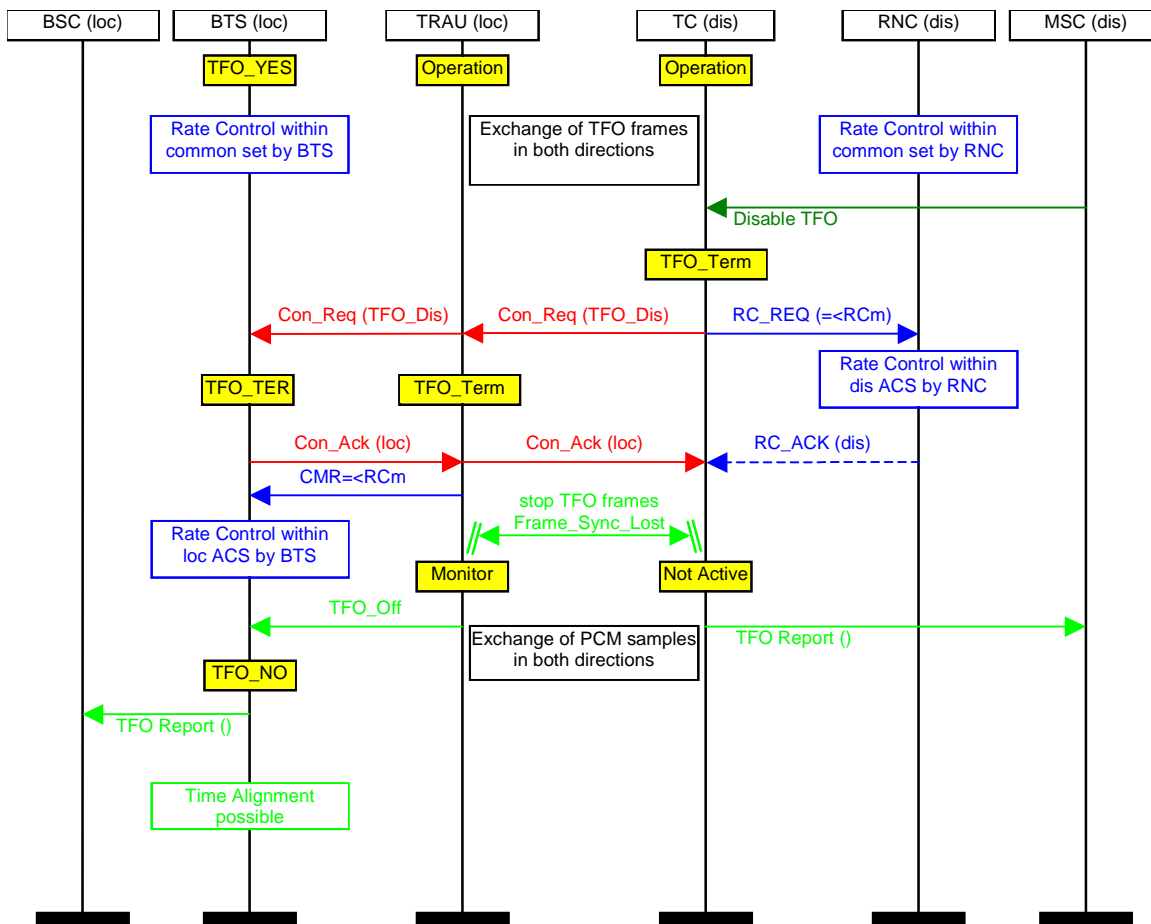


Figure G.3.2-1: TFO\_Disable during operation – passive partner: GSM

## G.4 Immediate TFO establishment for AMR

### G.4.1 Con\_Req / Con\_Ack used on the TFO Interface

The following protocol flow shows the example where immediate TFO setup is possible, either because both sides use identical Codec Types and Configurations, or because the Codec Types and/or Configurations are compatible in the "lower, contiguous subset". In the latter case potentially an optimisation phase might follow after TFO has been set up.

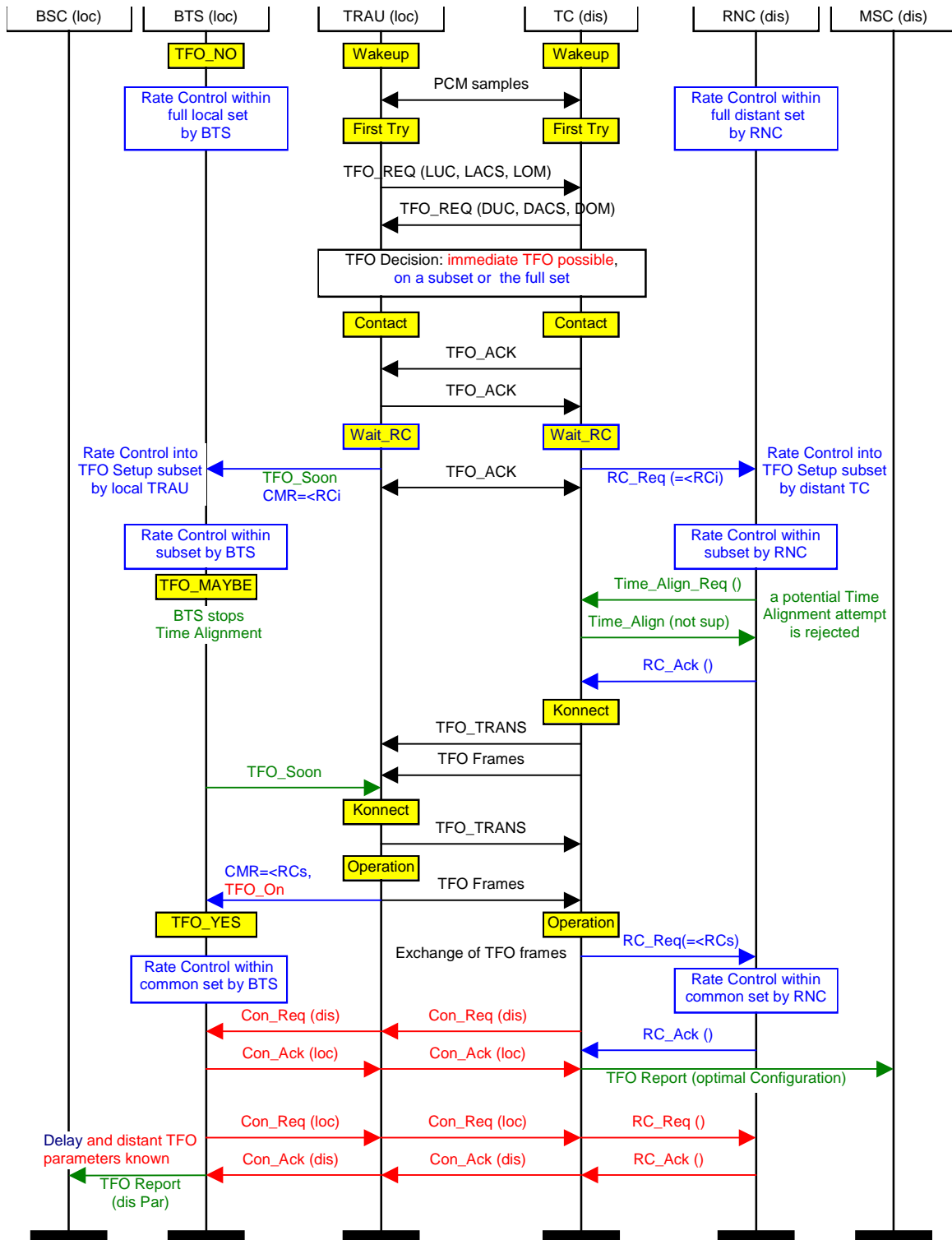


Figure G.4.1-1: Immediate TFO establishment for AMR with Con\_Req / Con\_Ack

NOTE: The round trip delay is important for the GSM side for optimal Link Adaptation. It should be as precise as possible and therefore the RNC on the distant side is taken into consideration, but not the Node B or the UE, because that would be too complicated. The round trip delay is not important on the UMTS side, since this radio channel is more stable due to fast power control.

## G.4.2 TFO\_REQ\_L / TFO\_ACK\_L used on the TFO Interface

The following protocol flow shows the same example as above. This version shows the option with TFO\_REQ\_L / TFO\_ACK\_L instead of Con\_Req / Con\_Ack on the TFO Interface. Please note that these TFO Messages take about 300ms for transmission, while Con\_Req / Con\_Ack need about 20ms.

In addition this example shows how the Optimal Configuration is reported to the BSC / MSC.

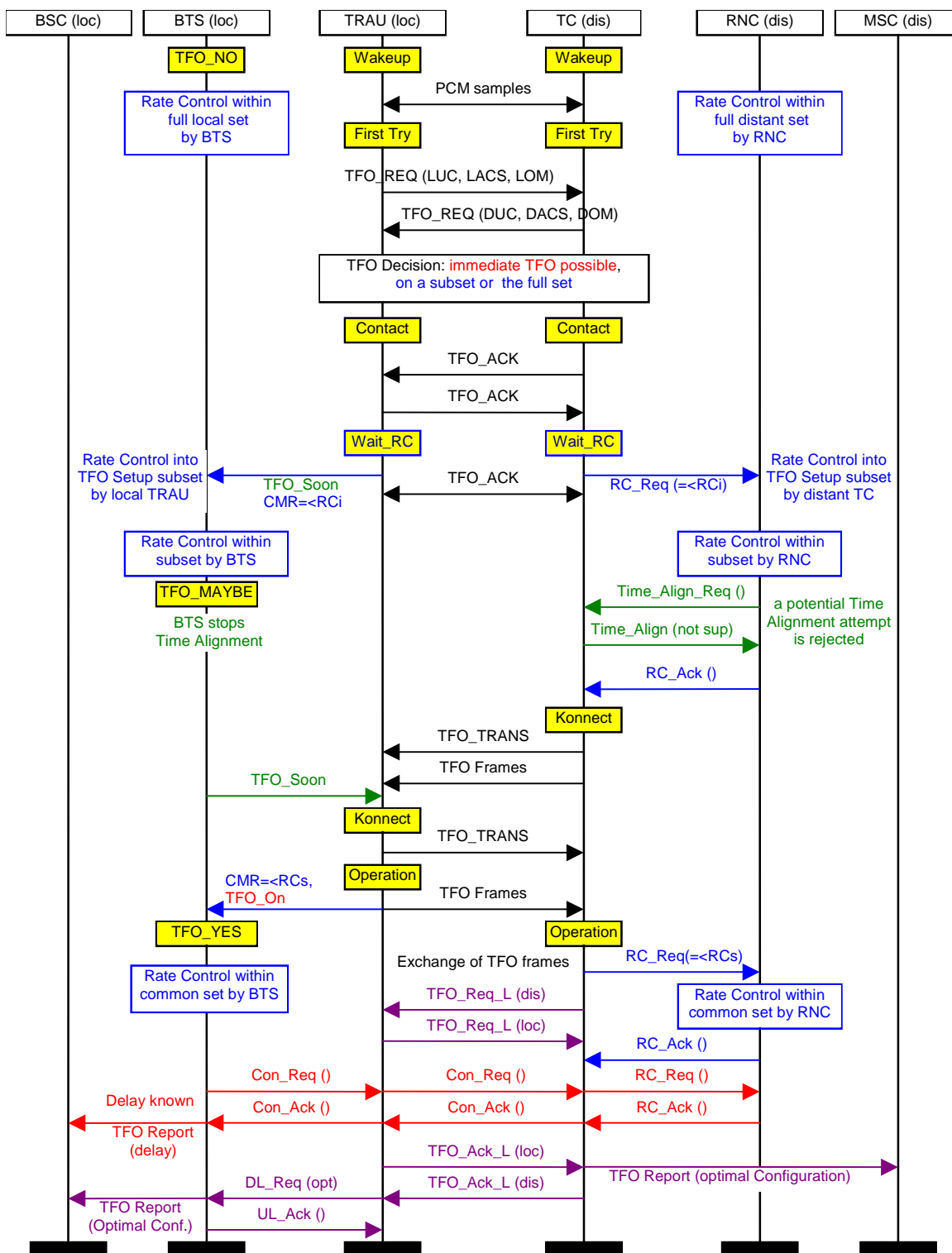


Figure G.4.2-1: Immediate TFO establishment for AMR with TFO\_REQ\_L / TFO\_ACK\_L

NOTE: The round trip delay measurement could be done without Config parameters (Par\_Type = 00).



## G.5 Configuration Optimisation

The following protocol flow shows the example where only the local side needs to change its AMR Configuration (the ACS) to the optimal configuration, while the distant side has this optimal configuration already (shown here), or does not need or want to change. Typically this optimisation takes place immediately after TFO setup and is triggered by the TFO Report to the BSC or the TFO Report to the SMSC.

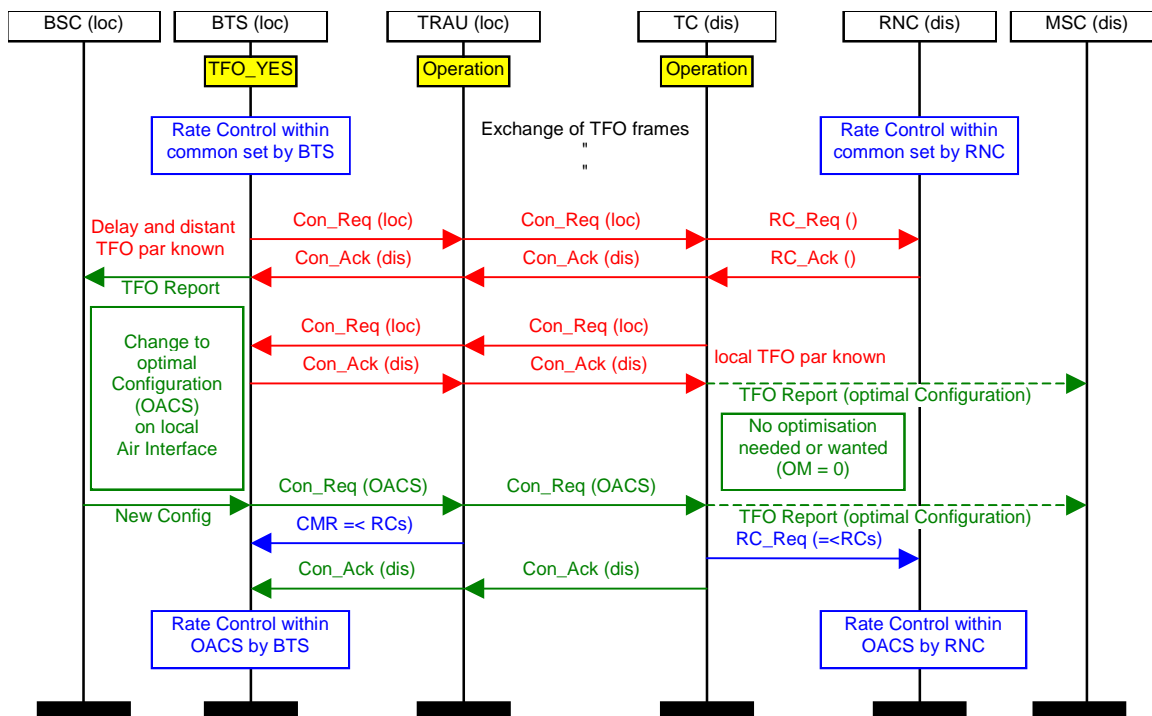


Figure G.5-1: Configuration Optimisation

## G.6 AMR TFO: Mismatch Case

The following protocol flow shows the example where the Codec Types or Codec Configurations do not match and where an immediate TFO is not possible. Potentially a mismatch resolution is following, in which case a second TFO establishment is attempted (indicated in dashed lines for local side).

In this example the TRAU reports the Optimal Configuration to the BTS and to the BSC.

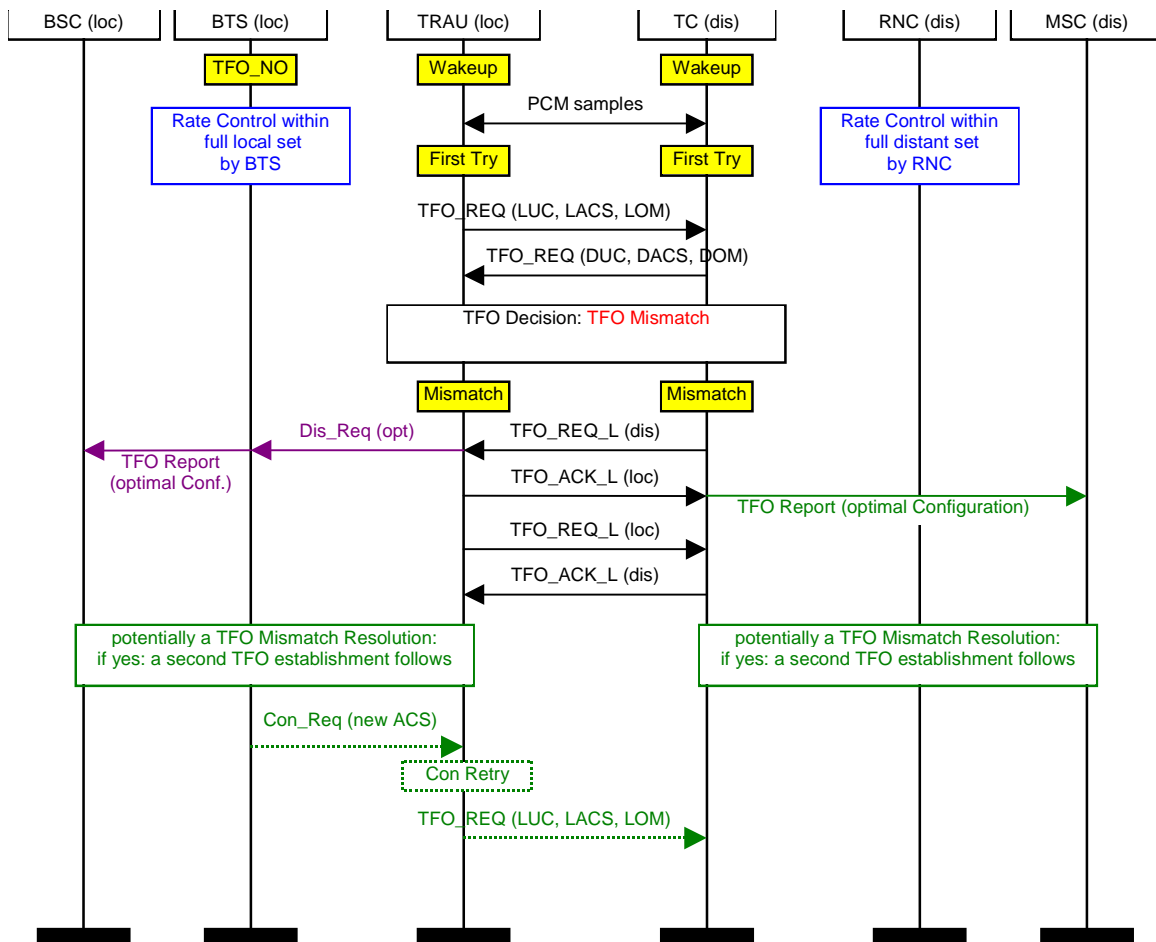


Figure G.6-1: AMR TFO Mismatch Case

## G.7 Intra BSC TFO Handover (TRAU remains)

### G.7.1 TRAU-TC TFO connection

The following protocol flow shows a local handover in a TRAU-TC TFO connection, where the local TRAU remains the same. The distant TC sees potentially some phase alignment of the TFO Frames, but no interruption of the TFO Operation. The Round Trip Delay is not important for TC and RNC and is therefore not measured. The local BTS estimates the round trip delay when it receives the first Con\_Ack.

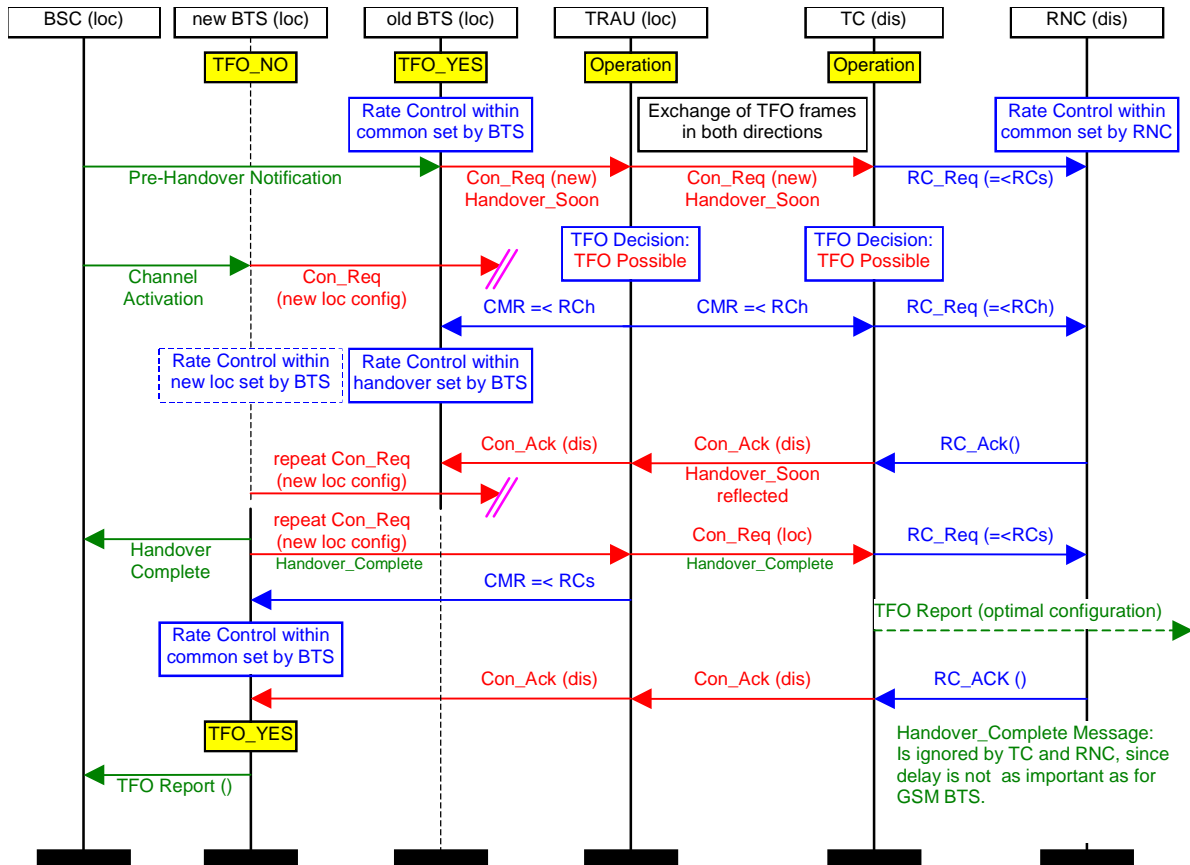


Figure G.7.1-1: Intra BSC TFO Handover - TRAU-TC TFO connection

## G.7.2 TRAU-TRAU TFO connection

The following protocol flow re-shows the lower part of a local handover in a TRAU-TRAU TFO connection, where the local TRAU remains the same. The Round Trip Delay is important for both BTSs and therefore the Handover\_Complete Message triggers a new delay measurement within the distant BTS.

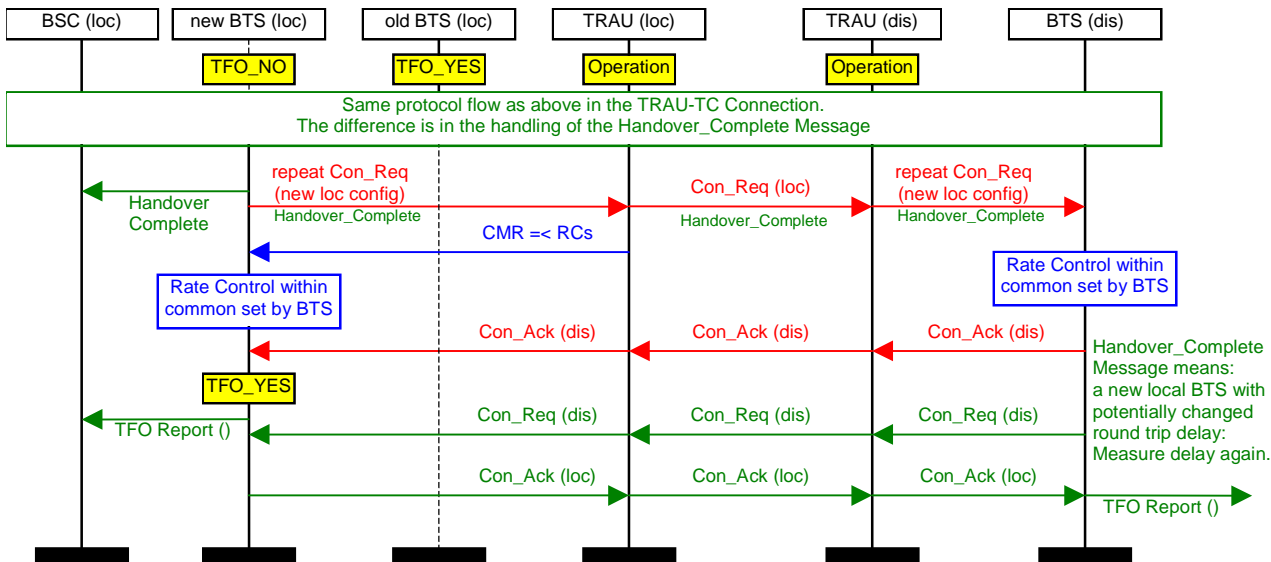


Figure G.7.2-1: Intra BSC TFO Handover - TRAU-TRAU TFO connection

## G.8 Inter BSC TFO Handover (TRAU changes)

The following protocol flow shows a hard local handover in TFO, where the local TRAU changes. New BTS and new TRAU are synchronised and working before the handover takes place. On the TFO Interface the fast handover handling is applied. The handling of the Handover\_Complete Message on the distant side is as described for the Intra BSC handover (not shown here again).

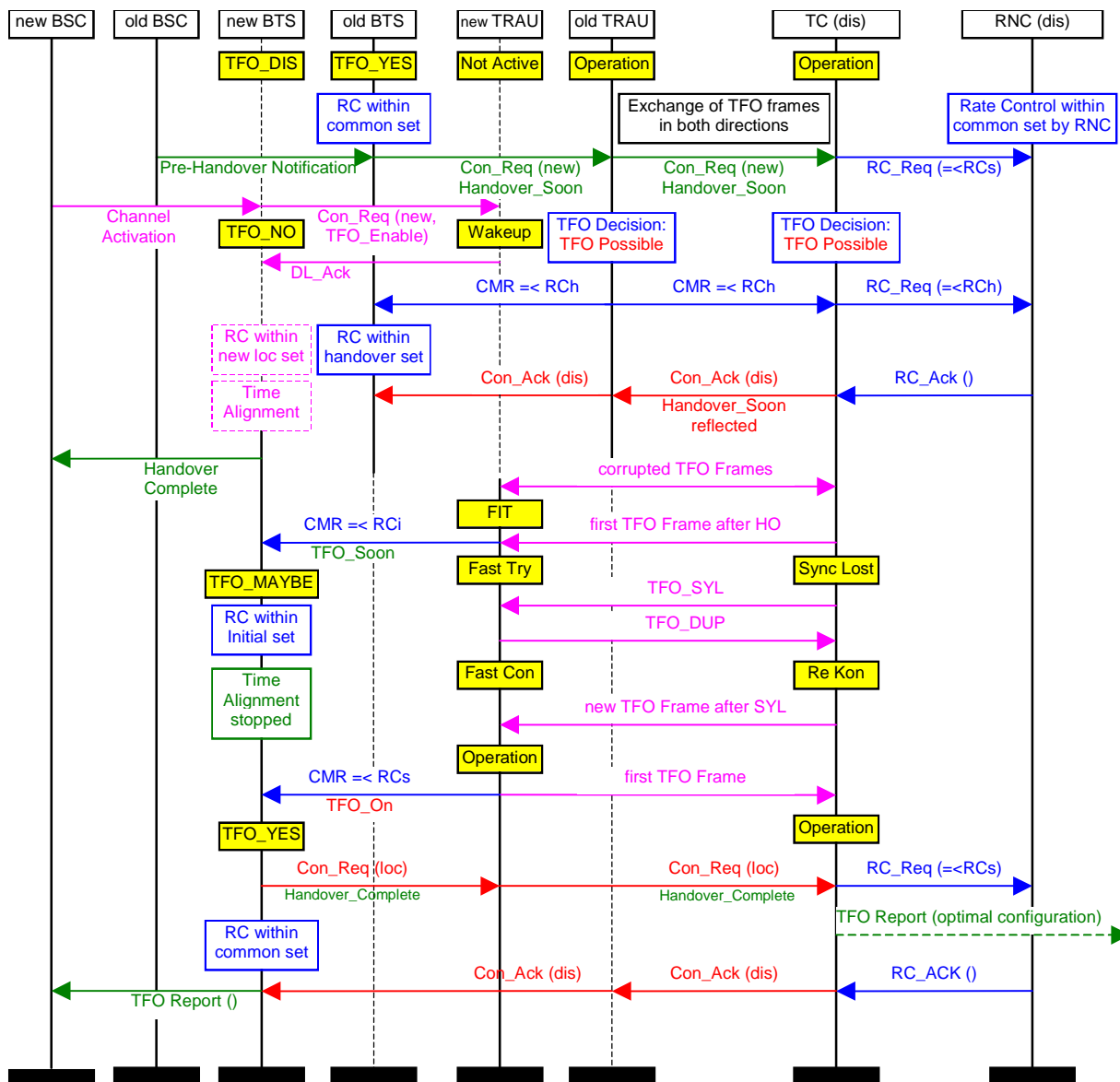


Figure G.8-1: Inter BSC TFO Handover (TRAU changes)

## G.9 Immediate Codec Type Optimization

The following protocol flow shows an example for Immediate Codec Type Optimization. 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.

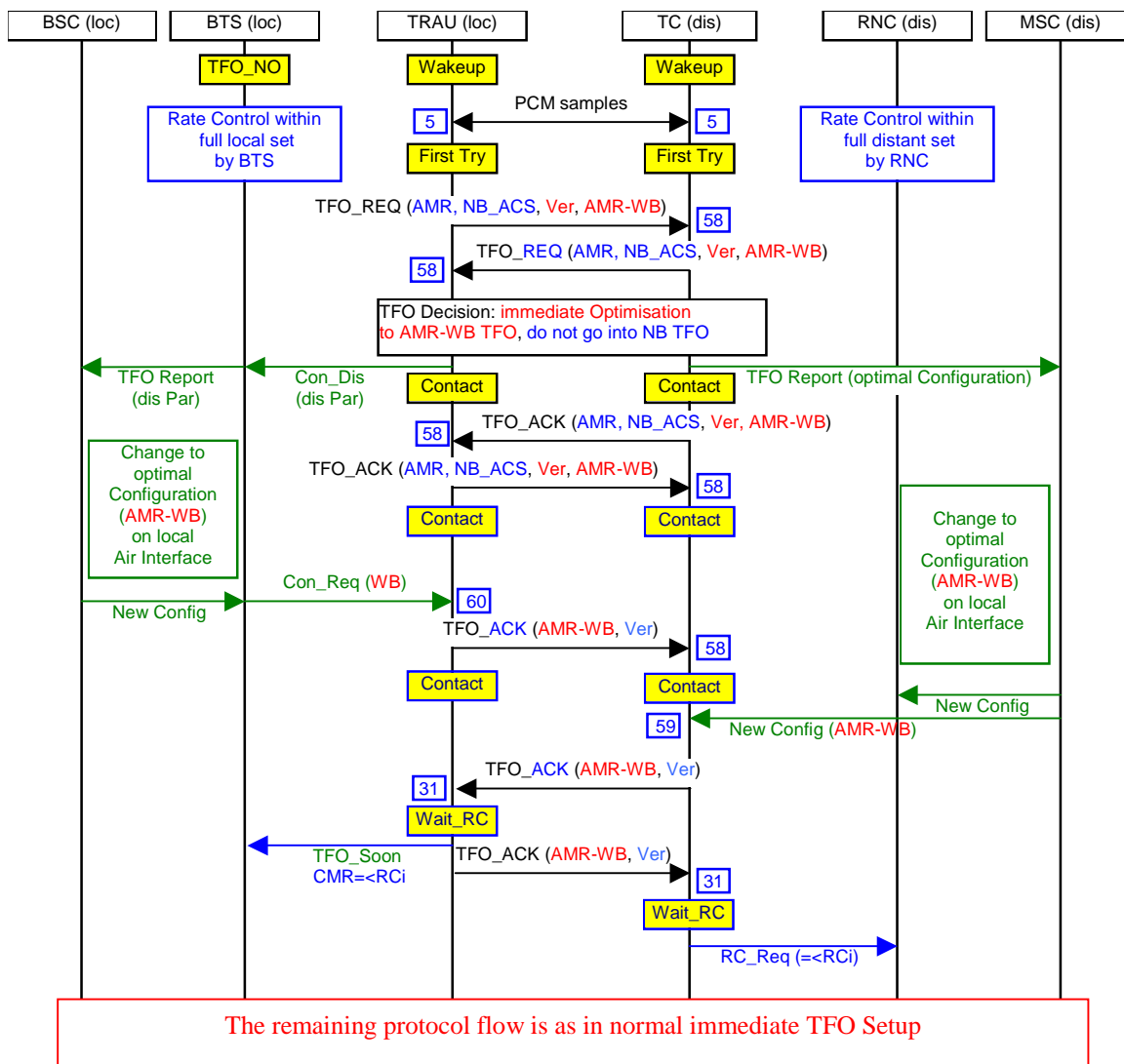


Figure G.9-1: Immediate Codec Type Optimisation for AMR-WB with AMR-NB at call setup

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.

# 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.

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 is used to code "REL 5.1.x".

Details for handling of the version in the TFO procedures 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

The definitions are given in Annex C.

### H.3.1.4 System Identification, TFO and DTX control

System Identification (Sys\_ID), DTXd, TFOE and OD are included in Generic Configuration Frames:

**Sys\_ID:** D21..D28, 8 bits (see TS 26.103 and Annex A.5).

**DTXd:** D29, 1 bit

**TFOE:** D30, 1 bit

**OD:** D31, 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.

**Active\_Codec\_Type:** D32..D39, 8 bits

**ACT\_Specific\_Length:** D40..D42, 3 bits.

**ACT\_Specific\_Extend:** D43, 1 bit.

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.

### 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

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 shall be placed at a fixed position at the very end of the 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:

- $b(1)D^{(N+8-1)} + b(2)D^{(N+8-2)} + \dots + b(N)D^8 + p(1)D^7 + p(2)D^6 + \dots + p(7)D^1 + p(8)$ ;
- $p(1) - p(8)$ : 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 Configuration 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.

Note: in that light the definition for AMR-WB is cumbersome, because it is somewhat "octet-hungry".

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 bits	0 bits	omitted in TS 28.062
Length_Indicator	8 bits	3 bits	"1.0.0" (4 octets following after the <u>Organisation Identifier Indicator</u> )
Extension_Indicator	-	1 bits	"0" no further Extension necessary
Compatibility_Information_Indicator	-	1 bit	"0" Compatibility_Information is omitted
Organisation_Identifier_Indicator	1 bit	1 bit	"0" Organisation_Identifier is omitted
Compatibility_Information	8 bits	0 bits	omitted, when not indicated
Organisation_Identifier	8 bits	0 bits	omitted, 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  $8 \times 8 = 64$  bits, while the coding in the Configuration frame takes  $6 + 8 + 3 \times 8 = 38$  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".

## Annex I (informative): Change History

Change history								
Date	TSG SA#	TSG Doc.	CR	Rev	Subject/Comment	Old	New	
2001-03	11	SP-010096			Version for Release 4		4.0.0	
2001-06	12	SP-010310	001	1	Reference to a deleted TFO message	4.0.0	4.1.0	
2001-08					Clean-up	4.1.0	4.1.1	
2001-12	14	SP-010704	002		Corrections	4.1.1	4.2.0	
2001-12	14	SP-010704	003		Corrections	4.1.1	4.2.0	
2002-03	15	SP-020090	004		Correction of OM & OD bits mapping in TFO 16k frames	4.2.0	4.3.0	
2002-03	15	SP-020090	005	1	Inclusion of the Non_Speech TFO frames in conditions for TFO_Frame	4.2.0	4.3.0	
2002-03	15	SP-020090	007	2	Corrections in TFO Protocol Tables	4.2.0	4.3.0	
2002-03	15	SP-020090	013		Corrected C-Code for AMR TFO decision rules	4.2.0	4.3.0	
2002-03	15	SP-020090	016		Corrections	4.2.0	4.3.0	
2002-03	15	SP-020091	009		Modification of TFO_Messages for AMR-WB introduction	4.3.0	5.0.0	
2002-03	15	SP-020091	010	2	Introduction of Generic Configuration Frames into TS 28.062, Annex H	4.3.0	5.0.0	
2002-03	15	SP-020091	015	1	Inclusion of AMR-WB codec types and codec type OHR_AMR (AMR-NB on 8PSK-HR channel) into TFO	4.3.0	5.0.0	
2002-03	15	SP-020092	14		Introduction of AMR-WB codec types and codec type OHR_AMR into reference implementation C-Code of AMR TFO decision rules	4.3.0	5.0.0	
2002-06	16	SP-020228	017	1	Editorial corrections and additions	5.0.0	5.1.0	
2002-06	16	SP-020228	019	2	Additional TFO_Message Elements for Immediate Codec Type Optimisation	5.0.0	5.1.0	
2002-06	16	SP-020228	020	2	Corrections to TS 28.062, sections 4 to 8	5.0.0	5.1.0	
2002-06	16	SP-020228	021	3	Corrections to TS 28.062, Annex C	5.0.0	5.1.0	
2002-06	16	SP-020228	022	2	TFO Version Handling	5.0.0	5.1.0	
2002-06	16	SP-020228	023	2	Configuration Exchange in Annex C	5.0.0	5.1.0	
2002-06	16	SP-020228	024	2	Corrections to Annex H	5.0.0	5.1.0	
2002-06	16	SP-020228	026	1	Corrections to sections 9 and 10	5.0.0	5.1.0	
2002-06	16	SP-020228	027	1	Immediate Codec Type Optimization	5.0.0	5.1.0	
2002-09	17	SP-020437	030	1	TFO-Signalling for allowed AMR-WB Configurations	5.1.0	5.2.0	
2002-09	17	SP-020437	031	2	Simplified TFO Decision for AMR-WB	5.1.0	5.2.0	
2002-09	17	SP-020438	33	2	TFO-Signalling for preferred AMR-NB Configurations	5.1.0	5.2.0	
2002-09	17	SP-020438	34		TFO Version Handling	5.1.0	5.2.0	
2002-12	18	SP-020696	036	1	Correction to TFO_Term state description	5.2.0	5.3.0	
2002-12	18	SP-020696	037	1	TFO version handling	5.2.0	5.3.0	
2002-12	18	SP-020696	038	1	Corrections to the TFO standard (wrong specification references)	5.2.0	5.3.0	
2002-12	18	SP-020696	039	1	Correction of TFO_REQ message for AMR-WB	5.2.0	5.3.0	
2003-09	21	SP-030451	040		Removal of Pre-Handover Notification for UMTS	5.3.0	5.4.0	
2004-12	26	SP-040845	043		TFO/TrFO Compatibility of UMTS_AMR and UMTS_AMR2	5.4.0	5.5.0	
2004-12	26	SP-040846	046		Operator Guidelines for UMTS_AMR	5.4.0	5.5.0	

---

## History

<b>Document history</b>		
V5.0.0	March 2002	Publication
V5.1.0	June 2002	Publication
V5.2.0	September 2002	Publication
V5.3.0	December 2002	Publication
V5.4.0	September 2003	Publication
V5.5.0	December 2004	Publication