Technical Specification Group Services and System Aspects ***TSGS#16(02)0227***
Meeting #16, Marco Island, Florida, USA, 10-13 June 2002

Source: **TSG-SA WG4**

Title: CRs to TS 28.062 on Corrections to Clauses 9 and 10, and Clarifications of Extendibility of TFO Messages (Release 4)

Document for: **Approval**

Agenda Item: **7.4.3**

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

Spec	CR	Rev	Phase	Subject	Cat	Vers	WG	Meeting	S4 doc
28.062	018	2	REL-4	Clarify Extendibility of TFO_Messages	F	4.3.0	S4	TSG-SA WG4#21	S4-020347
28.862	025		REL-4	Corrections to Clause 9 and 10	F	4.3.0	S4	TSG-SA WG4#21	S4-020313

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

S4-020347

CR-Form-v6.1

CHANGE REQUEST

⌘	TS 28.062	CR	CR 018	⌘ rev	2	⌘	Current version:	4.3.0	⌘
	Spec Title:	Inband Tandem Free Operation (TFO) of speech codecs							⌘

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

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

Title:	⌘	**Clarify Extendibility of TFO_Messages**
Source:	⌘	TSG SA WG4
Work item code:	⌘	TFO
Date:	⌘	2002-06-11
Category:	⌘	**F**
Release:	⌘	**REL-4**

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

Use one of the following releases:

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

Reason for change:	⌘	Clarify future Extendibility for TFO REQ Messages
Summary of change:	⌘	Some sentences to clarify Extendibility
Consequences if not approved:	⌘	REL-4 implementations are potentially not future-compatible

Clauses affected:	⌘	7
Other specs affected:	⌘	☐ Other core specifications ⌘ ☐ Test specifications ☐ O&M Specifications
Other comments:	⌘	see CR on chapter 7 for REL-5

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

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

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

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

7 TFO Messages

The TFO Messages, introduced in clause 6, follow the generic IS_Message principle defined in annex A.
The following definitions are provided for the ***Sender*** side:
TFO_REQ (): Identifies the source of the message as a TFO capable device, using a defined Codec_Type.
TFO_REQ contains the following parameters ():

- the System_Identification of the sender;
- the specific Local_Signature of the sender;
- the Local_Used_Codec_Type at sender side;
- possibly additional attributes for the Local_Used_Codec_Type
- possibly additionally 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 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). TFO_TRANS contains the following parameter ():

- the Local_Channel_Type (8 kbit/s or 16 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.

3GPP TSG-SA4 #21 **S4-020313**
Rennes, France, 13-17 May 2002

CR-Form-v6.1

CHANGE REQUEST

⌘	**TS 28.062**	**CR**	**025**	⌘ **rev**	**-**	⌘ Current version:	**4.3.0** ⌘
	Spec Title:	Inband Tandem Free Operation (TFO) of speech codecs					⌘

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

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

Title:	⌘	**Corrections to Clause 9 and 10 (Rel-4)**
Source:	⌘	TSG SA WG4
Work item code:	⌘	TFO
Date:	⌘	2002-06-11
Category:	⌘	**F**
Release:	⌘	**REL-4**

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

Use one of the following releases:

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

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

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

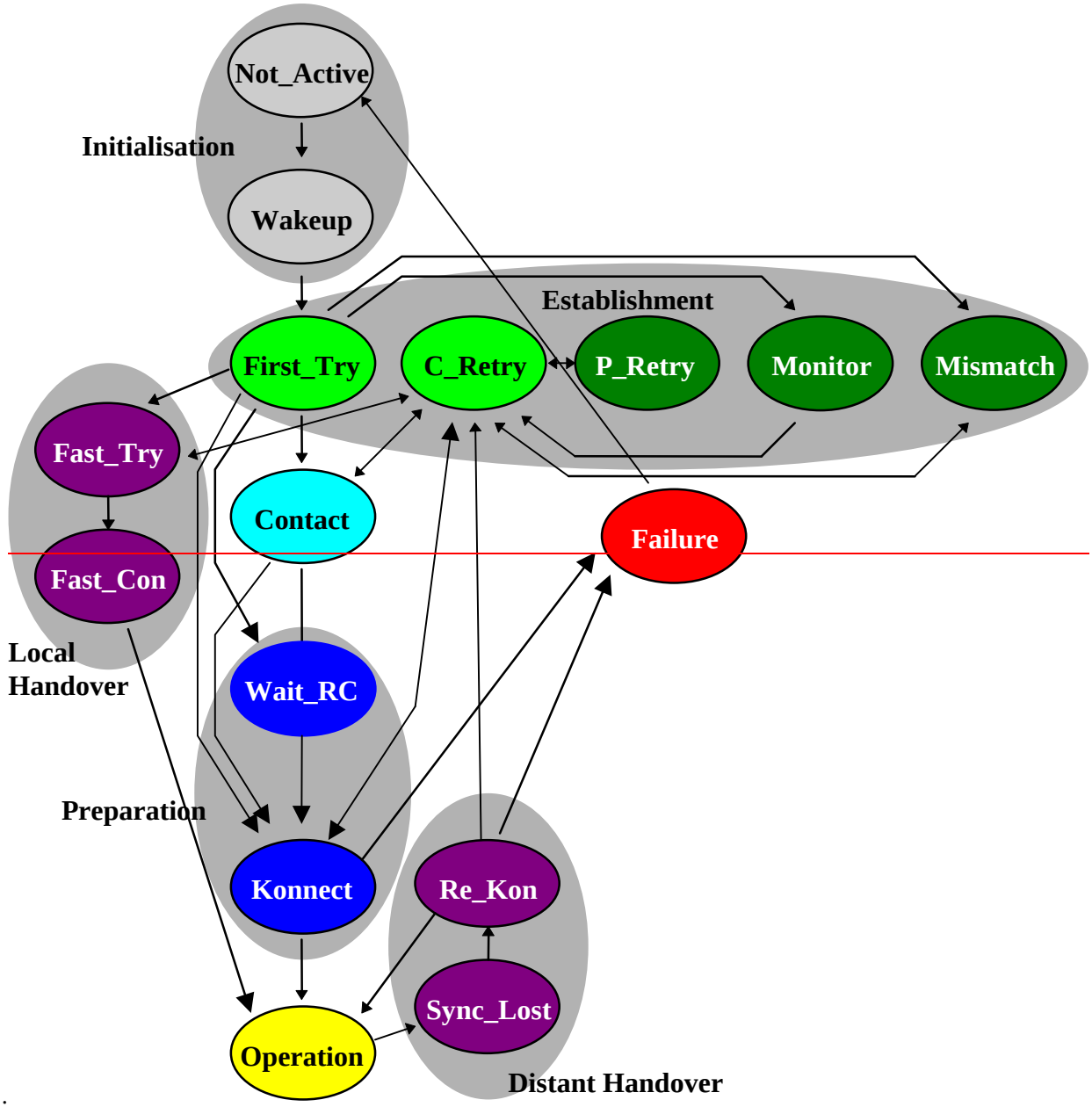
How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

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

Note to the Editor: Text with yellow background is used for special notes. It shall not appear in the specification. It is intended to provide additional information on editing and implementing the CR.

9 TFO State Machine

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


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

.

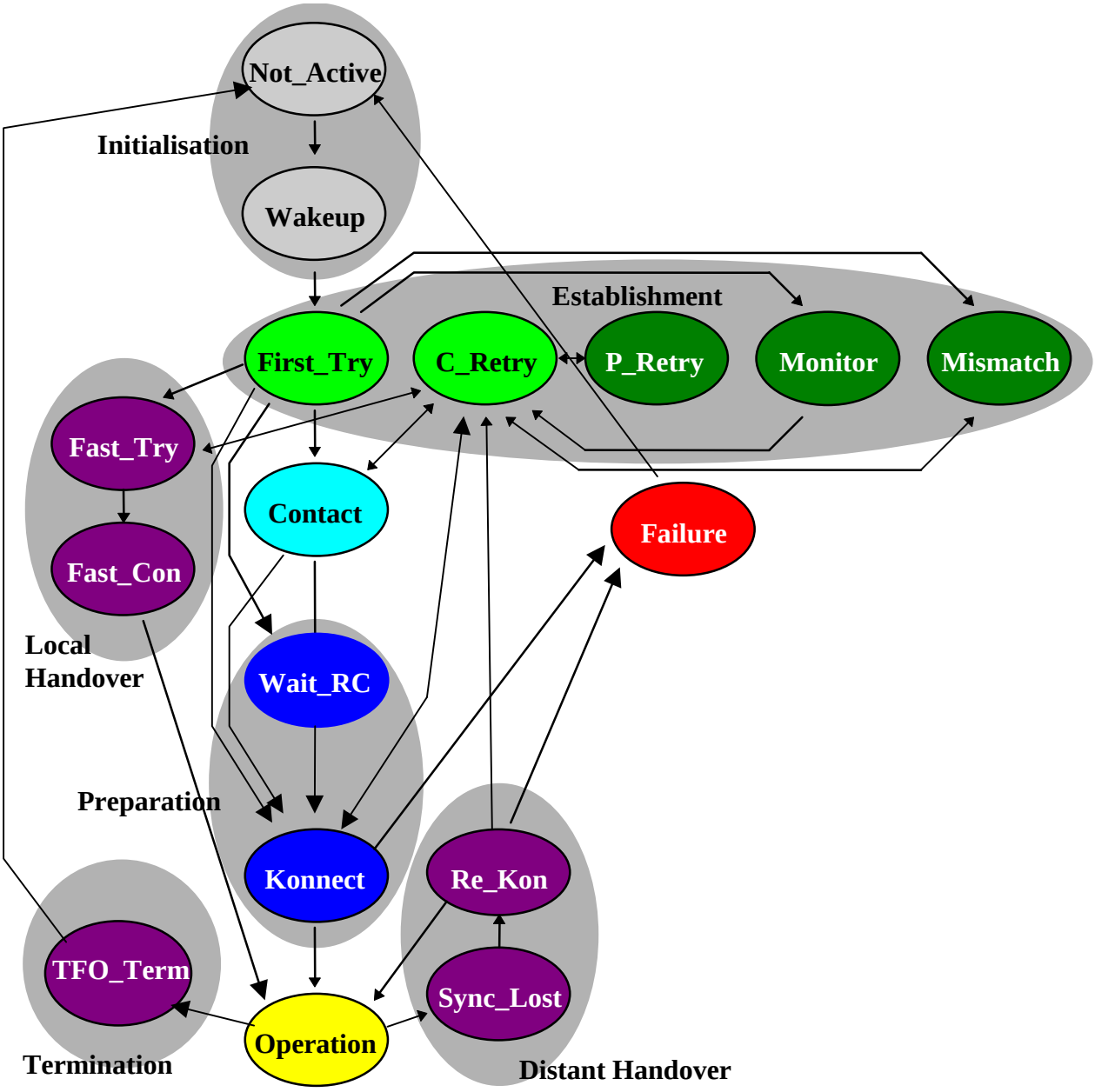


Figure 9-1: TFO_Protocol State Machine with most important transitions

There are five main States:

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

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

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

- Termination (• TFO_Term).

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

10.4 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 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 codec types). In 3G, this means that the Iu User Plan is initialised.
3	TFO_Disable	The event TFO_Disable occurs when TFO is disabled by the controlling entity. In GSM, the TFO_Disable event is also controlled by the TFOE bit of AMR 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)	This event occurs when the local used codec type changes and either the condition NA_TP or the condition A_TP is fulfilled.
15	New_Local_Codec and TM	This event occurs when the local used codec type changes and the condition TM is fulfilled.
14	New_Local_Config and (NA_TP | A_TP)	This event occurs when an AMR codec type is used and the local codec configuration changes and the condition A_TP is fulfilled.
16	New_Local_Config and TM	This event occurs when an AMR codec type is used and the local codec configuration changes and the condition TM is fulfilled.
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).
		delete row!
		delete row!
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	This event occurs when a TFO_REQ message is received, either the condition NA_TP or the condition A_TP is fulfilled, and the distant signature is different from the local signature and old (local) signature.
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	This event occurs when a TFO_REQ message is received, the condition TM is fulfilled, and the distant signature is different from the local signature.
9	TFO_ACK and NA_TP and Dsig==Lsig	This event occurs when a TFO_ACK message is received, the condition NA_TP is fulfilled, and the local and distant signatures are equal.
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 Dsig==?	This event occurs when a TFO_ACK message is received and the condition TM is fulfilled. The distant signature is ignored for this event.
31	TFO_ACK and A_TP and Dsig==Lsig	This event occurs when a TFO_ACK message is received, the condition A_TP is fulfilled, and the distant signature is equal to the local signature.
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 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	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

	Con_Req & TC	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.

10.5 Actions Table

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

Table 10.5-1: Definition of Syntax for Action Table

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

The following notations are used in Table 10.5-2.

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

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

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

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

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

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

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

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

Tx_TRAU := … means that a message is sent to the downlink Transmit Process of the Transcode **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;	As 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.
		delete row!
		delete row!
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 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 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 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 codec types and steers the rate control down to the Hand_Over_Mode in order to continue TFO after hand-over using this mode.
CA	Tx_TFO := Con_Ack();	Send a Con_Ack (config frame) to the distant TRAU/TC.
CA1	Wait round trip time to RNC; Tx_TFO := Con_Ack();	Wait round trip time to RNC (e.g. send first a RC_REQ to the RNC and wait for the corresponding RC_ACK). Then send a Con_Ack to the distant TRAU/TC.
CR	TX_TFO := Con_Req();	This action is conditional and only relevant for 3G systems (TC). If the entity is a TC then send a Con_Req with TFO_Disable to the distant TRAU/TC.

10.6 Protocol Tables

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

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

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

Table 10.6-2: PCM_Non_Idle and Loopback Handling

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

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

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

Table 10.6-4: In Call Modification and Handover

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

Table 10.6-5: Special Matching TFO Messages

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

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

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

Table 10.6-7 AMR Case: TFO_TRANS, TFO_ACK, RC_ack

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

Table 10.6-8 Handover_Soon

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

Table 10.6-9: Mismatching TFO_TRANS and TFO Frames

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

Table 10.6-10: Local Events, TFO_FILL, TFO_NORMAL

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

Table 10.6-11: Special Events, Timeouts

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

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

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

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

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

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

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