Appendix 2

(* Probes for PLMN, PLMN only *)

(* reqest detach for a member not attached, probe 2 *)

```
ch_uplink  ! init1 ! req_detach ! nomsg;
i; (* ch_answer ! init1 ! ack_detach ! nomsg *)
PROBE  ! P_2;
ch_downlink  ! init1 ! ack_detach ! nomsg;
probe  ! p_52;
```

(* request termination for a member not initiator, probe 17 *)

```
ch_uplink  ! init1 ! req_attach ! encode(1);
i; (* ch_answer ! init1 ! ack_attach ! nomsg *)
probe  ! p_0;
ch_downlink  ! init1 ! ack_attach ! nomsg;
probe  ! p_52;
ch_uplink  ! init1 ! req_init ! encode(1,n_way,qos1,1,no_join_leave_ind,send_to_all_ind,no_call_not);
i; (* ch_id ! succ(noc_id) *)
i; (* ch_answer ! init1 ! ack_c_init ! encode(1,succ(noc_id),cipher_key(succ(noc_id))) *)
i; (* ch_answer ! insert ! succ(noc_id) !
encode(1,n_way,qos1,1,no_join_leave_ind,send_to_all_ind,no_call_not) *)
i; (* ch_answer ! insert ! 1 of imgi ! succ(noc_id) *)
probe  ! p_12;
i; (* exit *)
probe  ! p_13;
ch_downlink  ! init1 ! ack_c_init ! encode(1,succ(noc_id),cipher_key(succ(noc_id)));
probe  ! p_52;
probe  ! p_56;
probe  ! p_60;
ch_uplink  ! mem1 ! req_attach ! encode(1);
i; (* ch_answer ! mem1 ! ack_attach ! nomsg *)
probe  ! p_0;
ch_downlink  ! mem1 ! ack_attach ! nomsg;
probe  ! p_52;
ch_uplink  ! mem1 ! req_term ! encode(1,succ(noc_id));
i; (* ch_answer ! mem1 ! err_rejterm ! encode(1,succ(noc_id),cause) *)
PROBE  ! P_17;
ch_downlink  ! mem1 ! err_rejterm ! encode(1,succ(noc_id),cause);
probe  ! p_51;
```

(* request join for a member not in right group, probe 26 *)

```
ch_uplink  ! mem3 ! req_join ! encode(1,succ(noc_id));
i; (* ch_answer ! mem3 ! err_rejjoin ! encode(1,succ(noc_id),cause) *)
PROBE  ! P_26;
ch_downlink  ! mem3 ! err_rejjoin ! encode(1,succ(noc_id),cause);
probe  ! p_51;
```

(* request send data with a DTM=noway, probe 35 *)

```
ch_uplink  ! init1 ! req_init ! encode(1,noway,qos1,1,no_join_leave_ind,no_send_to_all,no_call_not);
i; (* ch_id ! succ(succ(noc_id)) *)
i; (* ch_answer ! init1 ! ack_c_init ! encode(1,succ(succ(noc_id)),cipher_key(succ(succ(noc_id)))) *)
i; (* ch_answer ! insert ! succ(succ(noc_id)) !
encode(1,noway,qos1,1,no_join_leave_ind,no_send_to_all,no_call_not) *)
i; (* ch_answer ! insert ! 1 of imgi ! succ(succ(noc_id)) *)
probe  ! p_12;
i; (* exit *)
probe  ! p_13;
ch_downlink  ! init1 ! ack_c_init ! encode(1,succ(succ(noc_id)),cipher_key(succ(succ(noc_id))));
probe  ! p_52;
probe  ! p_56;
probe  ! p_60;
ch_uplink  ! mem1 ! req_join ! encode(1,succ(succ(noc_id)));
i; (* ch_answer ! ask ! succ(succ(noc_id)) ! nomsg *)
i; (* ch_answer ! succ(succ(noc_id)) !
ci(succ(succ(noc_id)),noway,qos1,1,no_join_leave_ind,no_send_to_all,no_call_not) *)
i; (* ch_answer ! mem1 ! ack_join ! encode(1,succ(succ(noc_id)),cipher_key(succ(succ(noc_id)))) *)
probe  ! p_24;
ch_downlink  ! mem1 ! ack_join ! encode(1,succ(succ(noc_id)),cipher_key(succ(succ(noc_id))));
probe  ! p_52;
probe  ! p_58;
ch_uplink  ! mem1 ! req_data ! encode(1,succ(succ(noc_id)),data);
i; (* ch_answer ! ask ! succ(succ(noc_id)) ! nomsg *)
i; (* ch_answer ! succ(succ(noc_id)) !
ci(succ(succ(noc_id)),noway,qos1,1,no_join_leave_ind,no_send_to_all,no_call_not) *)
i; (* ch_answer ! mem1 ! err_rejdat ! encode(1,succ(succ(noc_id)),cause) *)
PROBE  ! P_35;
ch_downlink  ! mem1 ! err_rejdat ! encode(1,succ(succ(noc_id)),cause);
probe  ! p_51;
probe  ! p_58;
```

(* request send data for a member not participant to a call, probe 36 *)

```
 ch_uplink  ! mem1 ! req_data ! encode(1,succ(noc_id),data);
 i; (* ch_answer ! mem1 ! err_rejdat ! encode(1,succ(noc_id),cause) *)
 PROBE  ! P_36;
 ch_downlink  ! mem1 ! err_rejdat ! encode(1,succ(noc_id),cause);
 probe  ! p_51;
```

(* request change geozone for a member not attached, probe 37 *)

```
ch_uplink  ! adm1 ! req_change ! encode(1);
i; (* ch_answer ! adm1 ! ack_change ! nomsg *)
PROBE  ! P_37;
ch_downlink  ! adm1 ! ack_change ! nomsg;
probe  ! p_52;
```

(* probe 40 not testable *)

(* probe 63 useless code *)

(* Probes for MS, Subscriber only *)

(* request call termination, rejected for initiator, probe 17 *)

```
ch_uplink  ! adm1 ! req_attach ! encode(1);
ch_downlink  ! adm1 ! ack_attach ! nomsg;
probe  ! p_0;
ch_uplink  ! adm1 ! req_init ! encode(1,n_way,qos1,1,join_leave_ind,send_to_all_ind,no_call_not);
ch_downlink  ! adm1 ! ack_c_init ! encode(1,succ(succ(noc_id)),cipher_key(succ(succ(noc_id))));
probe  ! p_4;
ch_uplink  ! adm1 ! req_term ! encode(1,succ(succ(noc_id)));
ch_downlink  ! adm1 ! err_rejterm ! encode(1,succ(succ(noc_id)),cause);
PROBE  ! P_17;
```

(* request send data, rejected for initiator, probe 18 *)

```
ch_uplink  ! adm1 ! req_attach ! encode(1);
ch_downlink  ! adm1 ! ack_attach ! nomsg;
probe  ! p_0;
ch_uplink  ! adm1 ! req_init ! encode(1,n_way,qos1,1,join_leave_ind,send_to_all_ind,no_call_not);
ch_downlink  ! adm1 ! ack_c_init ! encode(1,succ(noc_id),cipher_key(succ(noc_id)));
probe  ! p_4;
ch_uplink  ! adm1 ! req_data ! encode(1,succ(noc_id),data);
ch_downlink  ! adm1 ! err_rejdat ! encode(1,succ(noc_id),cause);
PROBE  ! P_18;
i; (* exit *)
probe  ! p_20;
```

(* request send data, rejected for participant with a N-way DTM, probe 23 *)

```
ch_uplink  ! mem1 ! req_attach ! encode(1);
ch_downlink  ! mem1 ! ack_attach ! nomsg;
probe  ! p_0;
ch_uplink  ! mem1 ! req_status ! encode(1,noc_id,insert(sta_callids,{}));
ch_downlink  ! mem1 ! ack_status ! encode(1,insert(succ(noc_id),{}));
probe  ! p_7;
ch_uplink  ! mem1 ! req_join ! encode(1,succ(noc_id));
ch_downlink  ! mem1 ! ack_join ! encode(1,succ(noc_id),cipher_key(noc_id));
probe  ! p_10;
ch_uplink  ! mem1 ! req_data ! encode(1,succ(noc_id),data);
ch_downlink  ! mem1 ! err_rejdat ! encode(1,succ(noc_id),cause);
PROBE  ! P_23;
i; (* exit *)
probe  ! p_27;
```